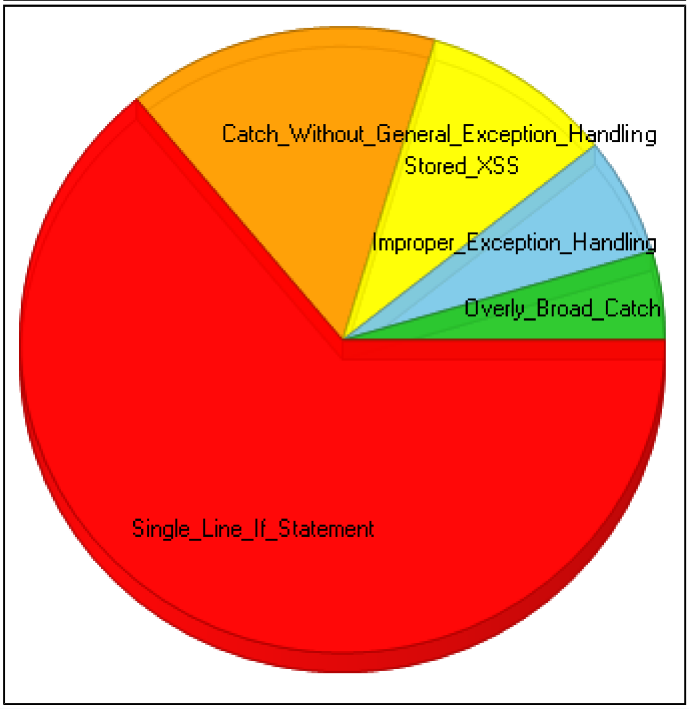
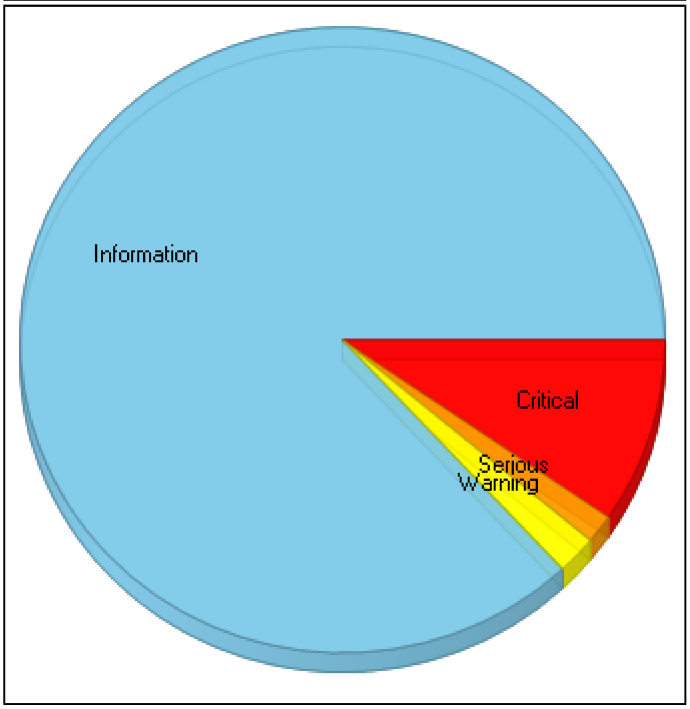
Checkmarx Report

Project Name:	**Opennms**
Load Time:	**27/02/2008 15:54:00**
CxSuite Version:	**2.4.3**

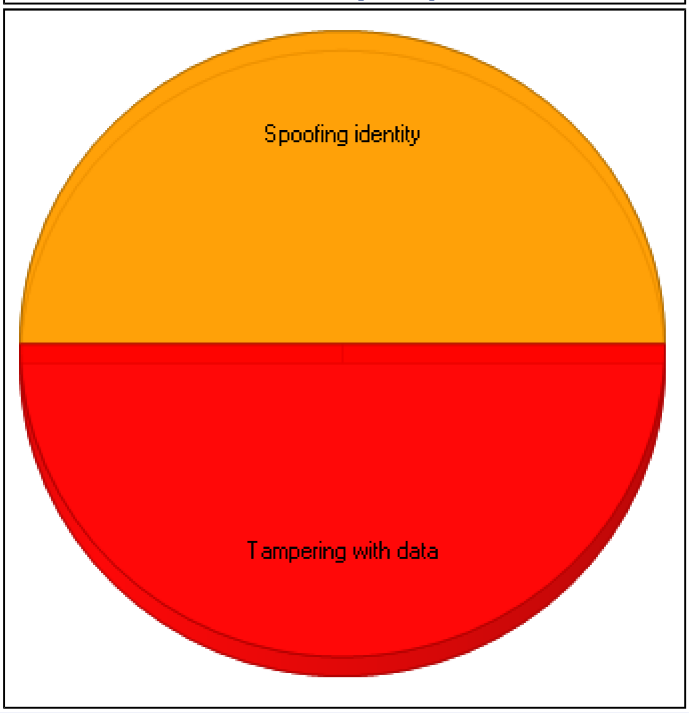
Problems by Queries



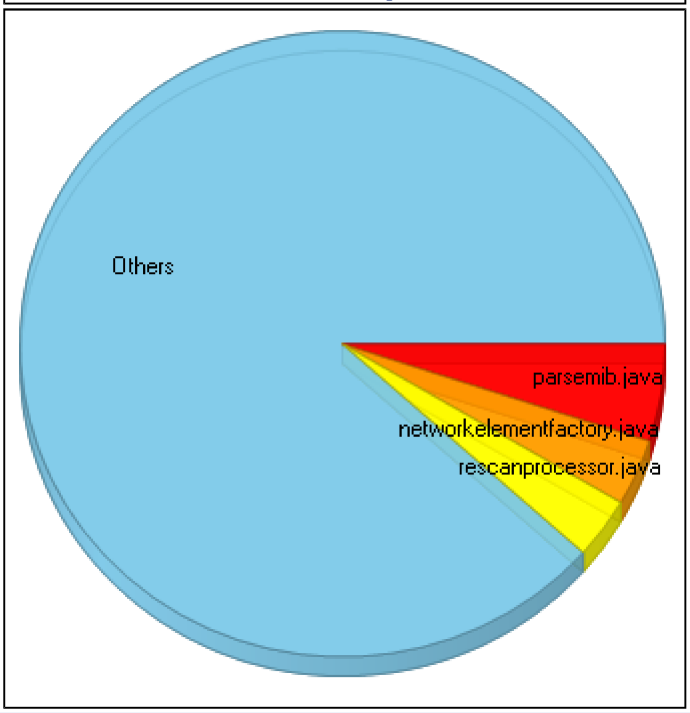
Problems by Severity



Problems by Impacts



Problems by Files



Summary

Query	Group	Problems found	Severity
Improper Exception Handling	Best Coding Practice	538	
Unclosed Objects	Best Coding Practice	256	
Catch Without General Exception Handling	Best Coding Practice	1379	
Single Line If Statement	Best Coding Practice	5637	
Empty Catch	Best Coding Practice	232	
Threads in WebApp	Best Coding Practice	5	
Sockets in WebApp	Best Coding Practice	2	
No Default Case	Best Coding Practice	11	
Erroneous String Compare	Best Coding Practice	9	
Magic Numbers	Best Coding Practice	65	
Confusing Naming	Best Coding Practice	6	
Leftover Debug Code	Best Coding Practice	28	
Missing Catch Block	Best Coding Practice	106	
Missing XML Validation	Best Coding Practice	4	
Overly Broad Catch	Best Coding Practice	405	
Overly Broad Throws	Best Coding Practice	258	
Use of System Output Stream	Best Coding Practice	299	
SQL injection	High Risk	4	
Second Order SQL Injection	High Risk	4	
Reflected XSS All Clients	High Risk	157	
Stored XSS	High Risk	853	

Query	Group	Problems found	Severity
Log Forgery	Low Visibility	43	
DoS by Unreleased Resources	Low Visibility	40	
Hardcoded Password	Low Visibility	5	
URL Redirection Attack	Low Visibility	44	
Equals without GetHashCode	Low Visibility	70	
Improper Transaction Handling	Low Visibility	10	
Verbose Error Reporting	Medium Threat	71	
Files Manipulation	Medium Threat	21	
XSRF	Medium Threat	7	
Parameter Tampering	Medium Threat	26	
Blind SQL Injections	Medium Threat	4	
Singleton HTTPServlet	Medium Threat	28	

Metrics

Metric Name	Value
Number of types	1189
Number of Classes	1110
Number of Interfaces	79
Avg methods per class	7
Number of Constructors	1100
Avg statements per Method	8
Number of Control Statements	12317
Number of Methods	8263
Lines of Code	266501

Top 10 Files

File name	Problems found
opennms\contrib\mibparser\src\parsemib.java	533
opennms\src\web\src\org\opennms\web\element\networkelementfactory.java	365
opennms\src\services\org\opennms\netmgt\capsd\rescanprocessor.java	335
opennms\src\install\org\opennms\install\installer.java	307
opennms\src\web\src\org\opennms\web\event\eventfactory.java	239
opennms\src\services\org\opennms\netmgt\collectd\snmpcollector.java	210
opennms\src\web\src\org\opennms\web\notification\notificationmodel.java	201
opennms\src\services\org\opennms\netmgt\capsd\suspecteventprocessor.java	197
opennms\contrib\mibparser\src\parsemibtokenmanager.java	170
opennms\src\services\org\opennms\netmgt\capsd\dbnodeentry.java	154

	Often Misused: Exception Handling
CWE ID	248
Description	Failing to catch an exception thrown from a dangerous function, can potentially cause the program to crash.
Demonstrative Examples	The _alloca() function allocates memory on the stack. If an allocation request is too large for the available stack space, _alloca() throws an exception. If the exception is not caught, the program will crash, potentially enabling a denial of service attack. _alloca() has been deprecated as of Microsoft Visual Studio 2005®. It has been replaced with the more secure _alloca_s().
	EnterCriticalSection() can raise an exception, potentially causing the program to crash. Under operating systems prior to Windows 2000, the EnterCriticalSection() function can raise an exception in low memory situations. If the exception is not caught, the program will crash, potentially enabling a denial of service attack.
Node Relationships	ChildOf - API Abuse (227) Peer - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	7 Pernicious Kingdoms - Often Misused: Exception Handling
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
115. public static InputStream post(URL url, InputStream dataStream, String
username, String password, int bufSize) throws IOException //httputils.java
     ...
157. int bytesRead = dataStream.read(b, 0, bufSize);
```

[Table of Contents] [Description]

Path 2:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
115. public static InputStream post(URL url, InputStream dataStream, String
username, String password, int bufSize) throws IOException //httputils.java
```

```
...
159. ostream.write(b, 0, bytesRead);
```

[Table of Contents] [Description]

Path 3:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
115. public static InputStream post(URL url, InputStream dataStream, String
username, String password, int bufSize) throws IOException //httputils.java
...
160. bytesRead = dataStream.read(b, 0, bufSize);
```

[Table of Contents] [Description]

Path 4:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
...
244. ostream.write(b, 0, bytesRead);
```

[Table of Contents] [Description]

Path 5:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
484. public void readTables(Reader reader) throws Exception //installer.java
...
502. while ((line = r.readLine()) != null) {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
...
627. ResultSet rs = st.executeQuery("SELECT version()");
```

[Table of Contents] [Description]

Path 7:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
        ...
633. String versionString = rs.getString(1);
```

[Table of Contents] [Description]

Path 8:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
719. public void checkOldTables() throws Exception //installer.java
        ...
721. ResultSet rs = st.executeQuery("SELECT relname FROM pg_class "
```

[Table of Contents] [Description]

Path 9:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
719. public void checkOldTables() throws Exception //installer.java
        ...
728. oldTables.add(rs.getString(1));
```

[Table of Contents] [Description]

Path 10:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
        ...
806. ResultSet rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 11:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
        ...
809. int count = rs.getInt(1);
```

[Table of Contents] [Description]

Path 12:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
        ...
813. rs = st.executeQuery("SELECT count(*) FROM " + table);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
        ...
815. int total = rs.getInt(1);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
843. public void fixConstraint() throws Exception //installer.java
        ...
897. int num = st.executeUpdate(query);
```

[Table of Contents] [Description]

Path 15:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
902. public boolean databaseUserExists() throws SQLException //installer.java
        ...
906. ResultSet rs = st.executeQuery("SELECT usename FROM pg_user WHERE "
```

[Table of Contents] [Description]

Path 16:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
917. public void databaseAddUser() throws SQLException //installer.java
        ...
919. st.execute("CREATE USER " + m_user + " WITH PASSWORD '" + m_pass
```

[Table of Contents] [Description]

Path 17:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
923. public boolean databaseDBExists() throws SQLException //installer.java
      ...
927. ResultSet rs = st.executeQuery("SELECT datname from pg_database "
```

[Table of Contents] [Description]

Path 18:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
938. public void databaseAddDB() throws Exception //installer.java
      ...
941. .execute("CREATE DATABASE " + m_database
```

[Table of Contents] [Description]

Path 19:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
      ...
993. rs = st.executeQuery("SELECT relname FROM pg_class "
```

[Table of Contents] [Description]

Path 20:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
      ...
998. st.execute("DROP SEQUENCE " + sequence);
```

[Table of Contents] [Description]

Path 21:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
      ...
1005. st.execute("CREATE SEQUENCE " + sequence + " minvalue " + minvalue);
```

[Table of Contents] [Description]

Path 22:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
        ...
1006.   st.execute("GRANT ALL on " + sequence + " TO " + m_user);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
        ...
1030.   rs = st.executeQuery("SELECT relname FROM pg_class "
```

[Table of Contents] [Description]

Path 24:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
        ...
1037.   st.execute("DROP TABLE " + table + m_cascade);
```

[Table of Contents] [Description]

Path 25:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
        ...
1044.   st.execute("CREATE TABLE " + table + " (" + create + ")");
```

[Table of Contents] [Description]

Path 26:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
        ...
1049.   st.execute("GRANT ALL ON " + table + " TO " + m_user);
```

[Table of Contents] [Description]

Path 27:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
      |...
      1064. st.execute("CREATE TABLE " + table + " (" + create
```

[Table of Contents] [Description]

Path 28:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
      |...
      1066. st.execute("GRANT ALL ON " + table + " TO " + m_user);
```

[Table of Contents] [Description]

Path 29:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
      |...
      1091. rs = st.executeQuery("SELECT relname FROM pg_class "
```

[Table of Contents] [Description]

Path 30:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
      |...
      1099. st.execute(getIndexFromSQL(index));
```

[Table of Contents] [Description]

Path 31:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1107. public Map getTypesFromDB() throws SQLException //installer.java
      |...
      1116. rs = st.executeQuery("SELECT oid,typname,typlen FROM pg_type");
```

[Table of Contents] [Description]

Path 32:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1165. public void fixData() throws Exception //installer.java
      ...
1169. st.execute("UPDATE ipinterface SET issnmpprimary='N' "
```

[Table of Contents] [Description]

Path 33:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1165. public void fixData() throws Exception //installer.java
      ...
1171. st.execute("UPDATE service SET servicename='SSH' "
```

[Table of Contents] [Description]

Path 34:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1165. public void fixData() throws Exception //installer.java
      ...
1173. st.execute("UPDATE snmpinterface SET snmpipadentnetmask=NULL");
```

[Table of Contents] [Description]

Path 35:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
      ...
1222. rs = st.executeQuery("SELECT encoding FROM pg_database WHERE "
```

[Table of Contents] [Description]

Path 36:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
      ...
1225. if (rs.getInt(1) == 5 || rs.getInt(1) == 6) {
```

[Table of Contents] [Description]

Path 37:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
        ...
1225. if (rs.getInt(1) == 5 || rs.getInt(1) == 6) {
```

[Table of Contents] [Description]

Path 38:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
        ...
1401. while ((line = r.readLine()) != null) {
```

[Table of Contents] [Description]

Path 39:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
        ...
1434. st.execute("DROP FUNCTION " + function + "(" + columns
```

[Table of Contents] [Description]

Path 40:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
        ...
1436. st.execute(create.toString());
```

[Table of Contents] [Description]

Path 41:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
        ...
1442. st.execute(create.toString());
```

[Table of Contents] [Description]

Path 42:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1469. public boolean functionExists(String function, int[]
columnTypes, //installer.java
      ...
1483. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 43:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1501. public void checkServerXmlOldOpennmsContext() throws
Exception //installer.java
      ...
1518. while ((line = r.readLine()) != null) {
```

[Table of Contents] [Description]

Path 44:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1587. public void updateTomcatConf() throws Exception //installer.java
      ...
1603. while ((line = r.readLine()) != null) {
```

[Table of Contents] [Description]

Path 45:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1587. public void updateTomcatConf() throws Exception //installer.java
      1219. m_out.print("- checking if database \"" + m_database
```

[Table of Contents] [Description]

Path 46:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      ...
1691. st.execute("CREATE FUNCTION iplike(text,text) RETURNS bool " + "AS '"
```

[Table of Contents] [Description]

Path 47:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      2401. m_out.println(" "
```

[Table of Contents] [Description]

Path 48:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1706. st.execute("CREATE FUNCTION plpgsql_call_handler () "
```

[Table of Contents] [Description]

Path 49:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      2412. m_out.println(" -i insert data into the database");
```

[Table of Contents] [Description]

Path 50:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1721. st.execute("CREATE TRUSTED PROCEDURAL LANGUAGE 'plpgsql' "
```

[Table of Contents] [Description]

Path 51:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1840. public boolean tableExists(String table) throws SQLException //installer.java
      |...
      1844. rs = st.executeQuery("SELECT DISTINCT tablename FROM pg_tables "
```

[Table of Contents] [Description]

Path 52:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |
      ...
      1876. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 53:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |
      ...
      1880. c.setName(rs.getString(1));
```

[Table of Contents] [Description]

Path 54:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |
      ...
      1881. c.parseColumnType(rs.getString(2));
```

[Table of Contents] [Description]

Path 55:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |
      ...
      1882. c.setNotNull(rs.getBoolean(3));
```

[Table of Contents] [Description]

Path 56:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |
      ...
      1905. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 57:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      1909. if (rs.getString(2).equals("p")) {
```

[Table of Contents] [Description]

Path 58:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      1910. constraint = new Constraint(rs.getString(1), rs
```

[Table of Contents] [Description]

Path 59:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      1911. .getString(4));
```

[Table of Contents] [Description]

Path 60:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      1912. } else if (rs.getString(2).equals("f")) {
```

[Table of Contents] [Description]

Path 61:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
```

```
1913. constraint = new Constraint(rs.getString(1), rs
```

[Table of Contents] [Description]

Path 62:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      ...
1914. .getString(4), rs.getString(5), rs.getString(6), rs
```

[Table of Contents] [Description]

Path 63:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      ...
1914. .getString(4), rs.getString(5), rs.getString(6), rs
```

[Table of Contents] [Description]

Path 64:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      ...
1914. .getString(4), rs.getString(5), rs.getString(6), rs
```

[Table of Contents] [Description]

Path 65:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      ...
1915. .getString(3));
```

[Table of Contents] [Description]

Path 66:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1918. + rs.getString(2) + "\" in constraint \""
```

[Table of Contents] [Description]

Path 67:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1919. + rs.getString(1) + "\"");
```

[Table of Contents] [Description]

Path 68:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1944. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 69:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1946. Constraint constraint = new Constraint(rs.getString(1), rs
```

[Table of Contents] [Description]

Path 70:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1947. .getString(2));
```

[Table of Contents] [Description]

Path 71:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1964. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 72:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1968. fkey = rs.getInt(1);
```

[Table of Contents] [Description]

Path 73:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1972. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 74:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
      |1976. fdel = rs.getInt(1);
```

[Table of Contents] [Description]

Path 75:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
      |...
```

```
1988. rs = st.executeQuery(query);
```

[Table of Contents] [Description]

Path 76:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
...
1991. String name = rs.getString(1);
```

[Table of Contents] [Description]

Path 77:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
...
1992. String[] args = new String(rs.getBytes(2)).split("\000");
```

[Table of Contents] [Description]

Path 78:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
...
1994. Constraint constraint = new Constraint(rs.getString(1),
```

[Table of Contents] [Description]

Path 79:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
...
1995. args[4], args[2], args[5], (rs.getInt(3) == fkey) ? "a"
```

[Table of Contents] [Description]

Path 80:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |1402. line = line.trim();
```

[Table of Contents] [Description]

Path 81:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      |...
      |2242. ResultSet rs = st.executeQuery("SELECT count(*) FROM " + oldTable);
```

[Table of Contents] [Description]

Path 82:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      |...
      |2244. long num_rows = rs.getLong(1);
```

[Table of Contents] [Description]

Path 83:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      |...
      |2272. rs = select.executeQuery();
```

[Table of Contents] [Description]

Path 84:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      |...
      |2292. obj = rs.getObject(change.getSelectIndex());
```

[Table of Contents] [Description]

Path 85:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      |...
      2378. st.execute("INSERT INTO events (eventid, eventuei, eventtime, "
```

[Table of Contents] [Description]

Path 86:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
83. public void destroyDatabase() throws SQLException //installerdbtest.java
      |...
      85. st.execute("DROP DATABASE " + m_testDatabase);
```

[Table of Contents] [Description]

Path 87:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
110. public void testBug1006HasOldTables() throws
SQLException //installerdbtest.java
      |...
      120. st.execute("CREATE TABLE " + table + " ( foo integer )");
```

[Table of Contents] [Description]

Path 88:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
137. public void executeSQL(String[] commands) throws
SQLException //installerdbtest.java
      |...
      144. st.execute(commands[i]);
```

[Table of Contents] [Description]

Path 89:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
123. public void testServerXmlContext() throws Exception //installerwebapptest.java
      |...
      138. w.print(context);
```

[Table of Contents] [Description]

Path 90:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void printStackTrace(PrintWriter
writer) //snmpbadconversionexception.java
      |...
      |119. writer.println("");
```

[Table of Contents] [Description]

Path 91:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void printStackTrace(PrintWriter
writer) //snmpbadconversionexception.java
      |...
      |120. writer.println("Original Reason");
```

[Table of Contents] [Description]

Path 92:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void printStackTrace(PrintWriter
writer) //snmpbadconversionexception.java
      |...
      |121. writer.println("");
```

[Table of Contents] [Description]

Path 93:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
412. private void sendToArchive(ResultSet eventsRS, int colCount) throws
SQLException //eventsarchiver.java
      |...
      |416. String colValue = eventsRS.getString(index);
```

[Table of Contents] [Description]

Path 94:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
104. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //citrixplugin.java
      |...
      |114. buffer.append((char) reader.read());
```

[Table of Contents] [Description]

Path 95:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
      ...
285. int rc = stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 96:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
      ...
408. int rc = stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 97:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
      ...
443. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 98:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
      ...
456. m_ifIndex = rset.getInt(ndx++);
```

[Table of Contents] [Description]

Path 99:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
      ...
462. m_lastGood = rset.getTimestamp(ndx++);
```

[Table of Contents] [Description]

Path 100:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
466. m_lastFail = rset.getTimestamp(ndx++);
```

[Table of Contents] [Description]

Path 101:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
470. m_qualifier = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 102:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
476. String str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 103:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
484. str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 104:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
492. str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 105:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      ...
462. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 106:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      ...
475. m_ifIndex = rset.getInt(ndx++);
```

[Table of Contents] [Description]

Path 107:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      ...
481. m_hostname = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 108:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      ...
487. String str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 109:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
```

```
...
495. m_status = rset.getInt(ndx++);
```

[Table of Contents] [Description]

Path 110:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
...
501. m_lastPoll = rset.getTimestamp(ndx++);
```

[Table of Contents] [Description]

Path 111:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
...
505. str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 112:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
904. DbIfServiceEntry[] getServices(Connection db) throws
SQLException //dbipinterfaceentry.java
...
909. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 113:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
904. DbIfServiceEntry[] getServices(Connection db) throws
SQLException //dbipinterfaceentry.java
...
913. int sid = rset.getInt(1);
```

[Table of Contents] [Description]

Path 114:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
726. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 115:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
739. m_createTime = rset.getTimestamp(ndx++);
```

[Table of Contents] [Description]

Path 116:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
743. m_parentId = rset.getInt(ndx++);
```

[Table of Contents] [Description]

Path 117:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
749. String str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 118:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
757. m_sysoid = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 119:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
763. m_sysname = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 120:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
769. m_sysdescr = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 121:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
775. m_syslocation = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 122:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
781. m_syscontact = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 123:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
787. m_label = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 124:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
793. str = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 125:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
801. m_nbName = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 126:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
807. m_nbDomainName = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 127:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
813. m_os = rset.getString(ndx++);
```

[Table of Contents] [Description]

Path 128:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
819. m_lastPoll = rset.getTimestamp(ndx++);
```

[Table of Contents] [Description]

Path 129:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1565. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 130:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1569. String saddr = rset.getString(1);
```

[Table of Contents] [Description]

Path 131:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1570. int ifIndex = rset.getInt(2);
```

[Table of Contents] [Description]

Path 132:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1622. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 133:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1626. String saddr = rset.getString(1);
```

[Table of Contents] [Description]

Path 134:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1627. int ifIndex = rset.getInt(2);
```

[Table of Contents] [Description]

Path 135:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1690. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 136:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1694. String saddr = rset.getString(1);
```

[Table of Contents] [Description]

Path 137:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1695. int ifIndex = rset.getInt(2);
```

[Table of Contents] [Description]

Path 138:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
1891. public void createAssetNodeEntry(Connection conn, int nodeid) throws
SQLException //dbnodeentry.java
```

```
...
1903. stmt.execute();
```

[Table of Contents] [Description]

Path 139:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
263. int rc = stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 140:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
200. private String retrieveIORText(String host, int port) throws
IOException //dominoiiopplugin.java
...
207. String line = dis.readLine();
```

[Table of Contents] [Description]

Path 141:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
124. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //imapplugin.java
...
134. String banner = lineRdr.readLine();
```

[Table of Contents] [Description]

Path 142:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
124. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //imapplugin.java
...
139. socket.getOutputStream().write(IMAP_LOGOUT_REQUEST.getBytes());
```

[Table of Contents] [Description]

Path 143:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
124. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //imapplugin.java
      |...
      |145. String response = lineRdr.readLine();
```

[Table of Contents] [Description]

Path 144:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
124. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //imapplugin.java
      |...
      |147. response = lineRdr.readLine();
```

[Table of Contents] [Description]

Path 145:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
290. public synchronized void saveCurrent() //collectdconfigfactory.java
      |...
      |302. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 146:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
128. protected void saveXML(String writerString) throws
IOException //destinationpathfactory.java
      |...
      |131. fileWriter.write(writerString);
```

[Table of Contents] [Description]

Path 147:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
346. public synchronized void saveCurrent() throws MarshalException, IOException,
ValidationException //eventconffactory.java
      |...
      |360. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 148:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
133. protected void saveXml(String data) throws IOException //groupfactory.java
     ...
136. fileWriter.write(data);
```

[Table of Contents] [Description]

Path 149:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
139. public synchronized void saveCurrent() throws IOException,
FileNotFoundException, MarshalException,
ValidationException //ksc_performancereportfactory.java
     ...
150. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 150:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
174. protected void saveXml(String xml) throws
IOException //notifdconfigfactory.java
     ...
177. fileWriter.write(xml);
```

[Table of Contents] [Description]

Path 151:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
134. protected void saveXML(String xmlString) throws
IOException //notificationfactory.java
     ...
137. fileWriter.write(xmlString);
```

[Table of Contents] [Description]

Path 152:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
     ...
841. parmMap.put(NotificationManager.PARAM_TEXT_MSG,
```

```
    resolutionPrefix+rs.getObject("textMsg"));
```

[Table of Contents] [Description]

Path 153:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
    ...
    842. parmMap.put(NotificationManager.PARAM_NUM_MSG, rs.getObject
    ("numericMsg"));
```

[Table of Contents] [Description]

Path 154:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
    ...
    843. parmMap.put(NotificationManager.PARAM_SUBJECT,
    resolutionPrefix+rs.getObject("subject"));
```

[Table of Contents] [Description]

Path 155:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
    ...
    844. parmMap.put(NotificationManager.PARAM_NODE, rs.getObject
    ("nodeID").toString());
```

[Table of Contents] [Description]

Path 156:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
    ...
    845. parmMap.put(NotificationManager.PARAM_INTERFACE, rs.getObject
    ("interfaceID"));
```

[Table of Contents] [Description]

Path 157:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
      ...
      846. parmMap.put(NotificationManager.PARAM_SERVICE, rs.getObject
      ("serviceName"));
```

[Table of Contents] [Description]

Path 158:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
      ...
      847. parmMap.put("noticeid", rs.getObject("notifyID").toString());
```

[Table of Contents] [Description]

Path 159:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
      ...
      848. parmMap.put("eventID", rs.getObject("eventID").toString());
```

[Table of Contents] [Description]

Path 160:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
840. public void processRow(ResultSet rs) throws
SQLException //notificationmanager.java
      ...
      849. parmMap.put("eventUEI", rs.getObject("eventUEI"));
```

[Table of Contents] [Description]

Path 161:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
837. public Map rebuildParamterMap(int notifId, final String resolutionPrefix)
throws Exception //notificationmanager.java
```

```
...
853. querier.execute(new Integer(notifId));
```

[Table of Contents] [Description]

Path 162:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
861. public void forEachUserNotification(int notifId, RowProcessor
rp) //notificationmanager.java
...
863. querier.execute(new Integer(notifId));
```

[Table of Contents] [Description]

Path 163:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
231. public synchronized void saveCurrent() throws MarshalException, IOException,
ValidationException //pollerconfigfactory.java
...
241. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 164:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
154. protected void saveXML(String xmlString) throws IOException, MarshalException,
ValidationException //polloutagesconfigfactory.java
...
158. fileWriter.write(xmlString);
```

[Table of Contents] [Description]

Path 165:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
186. public static synchronized void saveCurrent() throws MarshalException,
ValidationException, IOException //snmppeerfactory.java
...
194. fileWriter.write(marshalledConfig);
```

[Table of Contents] [Description]

Path 166:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
272. public synchronized void saveCurrent() //threshdconfigfactory.java
       ...
284. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 167:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
140. protected void saveXML(String writerString) throws
IOException //userfactory.java
       ...
143. fileWriter.write(writerString);
```

[Table of Contents] [Description]

Path 168:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
350. public synchronized void saveViews(Views views) throws IOException,
MarshalException, ValidationException //viewfactory.java
       ...
369. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 169:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
223. public static synchronized void saveCurrent() throws
Exception //vulnscandconfigfactory.java
       ...
231. fileWriter.write(stringWriter.toString());
```

[Table of Contents] [Description]

Path 170:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
309. public static boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //vulnscandconfigfactory.java
       ...
319. ResultSet rs = s.executeQuery();
```

[Table of Contents] [Description]

Path 171:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
334. public static int getInterfaceDbNodeId(Connection dbConn, InetAddress
ifAddress) throws SQLException //vulnscandconfigfactory.java
     ...
346. ResultSet rs = s.executeQuery();
```

[Table of Contents] [Description]

Path 172:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
334. public static int getInterfaceDbNodeId(Connection dbConn, InetAddress
ifAddress) throws SQLException //vulnscandconfigfactory.java
     ...
349. nodeid = rs.getInt(1);
```

[Table of Contents] [Description]

Path 173:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
206. void sendMessage(Message msg) throws IOException //client.java
     207. m_objsOut.writeObject(msg);
```

[Table of Contents] [Description]

Path 174:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
73. private void writeObject(ObjectOutputStream out) throws
IOException //message.java
     ...
77. out.writeInt(buf.length);
```

[Table of Contents] [Description]

Path 175:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
73. private void writeObject(ObjectOutputStream out) throws
IOException //message.java
     ...
77. out.writeInt(buf.length);
```

[Table of Contents] [Description]

Path 176:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
73. private void writeObject(ObjectOutputStream out) throws
IOException //message.java
      |...
      |78. out.write(buf);
```

[Table of Contents] [Description]

Path 177:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
81. private void readObject(ObjectInputStream in) throws IOException,
ClassNotFoundException //message.java
      |82. m_target = (InetAddress) in.readObject();
```

[Table of Contents] [Description]

Path 178:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
81. private void readObject(ObjectInputStream in) throws IOException,
ClassNotFoundException //message.java
      |...
      |84. byte[] buf = new byte[in.readInt()];
```

[Table of Contents] [Description]

Path 179:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
81. private void readObject(ObjectInputStream in) throws IOException,
ClassNotFoundException //message.java
      |...
      |85. in.read(buf, 0, buf.length);
```

[Table of Contents] [Description]

Path 180:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
      |...
```

```
289. ResultSet rset = m_getSvcIdStmt.executeQuery();
```

[Table of Contents] [Description]

Path 181:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
...
291. id = rset.getInt(1);
```

[Table of Contents] [Description]

Path 182:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
325. private String getHostName(String hostip) throws
SQLException //eventwriter.java
...
333. ResultSet rset = m_getHostNameStmt.executeQuery();
```

[Table of Contents] [Description]

Path 183:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
325. private String getHostName(String hostip) throws
SQLException //eventwriter.java
...
335. hostname = rset.getString(1);
```

[Table of Contents] [Description]

Path 184:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
...
373. ResultSet rs = m_getNextEventIdStmt.executeQuery();
```

[Table of Contents] [Description]

Path 185:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      375. eventID = rs.getInt(1);
```

[Table of Contents] [Description]

Path 186:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      598. m_eventInsStmt.executeUpdate();
```

[Table of Contents] [Description]

Path 187:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
285. public void update(String stmt, Object[] values) //mockdatabase.java
      |
      ...
      295. updater.execute(values);
```

[Table of Contents] [Description]

Path 188:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
324. public int countRows(String sql, Object[] values) //mockdatabase.java
      |
      ...
      326. querier.execute(values);
```

[Table of Contents] [Description]

Path 189:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
330. private boolean serviceDefined(String svcName) //mockdatabase.java
      |
      ...
      332. querier.execute(svcName);
```

[Table of Contents] [Description]

Path 190:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
     530. Outage outage = new Outage(rs.getInt("nodeId"), rs.getString("ipAddr"),
     rs.getInt("serviceId"));
```

[Table of Contents] [Description]

Path 191:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
     530. Outage outage = new Outage(rs.getInt("nodeId"), rs.getString("ipAddr"),
     rs.getInt("serviceId"));
```

[Table of Contents] [Description]

Path 192:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
     530. Outage outage = new Outage(rs.getInt("nodeId"), rs.getString("ipAddr"),
     rs.getInt("serviceId"));
```

[Table of Contents] [Description]

Path 193:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
     ...
     531. outage.setLostEvent(rs.getInt("svcLostEventID"), rs.getTimestamp
     ("ifLostService"));
```

[Table of Contents] [Description]

Path 194:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
     ...
     531. outage.setLostEvent(rs.getInt("svcLostEventID"), rs.getTimestamp
     ("ifLostService"));
```

[Table of Contents] [Description]

Path 195:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
      ...
532. boolean open = (rs.getObject("ifRegainedService") == null);
```

[Table of Contents] [Description]

Path 196:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
      ...
534. outage.setRegainedEvent(rs.getInt("svcRegainedEventID"), rs.getTimestamp
("ifRegainedService"));
```

[Table of Contents] [Description]

Path 197:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
529. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
      ...
534. outage.setRegainedEvent(rs.getInt("svcRegainedEventID"), rs.getTimestamp
("ifRegainedService"));
```

[Table of Contents] [Description]

Path 198:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
525. public Collection getOutages(String criteria, Object[]
values) //mockdatabase.java
      ...
539. loadExisting.execute(values);
```

[Table of Contents] [Description]

Path 199:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
591. public void processRow(ResultSet rs) throws SQLException //mockdatabase.java
592. notifyIds.add(rs.getObject(1));
```

[Table of Contents] [Description]

Path 200:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
588. public Collection findNoticesForEvent(Event event) //mockdatabase.java
       ...
595. loadExisting.execute(new Integer(event.getDbid()));
```

[Table of Contents] [Description]

Path 201:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
82. public void processRow(ResultSet rs) throws SQLException //mockdatabasetest.java
83. int nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 202:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
82. public void processRow(ResultSet rs) throws SQLException //mockdatabasetest.java
       ...
84. String label = rs.getString("nodeLabel");
```

[Table of Contents] [Description]

Path 203:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
80. public void testNodeQuery() //mockdatabasetest.java
       ...
91. querier.execute();
```

[Table of Contents] [Description]

Path 204:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public void processRow(ResultSet rs) throws SQLException //mockdatabasetest.java
98. int nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 205:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
97. public void processRow(ResultSet rs) throws SQLException //mockdatabasetest.java
      ...
      99. String ipAddr = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 206:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
95. public void testIFQuery() //mockdatabasetest.java
      ...
      106. querier.execute();
```

[Table of Contents] [Description]

Path 207:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
112. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      113. int nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 208:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
112. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      ...
      114. String ipAddr = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 209:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
112. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      ...
      115. int serviceId = rs.getInt("serviceId");
```

[Table of Contents] [Description]

Path 210:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
112. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      |...
      |116. String serviceName = rs.getString("serviceName");
```

[Table of Contents] [Description]

Path 211:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
112. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      |...
      |117. String status = rs.getString("status");
```

[Table of Contents] [Description]

Path 212:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
110. public void testServiceQuery() //mockdatabasetest.java
      |...
      |127. querier.execute();
```

[Table of Contents] [Description]

Path 213:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
147. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      |148. int nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 214:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
147. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      |...
      |149. String ipAddr = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 215:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
147. public void processRow(ResultSet rs) throws
SQLException //mockdatabasetest.java
      ...
150. int serviceId = rs.getInt("serviceId");
```

[Table of Contents] [Description]

Path 216:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
138. public void testOutage() //mockdatabasetest.java
      ...
156. querier.execute();
```

[Table of Contents] [Description]

Path 217:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
232. public void processRow(ResultSet rs) throws
SQLException //broadcasteventprocessor.java
      233. String userID = rs.getString("userID");
```

[Table of Contents] [Description]

Path 218:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
232. public void processRow(ResultSet rs) throws
SQLException //broadcasteventprocessor.java
      ...
234. String contactInfo = rs.getString("contactinfo");
```

[Table of Contents] [Description]

Path 219:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
232. public void processRow(ResultSet rs) throws
SQLException //broadcasteventprocessor.java
      ...
235. String autoNotifyChar = rs.getString("autonotify");
```

[Table of Contents] [Description]

Path 220:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
232. public void processRow(ResultSet rs) throws
SQLException //broadcasteventprocessor.java
      ...
239. String cmd = rs.getString("media");
```

[Table of Contents] [Description]

Path 221:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
756. public void processRow(ResultSet rs) throws SQLException //notifdtest.java
      ...
758. cmdList.add(rs.getString("userID"));
```

[Table of Contents] [Description]

Path 222:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
756. public void processRow(ResultSet rs) throws SQLException //notifdtest.java
      ...
759. cmdList.add(rs.getString("media"));
```

[Table of Contents] [Description]

Path 223:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
95. public int closeOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
      ...
106. int count = outageUpdater.executeUpdate();
```

[Table of Contents] [Description]

Path 224:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
      ...
```

```
134. ResultSet activeSvcsRS = activeSvcsStmt.executeQuery();
```

[Table of Contents] [Description]

Path 225:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
...
137. BasicService svc = getNetwork().getService(this, activeSvcsRS.getLong
(1));
```

[Table of Contents] [Description]

Path 226:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. public long getNextOutageId(Connection dbConn) throws
SQLException //basicnetwork.java
...
109. ResultSet seqRS = getNextOutageIdStmt.executeQuery();
```

[Table of Contents] [Description]

Path 227:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
101. public long getNextOutageId(Connection dbConn) throws
SQLException //basicnetwork.java
...
111. outageID = seqRS.getLong(1);
```

[Table of Contents] [Description]

Path 228:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
91. public int closeOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
...
102. int count = outageUpdater.executeUpdate();
```

[Table of Contents] [Description]

Path 229:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
    ...
129. ResultSet activeSvcsRS = activeSvcsStmt.executeQuery();
```

[Table of Contents] [Description]

Path 230:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
    ...
131. BasicService svc = getNetwork().getService(getNodeId(),
activeSvcsRS.getString(1), activeSvcsRS.getLong(2));
```

[Table of Contents] [Description]

Path 231:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
    ...
131. BasicService svc = getNetwork().getService(getNodeId(),
activeSvcsRS.getString(1), activeSvcsRS.getLong(2));
```

[Table of Contents] [Description]

Path 232:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
102. public void closeOutage(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicservice.java
    ...
114. outageUpdater.executeUpdate();
```

[Table of Contents] [Description]

Path 233:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
127. public boolean openOutage(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicservice.java
    ...
149. newOutageWriter.executeUpdate();
```

[Table of Contents] [Description]

Path 234:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
57. static public int executeCountQuery(PreparedStatement openStmt) throws
SQLException //dbutil.java
      ...
59. ResultSet rs = openStmt.executeQuery();
```

[Table of Contents] [Description]

Path 235:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
57. static public int executeCountQuery(PreparedStatement openStmt) throws
SQLException //dbutil.java
      ...
61. numOpenOutages = rs.getInt(1);
```

[Table of Contents] [Description]

Path 236:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
123. private void buildServiceTableMap(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
128. ResultSet rset = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 237:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
123. private void buildServiceTableMap(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
130. long svcid = rset.getLong(1);
```

[Table of Contents] [Description]

Path 238:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
123. private void buildServiceTableMap(java.sql.Connection dbConn) throws
```

```
SQLException //outagemanager.java
      ...
      131. String svcname = rset.getString(2);
```

[Table of Contents] [Description]

Path 239:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
      162. int num = closeForServicesStmt.executeUpdate();
```

[Table of Contents] [Description]

Path 240:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
      168. num = closeForInterfacesStmt.executeUpdate();
```

[Table of Contents] [Description]

Path 241:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      144. assertEquals(m_svc.getNodeId(), rs.getInt("nodeId"));
```

[Table of Contents] [Description]

Path 242:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      ...
      145. assertEquals(m_svc.getIpAddr(), rs.getString("ipAddr"));
```

[Table of Contents] [Description]

Path 243:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      |...
      |146. assertEquals(m_svc.getId(), rs.getInt("serviceId"));
```

[Table of Contents] [Description]

Path 244:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      |...
      |147. assertEquals(m_lostSvcEvent.getDbid(), rs.getInt("svcLostEventId"));
```

[Table of Contents] [Description]

Path 245:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      |...
      |148. assertEquals(m_lostSvcTime, rs.getTimestamp("ifLostService"));
```

[Table of Contents] [Description]

Path 246:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      |...
      |149. assertEquals(getRegainedEventId(), rs.getObject("svcRegainedEventId"));
```

[Table of Contents] [Description]

Path 247:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
143. public void processRow(ResultSet rs) throws SQLException //outagetest.java
      |...
      |150. assertEquals(m_regainedSvcTime, rs.getTimestamp("ifRegainedService"));
```

[Table of Contents] [Description]

Path 248:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
333. public void visitService(MockService svc) //outagetest.java
      |...
      |335. checker.execute(new Integer(svc.getNodeId()), svc.getIpAddr(), new
      |Integer(svc.getId()));
```

[Table of Contents] [Description]

Path 249:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
347. public List getInterfacesWithService(String svcName) throws
SQLException //defaultquerymanager.java
      |...
      |357. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 250:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
347. public List getInterfacesWithService(String svcName) throws
SQLException //defaultquerymanager.java
      |...
      |365. IfKey key = new IfKey(rs.getInt(1), rs.getString(2));
```

[Table of Contents] [Description]

Path 251:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
347. public List getInterfacesWithService(String svcName) throws
SQLException //defaultquerymanager.java
      |...
      |365. IfKey key = new IfKey(rs.getInt(1), rs.getString(2));
```

[Table of Contents] [Description]

Path 252:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
169. private void closeOutagesForUnmanagedServices() //poller.java
      |...
      |174. svcUpdater.execute(closeTime);
```

[Table of Contents] [Description]

Path 253:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
169. private void closeOutagesForUnmanagedServices() //poller.java
       ...
178. ifUpdater.execute(closeTime);
```

[Table of Contents] [Description]

Path 254:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
184. public void closeOutagesForNode(Date closeDate, int eventId, int
nodeId) //poller.java
       ...
188. svcUpdater.execute(closeTime, new Integer(eventId), new Integer(nodeId));
```

[Table of Contents] [Description]

Path 255:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
191. public void closeOutagesForInterface(Date closeDate, int eventId, int nodeId,
String ipAddr) //poller.java
       ...
195. svcUpdater.execute(closeTime, new Integer(eventId), new Integer(nodeId),
ipAddr);
```

[Table of Contents] [Description]

Path 256:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
198. public void closeOutagesForService(Date closeDate, int eventId, int nodeId,
String ipAddr, String serviceName) //poller.java
       ...
202. svcUpdater.execute(closeTime, new Integer(eventId), new Integer(nodeId),
ipAddr, serviceName);
```

[Table of Contents] [Description]

Path 257:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
       404. scheduleService(rs.getInt("nodeId"), rs.getString("ipAddr"), rs.getString
("serviceName"),
```

[Table of Contents] [Description]

Path 258:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
     404. scheduleService(rs.getInt("nodeId"), rs.getString("ipAddr"), rs.getString
     ("serviceName"),
```

[Table of Contents] [Description]

Path 259:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
     404. scheduleService(rs.getInt("nodeId"), rs.getString("ipAddr"), rs.getString
     ("serviceName"),
```

[Table of Contents] [Description]

Path 260:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
     ...
     405. (Number)rs.getObject("svcLostEventId"), rs.getTimestamp("ifLostService"),
```

[Table of Contents] [Description]

Path 261:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
     ...
     405. (Number)rs.getObject("svcLostEventId"), rs.getTimestamp("ifLostService"),
```

[Table of Contents] [Description]

Path 262:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
403. public void processRow(ResultSet rs) throws SQLException //poller.java
     ...
     406. rs.getString("svcLostEventUei"));
```

[Table of Contents] [Description]

Path 263:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
383. private int scheduleMatchingServices(String criteria) //poller.java
       ...
409. querier.execute();
```

[Table of Contents] [Description]

Path 264:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
       858. assertEquals(m_svc.getNodeId(), rs.getInt("nodeId"));
```

[Table of Contents] [Description]

Path 265:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
       ...
859. assertEquals(m_svc.getIpAddr(), rs.getString("ipAddr"));
```

[Table of Contents] [Description]

Path 266:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
       ...
860. assertEquals(m_svc.getId(), rs.getInt("serviceId"));
```

[Table of Contents] [Description]

Path 267:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
       ...
861. assertEquals(m_lostSvcEvent.getDbid(), rs.getInt("svcLostEventId"));
```

[Table of Contents] [Description]

Path 268:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
      |...
      |862. assertEquals(m_lostSvcTime, rs.getTimestamp("ifLostService"));
```

[Table of Contents] [Description]

Path 269:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
      |...
      |863. assertEquals(getRegainedEventId(), rs.getObject("svcRegainedEventId"));
```

[Table of Contents] [Description]

Path 270:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
857. public void processRow(ResultSet rs) throws SQLException //pollertest.java
      |...
      |864. assertEquals(m_regainedSvcTime, rs.getTimestamp("ifRegainedService"));
```

[Table of Contents] [Description]

Path 271:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
252. private String retrieveIORText(String host, int port) throws
IOException //dominoiiopmonitor.java
      |...
      |259. String line = dis.readLine();
```

[Table of Contents] [Description]

Path 272:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
27. public static NrpePacket executeQuery(String host, int port, String
buffer, //checknrpe.java
      |...
      |34. o.write(b);
```

[Table of Contents] [Description]

Path 273:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
39. public static NrpePacket sendPacket(short type, short resultCode, String buffer)
throws Exception //checknrpe.java
      ...
46. o.write(b);
```

[Table of Contents] [Description]

Path 274:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
78. public static NrpePacket receivePacket(InputStream i, int padding) throws
NrpeException, IOException //nrpepacket.java
      ...
85. for (k = 0; (j = i.read()) != -1; k++) {
```

[Table of Contents] [Description]

Path 275:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
      288. int nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 276:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
      ...
289. String ipAddr = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 277:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
      ...
290. String serviceName = rs.getString("serviceName");
```

[Table of Contents] [Description]

Path 278:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
269. static private PollableNetwork createPollableNetwork(final DbConnectionFactory
db, final ScheduleTimer scheduler, final PollerConfig pollerConfig, final
PollOutagesConfig pollOutageConfig, PollContext pollContext) throws
UnknownHostException //pollablestest.java
      ...
355. querier.execute();
```

[Table of Contents] [Description]

Path 279:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
379. public InputStream createGraph(String command, File workDir) throws
IOException, RrdException //jnirrdstrategy.java
      ...
394. String line = err.readLine();
```

[Table of Contents] [Description]

Path 280:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
379. public InputStream createGraph(String command, File workDir) throws
IOException, RrdException //jnirrdstrategy.java
      ...
400. line = err.readLine();
```

[Table of Contents] [Description]

Path 281:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
173. public static void main(final String[] args) throws
Exception //rrdstresser.java
      ...
183. while ((line = rdr.readLine()) != null) {
```

[Table of Contents] [Description]

Path 282:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
127. public void printStackTrace(java.io.PrintWriter
pw) //snmptraphelperexception.java
     |...
     |134. pw.println(this);
```

[Table of Contents] [Description]

Path 283:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
     |...
     |109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 284:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
     |...
     |116. if (rs.getString("snmpifdescr") != null) {
```

[Table of Contents] [Description]

Path 285:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
     |...
     |117. if (Pattern.matches(".*-cef.*", rs.getString("snmpifdescr")))
```

[Table of Contents] [Description]

Path 286:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
     |...
     |121. if ((AlphaNumeric.parseAndReplace(rs.getString("snmpifname"), '_').equals
     (desc)) || (AlphaNumeric.parseAndReplace(rs.getString("snmpifdescr"),
     '_').equals(desc))) {
```

[Table of Contents] [Description]

Path 287:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
121. if ((AlphaNumeric.parseAndReplace(rs.getString("snmpifname"), '_').equals
(desc)) || (AlphaNumeric.parseAndReplace(rs.getString("snmpifdescr"),
'_').equals(desc))) {
```

[Table of Contents] [Description]

Path 288:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
125. if (mac == null || mac.equals(rs.getString("snmpphysaddr"))) {
```

[Table of Contents] [Description]

Path 289:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
127. for (int i = 1; i <= rs.getMetaData().getColumnCount(); i++) {
```

[Table of Contents] [Description]

Path 290:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
```

[Table of Contents] [Description]

Path 291:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
```

```
String ifLabel) throws SQLException //iflabel.java
      ...
129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
```

[Table of Contents] [Description]

Path 292:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
      ...
636. IPv4Address ipv4Addr = new IPv4Address(rs.getString(1));
```

[Table of Contents] [Description]

Path 293:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
      ...
638. String hostName = rs.getString(2);
```

[Table of Contents] [Description]

Path 294:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
61. protected void executeStmt(PreparedStatement stmt) throws
SQLException //querier.java
      ...
63. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 295:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
49. public void processRow(ResultSet rs) throws
SQLException //singleresultquerier.java
      50. m_result = rs.getObject(1);
```

[Table of Contents] [Description]

Path 296:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
51. void executeStmt(PreparedStatement stmt) throws SQLException //updater.java
      52. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 297:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
      ...
      283. dataOut.writeShort(m_reqID);
```

[Table of Contents] [Description]

Path 298:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
      ...
      284. dataOut.writeShort((0 << SHIFT_QUERY) | (OPCODE_QUERY << SHIFT_OPCODE) |
      (1 << SHIFT_RECURSE_PLEASE));
```

[Table of Contents] [Description]

Path 299:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
      ...
      286. dataOut.writeShort(1); // # queries
```

[Table of Contents] [Description]

Path 300:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
      ...
      287. dataOut.writeShort(0); // # answers
```

[Table of Contents] [Description]

Path 301:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
       ...
298. dataOut.writeByte(0);
```

[Table of Contents] [Description]

Path 302:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
       ...
289. dataOut.writeShort(0); // # additional
```

[Table of Contents] [Description]

Path 303:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
       ...
294. dataOut.writeByte(label.length());
```

[Table of Contents] [Description]

Path 304:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
       ...
295. dataOut.writeBytes(label);
```

[Table of Contents] [Description]

Path 305:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
       ...
298. dataOut.writeByte(0);
```

[Table of Contents] [Description]

Path 306:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
     |...
     299. dataOut.writeShort(TYPE_ADDR);
```

[Table of Contents] [Description]

Path 307:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
279. public byte[] buildRequest() throws IOException //dnsaddressrequest.java
     |...
     300. dataOut.writeShort(CLASS_IN);
```

[Table of Contents] [Description]

Path 308:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
     |...
     324. int id = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 309:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
     |...
     331. int flags = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 310:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
     |...
     344. int rtype = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 311:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |...
      |335. int numAnswers = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 312:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |...
      |336. int numAuthorities = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 313:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |...
      |337. int numAdditional = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 314:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |...
      |343. String rname = dnsIn.readDomainName();
```

[Table of Contents] [Description]

Path 315:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |...
```

```
344. int rtype = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 316:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
    ...
345. int rclass = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 317:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
384. public void verifyResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
    ...
389. int id = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 318:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
384. public void verifyResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
    ...
396. int flags = dnsIn.readShort();
```

[Table of Contents] [Description]

Path 319:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public DNSAddressRR(String name, int type, int clas, long ttl, DNSInputStream
dnsIn) throws IOException //dnsaddressrr.java
    ...
165. ipAddress[i] = dnsIn.readByte();
```

[Table of Contents] [Description]

Path 320:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
166. public String readDomainName() throws IOException //dnsinputstream.java
       ...
221. return dnsIn.readDomainName();
```

[Table of Contents] [Description]

Path 321:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
263. protected void mailFileToUser(String mailScript, String filename, String
emailAddr) throws IOException //reportmailer.java
       ...
274. if (err.ready()) {
```

[Table of Contents] [Description]

Path 322:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
573. public static void streamToStream(InputStream in, OutputStream out, int
bufferSize) throws IOException //util.java
       ...
577. while ((length = in.read(b)) != -1) {
```

[Table of Contents] [Description]

Path 323:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
573. public static void streamToStream(InputStream in, OutputStream out, int
bufferSize) throws IOException //util.java
       ...
578. out.write(b, 0, length);
```

[Table of Contents] [Description]

Path 324:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
       ...
255. update.executeUpdate(query.toString());
```

[Table of Contents] [Description]

Path 325:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
    ...
275. update.executeUpdate(query.toString());
```

[Table of Contents] [Description]

Path 326:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
    ...
255. update.executeUpdate(query.toString());
```

[Table of Contents] [Description]

Path 327:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
    ...
275. update.executeUpdate(query.toString());
```

[Table of Contents] [Description]

Path 328:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
    ...
81. log.debug("doPost: firstIPAddress=" + firstIPAddress + ", "
```

[Table of Contents] [Description]

Path 329:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
277. Object element = new Integer(rs.getInt("nodeID"));
```

[Table of Contents] [Description]

Path 330:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
280. asset.setCategory(rs.getString("category"));
```

[Table of Contents] [Description]

Path 331:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
281. asset.setManufacturer(rs.getString("manufacturer"));
```

[Table of Contents] [Description]

Path 332:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
302. asset.setRoom(rs.getString("room"));
```

[Table of Contents] [Description]

Path 333:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
283. asset.setModelNumber(rs.getString("modelNumber"));
```

[Table of Contents] [Description]

Path 334:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
```

```
      ...
284.  asset.setSerialNumber(rs.getString("serialNumber"));
```

[Table of Contents] [Description]

Path 335:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
285.  asset.setDescription(rs.getString("description"));
```

[Table of Contents] [Description]

Path 336:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
296.  asset.setAddress2(rs.getString("address2"));
```

[Table of Contents] [Description]

Path 337:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
287.  asset.setAssetNumber(rs.getString("assetNumber"));
```

[Table of Contents] [Description]

Path 338:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
308.  asset.setMaintContract(rs.getString("maintContract"));
```

[Table of Contents] [Description]

Path 339:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |...
      |299. asset.setZip(rs.getString("zip"));
```

[Table of Contents] [Description]

Path 340:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |...
      |290. asset.setSlot(rs.getString("slot"));
```

[Table of Contents] [Description]

Path 341:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |...
      |291. asset.setPort(rs.getString("port"));
```

[Table of Contents] [Description]

Path 342:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |...
      |292. asset.setRegion(rs.getString("region"));
```

[Table of Contents] [Description]

Path 343:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |...
      |293. asset.setDivision(rs.getString("division"));
```

[Table of Contents] [Description]

Path 344:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
294. asset.setDepartment(rs.getString("department"));
```

[Table of Contents] [Description]

Path 345:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
295. asset.setAddress1(rs.getString("address1"));
```

[Table of Contents] [Description]

Path 346:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
296. asset.setAddress2(rs.getString("address2"));
```

[Table of Contents] [Description]

Path 347:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
297. asset.setCity(rs.getString("city"));
```

[Table of Contents] [Description]

Path 348:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
```

```
      ...
298.  asset.setState(rs.getString("state"));
```

[Table of Contents] [Description]

Path 349:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270.  protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
299.  asset.setZip(rs.getString("zip"));
```

[Table of Contents] [Description]

Path 350:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270.  protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
300.  asset.setBuilding(rs.getString("building"));
```

[Table of Contents] [Description]

Path 351:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270.  protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
301.  asset.setFloor(rs.getString("floor"));
```

[Table of Contents] [Description]

Path 352:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270.  protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
302.  asset.setRoom(rs.getString("room"));
```

[Table of Contents] [Description]

Path 353:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      303. asset.setVendorPhone(rs.getString("vendorPhone"));
```

[Table of Contents] [Description]

Path 354:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      304. asset.setVendorFax(rs.getString("vendorFax"));
```

[Table of Contents] [Description]

Path 355:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      305. asset.setUserLastModified(rs.getString("userLastModified"));
```

[Table of Contents] [Description]

Path 356:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      306. asset.setLease(rs.getString("lease"));
```

[Table of Contents] [Description]

Path 357:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      307. asset.setSupportPhone(rs.getString("supportPhone"));
```

[Table of Contents] [Description]

Path 358:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      308. asset.setMaintContract(rs.getString("maintContract"));
```

[Table of Contents] [Description]

Path 359:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      309. asset.setDateInstalled(rs.getString("dateInstalled"));
```

[Table of Contents] [Description]

Path 360:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      310. asset.setLeaseExpires(rs.getString("leaseExpires"));
```

[Table of Contents] [Description]

Path 361:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
      ...
      311. asset.setMaintContractExpires(rs.getString("maintContractExpires"));
```

[Table of Contents] [Description]

Path 362:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      |
```

```
...
312. asset.setVendorAssetNumber(rs.getString("vendorAssetNumber"));
```

[Table of Contents] [Description]

Path 363:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
313. asset.setDisplayCategory(rs.getString("displayCategory"));
```

[Table of Contents] [Description]

Path 364:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
314. asset.setNotifyCategory(rs.getString("notifyCategory"));
```

[Table of Contents] [Description]

Path 365:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
```

[Table of Contents] [Description]

Path 366:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```

[Table of Contents] [Description]

Path 367:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
317. asset.setComments(rs.getString("comment"));
```

[Table of Contents] [Description]

Path 368:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
    ...
319. element = rs.getTimestamp("lastModifiedDate");
```

[Table of Contents] [Description]

Path 369:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
    ...
114. out.println(buffer.toString());
```

[Table of Contents] [Description]

Path 370:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
    ...
206. out.println(buffer.toString());
```

[Table of Contents] [Description]

Path 371:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
234. out.println("<table width=\"100%\" border=\"1\" cellspacing=\"0\" " +
"cellpadding=\"2\" bordercolor=\"black\" " + "bgcolor=\"#cccccc\">");
```

[Table of Contents] [Description]

Path 372:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
242. out.println("<tr bgcolor=\"#999999\">");
```

[Table of Contents] [Description]

Path 373:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
243. out.println("<td width=\"50%\"><b>" + sectionName + "</b></td>");
```

[Table of Contents] [Description]

Path 374:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
244. out.println("<td width=\"20%\" align=\"right\">" +
"<b>Outages</b></td>");
```

[Table of Contents] [Description]

Path 375:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
245. out.println("<td width=\"30%\" align=\"right\">" + "<b>24hr
Avail</b></td>");
```

[Table of Contents] [Description]

Path 376:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
246. out.println("</tr>");
```

[Table of Contents] [Description]

Path 377:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
274. out.println("<tr>");
```

[Table of Contents] [Description]

Path 378:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
276. out.println("<td><a href=\"rtc/category.jsp?category=" +
URLEncoder.encode(response.encodeURL(categoryName), "UTF-8") + "\" title=\"" +
title + "\">" + categoryName + "</a></td>");
```

[Table of Contents] [Description]

Path 379:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
277. out.println("<td bgcolor=\"" + outageColor + "\" align=\"right\"
title=\"Updated: " + lastUpdated + "\">" + outageText + "</td>");
```

[Table of Contents] [Description]

Path 380:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
278. out.println("<td bgcolor=\"" + availColor + "\" align=\"right\"
title=\"Updated: " + lastUpdated + "\">" + availText + "</td>");
```

[Table of Contents] [Description]

Path 381:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
279. out.println("<!-- Last updated " + lastUpdated + " -->");
```

[Table of Contents] [Description]

Path 382:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
280. out.println("<!-- Epoch time: " + lastUpdatedTime + " -->");
```

[Table of Contents] [Description]

Path 383:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
282. out.println("</tr>");
```

[Table of Contents] [Description]

Path 384:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
286. out.println("<tr bgcolor=\"#999999\">");
```

[Table of Contents] [Description]

Path 385:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
```

```
IOException, MarshalException, ValidationException //categorylist.java
    ...
288. out.println("<td colspan=\"3\"><font color=\"#bb1111\">" + "OpenNMS
Disconnect -- is the OpenNMS daemon " + "running?<br/>Last update: " +
(earliestUpdate > 0 ? new Date(earliestUpdate).toString() : "one or more
categories have never been updated.") + "</font></td>");
```

[Table of Contents] [Description]

Path 386:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
290. out.println("<td colspan=\"3\">Percentage over last " + "24 hours</td>");
```

[Table of Contents] [Description]

Path 387:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
293. out.println("</tr>");
```

[Table of Contents] [Description]

Path 388:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
294. out.println("</table>");
```

[Table of Contents] [Description]

Path 389:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
62. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcdebugservlet.java
    ...
75. out.write("No data exists for this category. Please check your spelling of
the category name.");
```

[Table of Contents] [Description]

Path 390:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcpostservlet.java
      ...
128. out.println("Category data parsed successfully.");
```

[Table of Contents] [Description]

Path 391:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
614. node.m_nodeId = rs.getInt("nodeId");
```

[Table of Contents] [Description]

Path 392:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
615. node.m_dpname = rs.getString("dpName");
```

[Table of Contents] [Description]

Path 393:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
617. element = rs.getTimestamp("nodeCreateTime");
```

[Table of Contents] [Description]

Path 394:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
621. element = new Integer(rs.getInt("nodeParentID"));
```

[Table of Contents] [Description]

Path 395:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
626. element = rs.getString("nodeType");
```

[Table of Contents] [Description]

Path 396:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
```

[Table of Contents] [Description]

Path 397:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
```

[Table of Contents] [Description]

Path 398:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
```

[Table of Contents] [Description]

Path 399:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
```

[Table of Contents] [Description]

Path 400:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
```

[Table of Contents] [Description]

Path 401:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
      ...
      636. node.m_label = rs.getString("nodelabel");
```

[Table of Contents] [Description]

Path 402:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
      ...
      637. node.m_operatingSystem = rs.getString("operatingsystem");
```

[Table of Contents] [Description]

Path 403:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
      ...
      664. intf.m_nodeId = rs.getInt("nodeid");
```

[Table of Contents] [Description]

Path 404:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
675. element = rs.getTimestamp("ipLastCapsdPoll");
```

[Table of Contents] [Description]

Path 405:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
666. intf.m_ipStatus = rs.getInt("ipStatus");
```

[Table of Contents] [Description]

Path 406:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
667. intf.m_ipHostName = rs.getString("ipHostname");
```

[Table of Contents] [Description]

Path 407:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
668. intf.m_ipAddr = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 408:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
670. element = rs.getString("isManaged");
```

[Table of Contents] [Description]

Path 409:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
...
675. element = rs.getTimestamp("ipLastCapsdPoll");
```

[Table of Contents] [Description]

Path 410:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
...
702. ResultSet rs = pstmt.executeQuery();
```

[Table of Contents] [Description]

Path 411:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
...
705. intfs[i].m_snmpIfIndex = rs.getInt("snmpifindex");
```

[Table of Contents] [Description]

Path 412:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
...
706. intfs[i].m_snmpIpAdEntNetMask = rs.getString("snmpIpAdEntNetMask");
```

[Table of Contents] [Description]

Path 413:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
707. intfs[i].m_snmpPhysAddr = rs.getString("snmpPhysAddr");
```

[Table of Contents] [Description]

Path 414:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
708. intfs[i].m_snmpIfDescr = rs.getString("snmpIfDescr");
```

[Table of Contents] [Description]

Path 415:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
709. intfs[i].m_snmpIfName = rs.getString("snmpIfName");
```

[Table of Contents] [Description]

Path 416:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
710. intfs[i].m_snmpIfType = rs.getInt("snmpIfType");
```

[Table of Contents] [Description]

Path 417:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
711. intfs[i].m_snmpIfOperStatus = rs.getInt("snmpIfOperStatus");
```

[Table of Contents] [Description]

Path 418:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
712. intfs[i].m_snmpIfSpeed = rs.getLong("snmpIfSpeed");
```

[Table of Contents] [Description]

Path 419:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
713. intfs[i].m_snmpIfAdminStatus = rs.getInt("snmpIfAdminStatus");
```

[Table of Contents] [Description]

Path 420:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
714. intfs[i].m_snmpIfAlias = rs.getString("snmpIfAlias");
```

[Table of Contents] [Description]

Path 421:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
     ...
725. rs = pstmt.executeQuery();
```

[Table of Contents] [Description]

Path 422:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
691. protected static void augmentInterfacesWithSnmpData(Interface[] intfs,
Connection conn) throws SQLException //networkelementfactory.java
```

```
...
728. intfs[i].m_isSnmpPrimary = rs.getString("issnmpprimary");
```

[Table of Contents] [Description]

Path 423:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
746. service.m_nodeId = rs.getInt("nodeid");
```

[Table of Contents] [Description]

Path 424:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
747. service.m_ifIndex = rs.getInt("ifindex");
```

[Table of Contents] [Description]

Path 425:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
748. service.m_ipAddr = rs.getString("ipaddr");
```

[Table of Contents] [Description]

Path 426:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
750. element = rs.getTimestamp("lastgood");
```

[Table of Contents] [Description]

Path 427:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      |...
      |754. service.m_serviceId = rs.getInt("serviceid");
```

[Table of Contents] [Description]

Path 428:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      |...
      |755. service.m_serviceName = rs.getString("servicename");
```

[Table of Contents] [Description]

Path 429:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      |...
      |757. element = rs.getTimestamp("lastfail");
```

[Table of Contents] [Description]

Path 430:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      |...
      |761. service.m_notify = rs.getString("notify");
```

[Table of Contents] [Description]

Path 431:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      |...
      |763. element = rs.getString("status");
```

[Table of Contents] [Description]

Path 432:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |991. Object element = new Integer(rs.getInt("eventID"));
```

[Table of Contents] [Description]

Path 433:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |994. element = rs.getString("eventUei");
```

[Table of Contents] [Description]

Path 434:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |997. element = rs.getString("eventSnmp");
```

[Table of Contents] [Description]

Path 435:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1000. element = rs.getTimestamp("eventTime");
```

[Table of Contents] [Description]

Path 436:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |
```

```
...
1003. element = rs.getString("eventHost");
```

[Table of Contents] [Description]

Path 437:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1006. element = rs.getString("eventSnmpHost");
```

[Table of Contents] [Description]

Path 438:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1009. element = rs.getString("eventDpName");
```

[Table of Contents] [Description]

Path 439:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1012. element = rs.getString("eventParms");
```

[Table of Contents] [Description]

Path 440:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1016. element = rs.getObject("nodeID");
```

[Table of Contents] [Description]

Path 441:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1023. element = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 442:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1026. element = rs.getObject("serviceID");
```

[Table of Contents] [Description]

Path 443:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1029. element = rs.getString("nodeLabel");
```

[Table of Contents] [Description]

Path 444:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1032. element = rs.getString("serviceName");
```

[Table of Contents] [Description]

Path 445:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      |1035. element = rs.getTimestamp("eventCreateTime");
```

[Table of Contents] [Description]

Path 446:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      1038. element = rs.getString("eventDescr");
```

[Table of Contents] [Description]

Path 447:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      1041. element = rs.getString("eventLoggroup");
```

[Table of Contents] [Description]

Path 448:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      1044. element = rs.getString("eventLogmsg");
```

[Table of Contents] [Description]

Path 449:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      1047. element = new Integer(rs.getInt("eventSeverity"));
```

[Table of Contents] [Description]

Path 450:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |
```

```
...
1050. element = rs.getString("eventOperInstruct");
```

[Table of Contents] [Description]

Path 451:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1053. element = rs.getString("eventAutoAction");
```

[Table of Contents] [Description]

Path 452:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1056. element = rs.getString("eventOperAction");
```

[Table of Contents] [Description]

Path 453:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1059. element = rs.getString("eventOperActionMenuText");
```

[Table of Contents] [Description]

Path 454:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1062. element = rs.getString("eventNotification");
```

[Table of Contents] [Description]

Path 455:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1065. element = rs.getString("eventTticket");
```

[Table of Contents] [Description]

Path 456:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1068. element = rs.getObject("eventTticketState");
```

[Table of Contents] [Description]

Path 457:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1071. element = rs.getString("eventForward");
```

[Table of Contents] [Description]

Path 458:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1074. element = rs.getString("eventMouseOverText");
```

[Table of Contents] [Description]

Path 459:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1077. element = rs.getString("eventAckUser");
```

[Table of Contents] [Description]

Path 460:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      |...
      1080. element = rs.getTimestamp("eventAckTime");
```

[Table of Contents] [Description]

Path 461:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
106. public void setContainer(Container container) //log4jlogger.java
      |...
      108. Container oldContainer = this.container;
```

[Table of Contents] [Description]

Path 462:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
106. public void setContainer(Container container) //log4jlogger.java
      |...
      109. this.container = container;
```

[Table of Contents] [Description]

Path 463:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
106. public void setContainer(Container container) //log4jlogger.java
      |...
      110. support.firePropertyChange("container", oldContainer, this.container);
```

[Table of Contents] [Description]

Path 464:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
129. public int getVerbosity() //log4jlogger.java
      |...
      131. return (this.verbosity);
```

[Table of Contents] [Description]

Path 465:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
142. public void setVerbosity(int verbosity) //log4jlogger.java
       ...
144. this.verbosity = verbosity;
```

[Table of Contents] [Description]

Path 466:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       ...
158. this.verbosity = FATAL;
```

[Table of Contents] [Description]

Path 467:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       ...
160. this.verbosity = ERROR;
```

[Table of Contents] [Description]

Path 468:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       ...
162. this.verbosity = WARNING;
```

[Table of Contents] [Description]

Path 469:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       ...
164. this.verbosity = INFORMATION;
```

[Table of Contents] [Description]

Path 470:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       ...
166. this.verbosity = DEBUG;
```

[Table of Contents] [Description]

Path 471:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
171. public void setHomeDir(String homeDir) //log4jlogger.java
       ...
173. PropertyConfigurator.configure(homeDir + Log4JLogger.propFilename);
```

[Table of Contents] [Description]

Path 472:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
171. public void setHomeDir(String homeDir) //log4jlogger.java
       ...
174. this.logger = Category.getInstance("OpenNMS.WEB");
```

[Table of Contents] [Description]

Path 473:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
200. public void log(String msg) //log4jlogger.java
       ...
202. this.logger.debug(msg);
```

[Table of Contents] [Description]

Path 474:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
200. public void log(String msg) //log4jlogger.java
       ...
204. this.logger.info(msg);
```

[Table of Contents] [Description]

Path 475:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
220. public void log(Exception exception, String msg) //log4jlogger.java
     |221. this.log(msg, exception);
```

[Table of Contents] [Description]

Path 476:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
235. public void log(String msg, Throwable throwable) //log4jlogger.java
     |...
     |237. this.logger.warn(msg, throwable);
```

[Table of Contents] [Description]

Path 477:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
235. public void log(String msg, Throwable throwable) //log4jlogger.java
     |...
     |239. this.logger.fatal(msg, throwable);
```

[Table of Contents] [Description]

Path 478:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
254. public void log(String message, int verbosity) //log4jlogger.java
     |...
     |256. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 479:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
273. public void log(String message, Throwable throwable, int
verbosity) //log4jlogger.java
     |...
     |275. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 480:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
      |
      ...
      110. stdinWriter.print(msg);
```

[Table of Contents] [Description]

Path 481:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
      |
      ...
      116. if (err.ready()) {
```

[Table of Contents] [Description]

Path 482:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
      882. Object element = new Integer(rs.getInt("notifyid"));
```

[Table of Contents] [Description]

Path 483:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
      885. element = rs.getTimestamp("pagetime");
```

[Table of Contents] [Description]

Path 484:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
```

```
888. element = rs.getTimestamp("respondtime");
```

[Table of Contents] [Description]

Path 485:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
    ...
893. element = rs.getString("textmsg");
```

[Table of Contents] [Description]

Path 486:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
    ...
896. element = rs.getString("numericmsg");
```

[Table of Contents] [Description]

Path 487:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
    ...
899. element = rs.getString("answeredby");
```

[Table of Contents] [Description]

Path 488:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
    ...
902. element = new Integer(rs.getInt("nodeid"));
```

[Table of Contents] [Description]

Path 489:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
      905. element = rs.getString("interfaceid");
```

[Table of Contents] [Description]

Path 490:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
      908. element = new Integer(rs.getInt("eventid"));
```

[Table of Contents] [Description]

Path 491:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
875. protected static Notification[] rs2Notices(ResultSet rs) throws
SQLException //noticefactory.java
      |
      ...
      911. element = new Integer(rs.getInt("serviceid"));
```

[Table of Contents] [Description]

Path 492:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |
      ...
      542. outage.outageId = rs.getInt("outageid");
```

[Table of Contents] [Description]

Path 493:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |
      ...
      543. outage.nodeId = rs.getInt("nodeid");
```

[Table of Contents] [Description]

Path 494:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |...
      |544. outage.ipAddress = rs.getString("ipaddr");
```

[Table of Contents] [Description]

Path 495:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |...
      |545. outage.serviceId = rs.getInt("serviceid");
```

[Table of Contents] [Description]

Path 496:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |...
      |548. element = rs.getTimestamp("iflostservice");
```

[Table of Contents] [Description]

Path 497:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |...
      |552. outage.hostname = rs.getString("iphostname"); // from ipinterface
```

[Table of Contents] [Description]

Path 498:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |...
```

```
556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
```

[Table of Contents] [Description]

Path 499:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
    ...
559. outage.serviceName = rs.getString("servicename"); // from service
```

[Table of Contents] [Description]

Path 500:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
    ...
563. element = rs.getTimestamp("ifregainedservice");
```

[Table of Contents] [Description]

Path 501:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
    ...
569. intElement = rs.getInt("svcLostEventID");
```

[Table of Contents] [Description]

Path 502:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
    ...
575. intElement = rs.getInt("svcRegainedEventID");
```

[Table of Contents] [Description]

Path 503:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |
      ...
      581. intElement = rs.getInt("notifyid");
```

[Table of Contents] [Description]

Path 504:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      |
      ...
      587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
      ("answeredby");
```

[Table of Contents] [Description]

Path 505:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      |
      ...
      435. Object element = new Integer(rs.getInt("nodeid"));
```

[Table of Contents] [Description]

Path 506:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      |
      ...
      438. element = rs.getString("ipaddr");
```

[Table of Contents] [Description]

Path 507:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      |
      ...
      461. element = rs.getTimestamp("ifregainedservice");
```

[Table of Contents] [Description]

Path 508:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
444. element = rs.getString("nodeLabel");
```

[Table of Contents] [Description]

Path 509:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
447. element = rs.getString("iphostname");
```

[Table of Contents] [Description]

Path 510:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
450. element = rs.getString("servicename");
```

[Table of Contents] [Description]

Path 511:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
453. outage.outageId = rs.getInt("outageid");
```

[Table of Contents] [Description]

Path 512:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      455. element = rs.getTimestamp("iflostservice");
```

[Table of Contents] [Description]

Path 513:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      461. element = rs.getTimestamp("ifregainedservice");
```

[Table of Contents] [Description]

Path 514:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      468. int intElement = rs.getInt("svclosteventid");
```

[Table of Contents] [Description]

Path 515:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      473. intElement = rs.getInt("notifyid");
```

[Table of Contents] [Description]

Path 516:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
478. element = rs.getString("answeredby");
```

[Table of Contents] [Description]

Path 517:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
133. protected void loadDTD(String dtdFileName) throws FileNotFoundException,
IOException //xmlwriter.java
...
139. while ((input = originalBuffer.readLine()) != null) {
```

[Table of Contents] [Description]

Path 518:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
95. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
...
113. if (err.ready()) {
```

[Table of Contents] [Description]

Path 519:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
99. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
...
117. if (err.ready()) {
```

[Table of Contents] [Description]

Path 520:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
449. Object element = new Integer(rs.getInt("vulnerabilityId"));
```

[Table of Contents] [Description]

Path 521:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      453. element = rs.getObject("nodeID");
```

[Table of Contents] [Description]

Path 522:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      459. element = rs.getString("ipAddr");
```

[Table of Contents] [Description]

Path 523:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      463. element = rs.getObject("serviceID");
```

[Table of Contents] [Description]

Path 524:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      469. element = rs.getString("nodeLabel");
```

[Table of Contents] [Description]

Path 525:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      473. element = rs.getString("serviceName");
```

[Table of Contents] [Description]

Path 526:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      |
      ...
      476. element = rs.getTimestamp("creationTime");
```

[Table of Contents] [Description]

Path 527:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      |
      ...
      479. element = rs.getTimestamp("lastAttemptTime");
```

[Table of Contents] [Description]

Path 528:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      |
      ...
      482. element = rs.getTimestamp("lastScanTime");
```

[Table of Contents] [Description]

Path 529:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      |
      ...
      486. element = rs.getTimestamp("resolvedTime");
```

[Table of Contents] [Description]

Path 530:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      |
```

```
...
492. element = rs.getString("descr");
```

[Table of Contents] [Description]

Path 531:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
496. element = rs.getString("logmsg");
```

[Table of Contents] [Description]

Path 532:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
499. element = new Integer(rs.getInt("severity"));
```

[Table of Contents] [Description]

Path 533:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
502. element = new Integer(rs.getInt("pluginID"));
```

[Table of Contents] [Description]

Path 534:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
505. element = new Integer(rs.getInt("pluginSubID"));
```

[Table of Contents] [Description]

Path 535:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      509. element = rs.getString("descr");
```

[Table of Contents] [Description]

Path 536:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      513. element = rs.getObject("port");
```

[Table of Contents] [Description]

Path 537:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      519. element = rs.getString("protocol");
```

[Table of Contents] [Description]

Path 538:

Query Name - **Improper_Exception_Handling**

Severity - **Information**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      523. element = rs.getString("cveEntry");
```

[Table of Contents] [Description]

Unclosed objects

CWE ID	10031
Description	Closing objects inside a try clause, without a matching Close statement in the finally section may lead to resouce leaking in case of exception.
Likelihood of Exploit	Low
Common Consequences	In case of exception, open objects may not be closed. Make sure that the marked Close statement, which is

within a try clause, has a matching Close statement in the finally clause.

Potential Mitigations Add a matching close statement to the finally clause.

Applicable Platforms All

Path 1:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      ...
131. inputStream.close();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
      ...
102. ir.close();
```

[Table of Contents] [Description]

Path 3:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
      ...
198. is.close();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      ...
102. ir.close();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      |...
      |163. is.close();
```

[Table of Contents] [Description]

Path 6:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
625. public void stop() //runnableconsumerthreadpool.java
      |...
      |635. m_delegateQ.close();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      |...
      |122. processInput.close();
```

[Table of Contents] [Description]

Path 8:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
83. public void run() //processexec.java
      |...
      |96. m_inputStream.close();
```

[Table of Contents] [Description]

Path 9:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
459. public void closeConnection() //availabilitydata.java
      |...
      |463. m_availConn.close();
```

[Table of Contents] [Description]

Path 10:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
      |614. svcRS.close();
```

[Table of Contents] [Description]

Path 11:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      |...
      |723. rs.close();
```

[Table of Contents] [Description]

Path 12:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
      |...
      |229. fileWriter.close();
```

[Table of Contents] [Description]

Path 13:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
      |113. fotFileName.close();
```

[Table of Contents] [Description]

Path 14:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      |...
      |164. fot.close();
```

[Table of Contents] [Description]

Path 15:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
432. private void close() //eventsarchiver.java
      ...
437. m_eventsGetStmt.close();
```

[Table of Contents] [Description]

Path 16:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
432. private void close() //eventsarchiver.java
      ...
438. m_eventDeleteStmt.close();
```

[Table of Contents] [Description]

Path 17:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
432. private void close() //eventsarchiver.java
      ...
440. m_dbConn.close();
```

[Table of Contents] [Description]

Path 18:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
475. public static void main(String[] args) //eventsarchiver.java
      ...
485. ea.close();
```

[Table of Contents] [Description]

Path 19:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
171. protected void closeSocket(Socket socket, ConnectionConfig
config) //abstracttcpplugin.java
      ...
174. socket.close();
```

[Table of Contents] [Description]

Path 20:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
       ...
540. rs.close();
```

[Table of Contents] [Description]

Path 21:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
       ...
939. session.close();
```

[Table of Contents] [Description]

Path 22:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
       ...
1614. session.close();
```

[Table of Contents] [Description]

Path 23:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
       ...
1691. session.close();
```

[Table of Contents] [Description]

Path 24:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1589. private List getAddressesFromURL(String url) //capsdconfigfactory.java
       ...
1627. buffer.close();
```

[Table of Contents] [Description]

Path 25:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
      ...
916. disconnect.m_delegate.close();
```

[Table of Contents] [Description]

Path 26:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      ...
237. rows.close();
```

[Table of Contents] [Description]

Path 27:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      ...
242. stmt.close();
```

[Table of Contents] [Description]

Path 28:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
225. public synchronized void stop() //client.java
      ...
228. m_objsOut.close();
```

[Table of Contents] [Description]

Path 29:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
225. public synchronized void stop() //client.java
      ...
```

```
229. m_client.close();
```

[Table of Contents] [Description]

Path 30:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
245. public void run() //client.java
     ...
328. input.close();
```

[Table of Contents] [Description]

Path 31:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
245. public void run() //client.java
     ...
329. m_client.close();
```

[Table of Contents] [Description]

Path 32:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
250. m_server.close();
```

[Table of Contents] [Description]

Path 33:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
265. m_server.close();
```

[Table of Contents] [Description]

Path 34:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
281. public synchronized void stop() //dhcpd.java
```

```
...
296. m_server.close();
```

[Table of Contents] [Description]

Path 35:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
...
253. m_ins.close();
```

[Table of Contents] [Description]

Path 36:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
...
264. m_ins.close();
```

[Table of Contents] [Description]

Path 37:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
...
255. m_connection.close();
```

[Table of Contents] [Description]

Path 38:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
...
264. m_ins.close();
```

[Table of Contents] [Description]

Path 39:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
```

```
...
265. m_outs.close();
```

[Table of Contents] [Description]

Path 40:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
...
266. m_connection.close();
```

[Table of Contents] [Description]

Path 41:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
280. public void close() //poller.java
...
285. m_ins.close();
```

[Table of Contents] [Description]

Path 42:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
280. public void close() //poller.java
...
286. m_outs.close();
```

[Table of Contents] [Description]

Path 43:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
280. public void close() //poller.java
...
287. m_connection.close();
```

[Table of Contents] [Description]

Path 44:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
164. private boolean addToSpecificsFromURL(List specifics, String url, long timeout,
```

```
int retries) //discovery.java
    ...
206. buffer.close();
```

[Table of Contents] [Description]

Path 45:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
    ...
458. m_eventReader.close();
```

[Table of Contents] [Description]

Path 46:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
150. public static void loadConfiguration(Reader rdr) throws IOException,
MarshalException, ValidationException //eventconfigurationmanager.java
    ...
188. rdr.close();
```

[Table of Contents] [Description]

Path 47:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
232. public static void mergeConfiguration(Reader rdr) throws MarshalException,
ValidationException //eventconfigurationmanager.java
    ...
246. rdr.close();
```

[Table of Contents] [Description]

Path 48:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
    ...
447. m_out.close();
```

[Table of Contents] [Description]

Path 49:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
261. m_tcpSock.close();
```

[Table of Contents] [Description]

Path 50:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
378. m_tcpSock.close();
```

[Table of Contents] [Description]

Path 51:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
193. m_connection.close();
```

[Table of Contents] [Description]

Path 52:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
502. m_connection.close();
```

[Table of Contents] [Description]

Path 53:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
203. rset.close();
```

[Table of Contents] [Description]

Path 54:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
208. stmt.close();
```

[Table of Contents] [Description]

Path 55:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
291. rset.close();
```

[Table of Contents] [Description]

Path 56:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
296. stmt.close();
```

[Table of Contents] [Description]

Path 57:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
72. public static void close(Connection dbConn) //dbutil.java
       ...
78. dbConn.close();
```

[Table of Contents] [Description]

Path 58:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
386. m_eventReceiver.close();
```

[Table of Contents] [Description]

Path 59:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
179. socket.close();
```

[Table of Contents] [Description]

Path 60:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
     ...
291. session.close();
```

[Table of Contents] [Description]

Path 61:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
415. public void close() //nsclientmanager.java
     ...
417. m_Socket.close();
```

[Table of Contents] [Description]

Path 62:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
562. svcRS.close();
```

[Table of Contents] [Description]

Path 63:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
604. servicesGetStmt.close();
```

[Table of Contents] [Description]

Path 64:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
607. ipInfoGetStmt.close();
```

[Table of Contents] [Description]

Path 65:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
610. outagesGetStmt.close();
```

[Table of Contents] [Description]

Path 66:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1249. servicesGetStmt.close();
```

[Table of Contents] [Description]

Path 67:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1252. ipInfoGetStmt.close();
```

[Table of Contents] [Description]

Path 68:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
```

```
1255. outagesGetStmt.close();
```

[Table of Contents] [Description]

Path 69:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
...
305. inp.close();
```

[Table of Contents] [Description]

Path 70:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
...
306. inr.close();
```

[Table of Contents] [Description]

Path 71:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
...
444. inp.close();
```

[Table of Contents] [Description]

Path 72:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
...
445. inr.close();
```

[Table of Contents] [Description]

Path 73:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
        ...
682. m_eventReceiver.close();
```

[Table of Contents] [Description]

Path 74:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
71. public void run() //pipedmarshaller.java
        ...
75. m_out.close();
```

[Table of Contents] [Description]

Path 75:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
        ...
577. rs.close();
```

[Table of Contents] [Description]

Path 76:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
        ...
504. m_trapSession.close();
```

[Table of Contents] [Description]

Path 77:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
180. public static String[] getIfLabels(int nodeId) throws
SQLException //iflabel.java
        ...
198. rs.close();
```

[Table of Contents] [Description]

Path 78:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
180. public static String[] getIfLabels(int nodeId) throws
SQLException //iflabel.java
      ...
      199. stmt.close();
```

[Table of Contents] [Description]

Path 79:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
209. public static String getIfLabel(int nodeId, String ipAddr) throws
SQLException //iflabel.java
      ...
      241. rs.close();
```

[Table of Contents] [Description]

Path 80:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
209. public static String getIfLabel(int nodeId, String ipAddr) throws
SQLException //iflabel.java
      ...
      242. stmt.close();
```

[Table of Contents] [Description]

Path 81:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
      ...
      283. rs.close();
```

[Table of Contents] [Description]

Path 82:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
      ...
      284. stmt.close();
```

[Table of Contents] [Description]

Path 83:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
87. public static java.util.List parse(String url) //iplistfromurl.java
      ...
123. buffer.close();
```

[Table of Contents] [Description]

Path 84:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
132. public void send(Log eventLog) throws EventProxyException //tcpeventproxy.java
      ...
138. connection.close();
```

[Table of Contents] [Description]

Path 85:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
191. public void close() //tcpeventproxy.java
      ...
194. m_sock.close();
```

[Table of Contents] [Description]

Path 86:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
250. public void runUpdate(String sql) //vacuumd.java
      ...
263. stmt.close();
```

[Table of Contents] [Description]

Path 87:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
428. public void doTestLoadLibraries() //manager.java
      ...
446. s.close();
```

[Table of Contents] [Description]

Path 88:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
       ...
486. in.close();
```

[Table of Contents] [Description]

Path 89:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
       ...
499. in.close();
```

[Table of Contents] [Description]

Path 90:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
59. static public void releaseConnection(Socket
socket) //nessusconnectionfactory.java
       ...
63. socket.close();
```

[Table of Contents] [Description]

Path 91:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
       ...
151. rs.close();
```

[Table of Contents] [Description]

Path 92:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
       ...
152. stmt.close();
```

[Table of Contents] [Description]

Path 93:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
292. private void writeURLFile(java.util.List interfaceList) throws
ServletException //managenodeservlet.java
      ...
307. fileWriter.close();
```

[Table of Contents] [Description]

Path 94:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
292. private void writeURLFile(java.util.List interfaceList) throws
ServletException //managenodesservlet.java
      ...
307. fileWriter.close();
```

[Table of Contents] [Description]

Path 95:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
68. rs.close();
```

[Table of Contents] [Description]

Path 96:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
69. stmt.close();
```

[Table of Contents] [Description]

Path 97:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
77. public Asset[] getAllAssets() throws SQLException //assetmodel.java
      ...
87. rs.close();
```

[Table of Contents] [Description]

Path 98:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
77. public Asset[] getAllAssets() throws SQLException //assetmodel.java
      ...
88. stmt.close();
```

[Table of Contents] [Description]

Path 99:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
96. public void createAsset(Asset asset) throws SQLException //assetmodel.java
      ...
147. stmt.close();
```

[Table of Contents] [Description]

Path 100:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
153. public void modifyAsset(Asset asset) throws SQLException //assetmodel.java
      ...
204. stmt.close();
```

[Table of Contents] [Description]

Path 101:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
      ...
255. rs.close();
```

[Table of Contents] [Description]

Path 102:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
      ...
```

```
256. stmt.close();
```

[Table of Contents] [Description]

Path 103:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
192. public List getCurrentAssetNodesList() throws
SQLException //importassetsservlet.java
...
204. rs.close();
```

[Table of Contents] [Description]

Path 104:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
192. public List getCurrentAssetNodesList() throws
SQLException //importassetsservlet.java
...
205. stmt.close();
```

[Table of Contents] [Description]

Path 105:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
...
105. out.close();
```

[Table of Contents] [Description]

Path 106:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
105. rs.close();
```

[Table of Contents] [Description]

Path 107:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      |...
      |106. stmt.close();
```

[Table of Contents] [Description]

Path 108:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      |...
      |130. rs.close();
```

[Table of Contents] [Description]

Path 109:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      |...
      |231. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 110:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      |...
      |152. rs.close();
```

[Table of Contents] [Description]

Path 111:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      |...
      |153. stmt.close();
```

[Table of Contents] [Description]

Path 112:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
     |...
     |185. rs.close();
```

[Table of Contents] [Description]

Path 113:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
     |...
     |186. stmt.close();
```

[Table of Contents] [Description]

Path 114:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
     |...
     |212. rs.close();
```

[Table of Contents] [Description]

Path 115:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
     |...
     |213. stmt.close();
```

[Table of Contents] [Description]

Path 116:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
     |...
```

```
615. node.m_dpname = rs.getString("dpName");
```

[Table of Contents] [Description]

Path 117:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
...
636. node.m_label = rs.getString("nodelabel");
```

[Table of Contents] [Description]

Path 118:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
248. public static String getHostname(String ipAddress) throws
SQLException //networkelementfactory.java
...
265. rs.close();
```

[Table of Contents] [Description]

Path 119:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
248. public static String getHostname(String ipAddress) throws
SQLException //networkelementfactory.java
...
266. stmt.close();
```

[Table of Contents] [Description]

Path 120:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
274. public static Interface getInterface(int nodeId, String ipAddress) throws
SQLException //networkelementfactory.java
210. nodes = rs2Nodes(rs);
```

[Table of Contents] [Description]

Path 121:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
274. public static Interface getInterface(int nodeId, String ipAddress) throws
SQLException //networkelementfactory.java
      ...
      291. stmt.close();
```

[Table of Contents] [Description]

Path 122:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
306. public static Interface getInterface(int nodeId, String ipAddress, int ifindex)
throws SQLException //networkelementfactory.java
      ...
      324. rs.close();
```

[Table of Contents] [Description]

Path 123:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
306. public static Interface getInterface(int nodeId, String ipAddress, int ifindex)
throws SQLException //networkelementfactory.java
      ...
      325. stmt.close();
```

[Table of Contents] [Description]

Path 124:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
340. public static Interface[] getInterfacesWithIpAddress(String ipAddress) throws
SQLException //networkelementfactory.java
      ...
      355. rs.close();
```

[Table of Contents] [Description]

Path 125:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
340. public static Interface[] getInterfacesWithIpAddress(String ipAddress) throws
SQLException //networkelementfactory.java
      ...
      356. stmt.close();
```

[Table of Contents] [Description]

Path 126:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
367. public static Interface[] getAllInterfacesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
     ...
378. rs.close();
```

[Table of Contents] [Description]

Path 127:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
367. public static Interface[] getAllInterfacesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
     ...
379. stmt.close();
```

[Table of Contents] [Description]

Path 128:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
389. public static Interface[] getActiveInterfacesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
     ...
400. rs.close();
```

[Table of Contents] [Description]

Path 129:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
389. public static Interface[] getActiveInterfacesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
     ...
401. stmt.close();
```

[Table of Contents] [Description]

Path 130:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
423. public static Interface[] getAllInterfaces(boolean includeSNMP) throws
SQLException //networkelementfactory.java
     ...
```

```
433. rs.close();
```

[Table of Contents] [Description]

Path 131:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
423. public static Interface[] getAllInterfaces(boolean includeSNMP) throws
SQLException //networkelementfactory.java
...
434. stmt.close();
```

[Table of Contents] [Description]

Path 132:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
...
488. rs.close();
```

[Table of Contents] [Description]

Path 133:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
...
489. stmt.close();
```

[Table of Contents] [Description]

Path 134:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
497. public static Service[] getAllServices() throws
SQLException //networkelementfactory.java
...
507. rs.close();
```

[Table of Contents] [Description]

Path 135:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
497. public static Service[] getAllServices() throws
SQLException //networkelementfactory.java
      |
      ...
      508. stmt.close();
```

[Table of Contents] [Description]

Path 136:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      |
      ...
      542. rs.close();
```

[Table of Contents] [Description]

Path 137:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      |
      ...
      543. stmt.close();
```

[Table of Contents] [Description]

Path 138:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
554. public static Service[] getServicesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
      |
      ...
      565. rs.close();
```

[Table of Contents] [Description]

Path 139:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
554. public static Service[] getServicesOnNode(int nodeId) throws
SQLException //networkelementfactory.java
      |
      ...
      566. stmt.close();
```

[Table of Contents] [Description]

Path 140:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
577. public static Service[] getServicesOnNode(int nodeId, int serviceId) throws
SQLException //networkelementfactory.java
      |...
      |589. rs.close();
```

[Table of Contents] [Description]

Path 141:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
577. public static Service[] getServicesOnNode(int nodeId, int serviceId) throws
SQLException //networkelementfactory.java
      |...
      |590. stmt.close();
```

[Table of Contents] [Description]

Path 142:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
      |...
      |843. rs.close();
```

[Table of Contents] [Description]

Path 143:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
      |...
      |844. stmt.close();
```

[Table of Contents] [Description]

Path 144:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
191. public static int getEventCount(AcknowledgeType ackType, Filter[] filters)
throws SQLException //eventfactory.java
      |...
```

```
217. rs.close();
```

[Table of Contents] [Description]

Path 145:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
191. public static int getEventCount(AcknowledgeType ackType, Filter[] filters)
throws SQLException //eventfactory.java
...
218. stmt.close();
```

[Table of Contents] [Description]

Path 146:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. public static int[] getEventCountBySeverity(AcknowledgeType ackType, Filter[]
filters) throws SQLException //eventfactory.java
...
263. rs.close();
```

[Table of Contents] [Description]

Path 147:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
233. public static int[] getEventCountBySeverity(AcknowledgeType ackType, Filter[]
filters) throws SQLException //eventfactory.java
...
264. stmt.close();
```

[Table of Contents] [Description]

Path 148:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
289. rs.close();
```

[Table of Contents] [Description]

Path 149:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      |
      ...
      390. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 150:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      |
      ...
      394. rs.close();
```

[Table of Contents] [Description]

Path 151:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      |
      ...
      395. stmt.close();
```

[Table of Contents] [Description]

Path 152:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
713. public static Event[] getEventsForPoller(String poller, boolean
includeAcknowledged) throws SQLException //eventfactory.java
      |
      ...
      737. rs.close();
```

[Table of Contents] [Description]

Path 153:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
713. public static Event[] getEventsForPoller(String poller, boolean
includeAcknowledged) throws SQLException //eventfactory.java
      |
      ...
      738. stmt.close();
```

[Table of Contents] [Description]

Path 154:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
782. public static void acknowledge(int[] eventIds, String user, Date time) throws
SQLException //eventfactory.java
      |...
      |848. stmt.close();
```

[Table of Contents] [Description]

Path 155:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
827. public static void acknowledge(Filter[] filters, String user, Date time) throws
SQLException //eventfactory.java
      |...
      |848. stmt.close();
```

[Table of Contents] [Description]

Path 156:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
866. public static void acknowledgeAll(String user, Date time) throws
SQLException //eventfactory.java
      |...
      |879. stmt.close();
```

[Table of Contents] [Description]

Path 157:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
906. public static void unacknowledge(int[] eventIds) throws
SQLException //eventfactory.java
      |...
      |928. stmt.close();
```

[Table of Contents] [Description]

Path 158:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
938. public static void unacknowledge(Filter[] filters) throws
SQLException //eventfactory.java
      |...
```

```
957. stmt.close();
```

[Table of Contents] [Description]

Path 159:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
966. public static void unacknowledgeAll() throws SQLException //eventfactory.java
     ...
972. stmt.close();
```

[Table of Contents] [Description]

Path 160:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
     ...
205. tempIn.close();
```

[Table of Contents] [Description]

Path 161:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
     ...
207. out.close();
```

[Table of Contents] [Description]

Path 162:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparentservlet.java
     ...
79. stmt.close();
```

[Table of Contents] [Description]

Path 163:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
58. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException //svgservlet.java
      ...
      78. docwriter.close();
```

[Table of Contents] [Description]

Path 164:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException //svgtranscoder.java
      ...
      118. docwriter.close();
```

[Table of Contents] [Description]

Path 165:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
395. public static int getNoticeCount(AcknowledgeType ackType, Filter[] filters)
throws SQLException //noticefactory.java
      ...
      419. rs.close();
```

[Table of Contents] [Description]

Path 166:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
395. public static int getNoticeCount(AcknowledgeType ackType, Filter[] filters)
throws SQLException //noticefactory.java
      ...
      420. stmt.close();
```

[Table of Contents] [Description]

Path 167:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
429. public static Notification getNotice(int noticeId) throws
SQLException //noticefactory.java
      ...
      445. rs.close();
```

[Table of Contents] [Description]

Path 168:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
429. public static Notification getNotice(int noticeId) throws
SQLException //noticefactory.java
      |...
      |446. stmt.close();
```

[Table of Contents] [Description]

Path 169:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. public static boolean canDisplayEvent(int eventId) //noticefactory.java
      |...
      |481. statement.close();
```

[Table of Contents] [Description]

Path 170:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
462. public static boolean canDisplayEvent(int eventId) //noticefactory.java
      |...
      |482. results.close();
```

[Table of Contents] [Description]

Path 171:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
556. public static Notification[] getNotices(SortStyle sortStyle, AcknowledgeType
ackType, Filter[] filters, int limit, int offset) throws
SQLException //noticefactory.java
      |...
      |593. rs.close();
```

[Table of Contents] [Description]

Path 172:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
556. public static Notification[] getNotices(SortStyle sortStyle, AcknowledgeType
ackType, Filter[] filters, int limit, int offset) throws
SQLException //noticefactory.java
      |...
```

```
594. stmt.close();
```

[Table of Contents] [Description]

Path 173:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
641. public static Notification[] getNoticesForInterface(int nodeId, String
ipAddress, boolean includeAcknowledged) throws SQLException //noticefactory.java
...
665. rs.close();
```

[Table of Contents] [Description]

Path 174:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
641. public static Notification[] getNoticesForInterface(int nodeId, String
ipAddress, boolean includeAcknowledged) throws SQLException //noticefactory.java
...
666. stmt.close();
```

[Table of Contents] [Description]

Path 175:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
686. public static Notification[] getNoticesForInterface(String ipAddress, boolean
includeAcknowledged) throws SQLException //noticefactory.java
...
709. rs.close();
```

[Table of Contents] [Description]

Path 176:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
686. public static Notification[] getNoticesForInterface(String ipAddress, boolean
includeAcknowledged) throws SQLException //noticefactory.java
...
710. stmt.close();
```

[Table of Contents] [Description]

Path 177:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
727. public static Notification[] getNoticesForService(int nodeId, String ipAddress,
int serviceId, boolean includeAcknowledged) throws SQLException //noticefactory.java
      ...
752. rs.close();
```

[Table of Contents] [Description]

Path 178:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
727. public static Notification[] getNoticesForService(int nodeId, String ipAddress,
int serviceId, boolean includeAcknowledged) throws SQLException //noticefactory.java
      ...
753. stmt.close();
```

[Table of Contents] [Description]

Path 179:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
774. public static Notification[] getNoticesForService(int serviceId, boolean
includeAcknowledged) throws SQLException //noticefactory.java
      ...
793. rs.close();
```

[Table of Contents] [Description]

Path 180:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
774. public static Notification[] getNoticesForService(int serviceId, boolean
includeAcknowledged) throws SQLException //noticefactory.java
      ...
794. stmt.close();
```

[Table of Contents] [Description]

Path 181:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
837. public static void acknowledge(int[] noticeIds, String user, Date time) throws
SQLException //noticefactory.java
      ...
863. stmt.close();
```

[Table of Contents] [Description]

Path 182:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |...
      |185. rs.close();
```

[Table of Contents] [Description]

Path 183:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |...
      |186. pstmt.close();
```

[Table of Contents] [Description]

Path 184:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
202. public Notification[] allNotifications() throws
SQLException //notificationmodel.java
      |...
      |212. rs.close();
```

[Table of Contents] [Description]

Path 185:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
202. public Notification[] allNotifications() throws
SQLException //notificationmodel.java
      |...
      |213. stmt.close();
```

[Table of Contents] [Description]

Path 186:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      |...
```

```
283. rstmp.close();
```

[Table of Contents] [Description]

Path 187:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
...
284. stmttmp.close();
```

[Table of Contents] [Description]

Path 188:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
306. public Notification[] getOutstandingNotices() throws
SQLException //notificationmodel.java
...
316. rs.close();
```

[Table of Contents] [Description]

Path 189:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
306. public Notification[] getOutstandingNotices() throws
SQLException //notificationmodel.java
...
317. stmt.close();
```

[Table of Contents] [Description]

Path 190:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
332. public int getOutstandingNoticeCount() throws
SQLException //notificationmodel.java
...
345. rs.close();
```

[Table of Contents] [Description]

Path 191:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
332. public int getOutstandingNoticeCount() throws
SQLException //notificationmodel.java
      |...
      |346. stmt.close();
```

[Table of Contents] [Description]

Path 192:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
361. public int getOutstandingNoticeCount(String username) throws
SQLException //notificationmodel.java
      |...
      |379. rs.close();
```

[Table of Contents] [Description]

Path 193:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
361. public int getOutstandingNoticeCount(String username) throws
SQLException //notificationmodel.java
      |...
      |380. pstmt.close();
```

[Table of Contents] [Description]

Path 194:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      |...
      |407. rs.close();
```

[Table of Contents] [Description]

Path 195:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      |...
      |408. pstmt.close();
```

[Table of Contents] [Description]

Path 196:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
424. public void acknowledged(String name, int noticeId) throws
SQLException //notificationmodel.java
      |
      ...
      438. pstmt.close();
```

[Table of Contents] [Description]

Path 197:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
451. public void insert(Notification nbean) throws
SQLException //notificationmodel.java
      |
      ...
      474. pstmt.close();
```

[Table of Contents] [Description]

Path 198:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
277. public int execute() //command.java
      |
      ...
      335. processInput.close();
```

[Table of Contents] [Description]

Path 199:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
188. public static int getOutageCount() throws SQLException //outagefactory.java
      |
      ...
      200. rs.close();
```

[Table of Contents] [Description]

Path 200:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
188. public static int getOutageCount() throws SQLException //outagefactory.java
      |
      ...
      201. stmt.close();
```

[Table of Contents] [Description]

Path 201:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
212. public static int getOutageCount(OutageType ackType, Filter[] filters) throws
SQLException //outagefactory.java
      |
      ...
      236. rs.close();
```

[Table of Contents] [Description]

Path 202:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
212. public static int getOutageCount(OutageType ackType, Filter[] filters) throws
SQLException //outagefactory.java
      |
      ...
      237. stmt.close();
```

[Table of Contents] [Description]

Path 203:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      |
      ...
      260. rs.close();
```

[Table of Contents] [Description]

Path 204:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      |
      ...
      261. stmt.close();
```

[Table of Contents] [Description]

Path 205:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
314. public static Outage[] getOutages(SortStyle sortStyle, OutageType outType,
Filter[] filters, int limit, int offset) throws SQLException //outagefactory.java
      |
```

```
...
352. rs.close();
```

[Table of Contents] [Description]

Path 206:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
314. public static Outage[] getOutages(SortStyle sortStyle, OutageType outType,
Filter[] filters, int limit, int offset) throws SQLException //outagefactory.java
...
353. stmt.close();
```

[Table of Contents] [Description]

Path 207:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
75. public Outage[] getCurrentOutages() throws SQLException //outagemodel.java
...
85. rs.close();
```

[Table of Contents] [Description]

Path 208:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
75. public Outage[] getCurrentOutages() throws SQLException //outagemodel.java
...
86. stmt.close();
```

[Table of Contents] [Description]

Path 209:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public int getCurrentOutageCount() throws SQLException //outagemodel.java
...
106. rs.close();
```

[Table of Contents] [Description]

Path 210:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public int getCurrentOutageCount() throws SQLException //outagemodel.java
      ...
107. stmt.close();
```

[Table of Contents] [Description]

Path 211:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
115. public Outage[] getCurrentOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
126. rs.close();
```

[Table of Contents] [Description]

Path 212:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
115. public Outage[] getCurrentOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
127. stmt.close();
```

[Table of Contents] [Description]

Path 213:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
135. public Outage[] getNonCurrentOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
146. rs.close();
```

[Table of Contents] [Description]

Path 214:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
135. public Outage[] getNonCurrentOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
147. stmt.close();
```

[Table of Contents] [Description]

Path 215:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
158. public Outage[] getOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
      169. rs.close();
```

[Table of Contents] [Description]

Path 216:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
158. public Outage[] getOutagesForNode(int nodeId) throws
SQLException //outagemodel.java
      ...
      170. stmt.close();
```

[Table of Contents] [Description]

Path 217:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      206. rs.close();
```

[Table of Contents] [Description]

Path 218:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      227. rs.close();
```

[Table of Contents] [Description]

Path 219:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
215. public Outage[] getOutagesForInterface(int nodeId, String ipInterface) throws
SQLException //outagemodel.java
      ...
      227. rs.close();
```

[Table of Contents] [Description]

Path 220:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
215. public Outage[] getOutagesForInterface(int nodeId, String ipInterface) throws
SQLException //outagemodel.java
      ...
228. stmt.close();
```

[Table of Contents] [Description]

Path 221:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
267. rs.close();
```

[Table of Contents] [Description]

Path 222:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
268. stmt.close();
```

[Table of Contents] [Description]

Path 223:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
276. public Outage[] getOutagesForService(int nodeId, String ipInterface, int
serviceId) throws SQLException //outagemodel.java
      ...
289. rs.close();
```

[Table of Contents] [Description]

Path 224:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
276. public Outage[] getOutagesForService(int nodeId, String ipInterface, int
serviceId) throws SQLException //outagemodel.java
```

```
...
290. stmt.close();
```

[Table of Contents] [Description]

Path 225:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
...
332. rs.close();
```

[Table of Contents] [Description]

Path 226:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
...
333. stmt.close();
```

[Table of Contents] [Description]

Path 227:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
...
367. rs.close();
```

[Table of Contents] [Description]

Path 228:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
...
368. stmt.close();
```

[Table of Contents] [Description]

Path 229:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
385. public OutageSummary[] getCurrentSDSOutageSummaries() throws
SQLException //outagemodel.java
      |...
      405. rs.close();
```

[Table of Contents] [Description]

Path 230:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
385. public OutageSummary[] getCurrentSDSOutageSummaries() throws
SQLException //outagemodel.java
      |...
      406. stmt.close();
```

[Table of Contents] [Description]

Path 231:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
72. public static void makeBackup(String original, String fileName) throws
IOException //filebackup.java
      |...
      116. originalBuffer.close();
```

[Table of Contents] [Description]

Path 232:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
72. public static void makeBackup(String original, String fileName) throws
IOException //filebackup.java
      |...
      118. backupBuffer.close();
```

[Table of Contents] [Description]

Path 233:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
112. public static Properties loadLocal(String aFileName) //propertyloader.java
      |...
      120. pfile.close();
```

[Table of Contents] [Description]

Path 234:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
139. public static Properties loadLocal(String aFileName, Properties
someProperties) //propertyloader.java
      |
      ...
      147. pfile.close();
```

[Table of Contents] [Description]

Path 235:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
73. public static void backup(String original, String fileName) throws
XMLWriteException //xmlbackup.java
      |
      ...
      117. originalBuffer.close();
```

[Table of Contents] [Description]

Path 236:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
73. public static void backup(String original, String fileName) throws
XMLWriteException //xmlbackup.java
      |
      ...
      119. backupBuffer.close();
```

[Table of Contents] [Description]

Path 237:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
199. protected void serializeToFile() throws XMLWriteException //xmlwriter.java
      |
      ...
      218. writer.close();
```

[Table of Contents] [Description]

Path 238:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
66. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
      |
      ...
      82. out.close();
```

[Table of Contents] [Description]

Path 239:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
     ...
457. rs.close();
```

[Table of Contents] [Description]

Path 240:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
     ...
458. stmt.close();
```

[Table of Contents] [Description]

Path 241:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
     ...
86. out.close();
```

[Table of Contents] [Description]

Path 242:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
210. public static int getVulnerabilityCount(ResolutionType resType, Filter[]
filters) throws SQLException //vulnerabilityfactory.java
     ...
234. rs.close();
```

[Table of Contents] [Description]

Path 243:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
210. public static int getVulnerabilityCount(ResolutionType resType, Filter[]
```

```
filters) throws SQLException //vulnerabilityfactory.java
    ...
235. stmt.close();
```

[Table of Contents] [Description]

Path 244:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
    ...
261. rs.close();
```

[Table of Contents] [Description]

Path 245:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
    ...
262. stmt.close();
```

[Table of Contents] [Description]

Path 246:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
390. public static Vulnerability[] getVulnerabilities(SortStyle sortStyle,
ResolutionType resType, Filter[] filters, int limit, int offset) throws
SQLException //vulnerabilityfactory.java
    ...
428. rs.close();
```

[Table of Contents] [Description]

Path 247:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
390. public static Vulnerability[] getVulnerabilities(SortStyle sortStyle,
ResolutionType resType, Filter[] filters, int limit, int offset) throws
SQLException //vulnerabilityfactory.java
    ...
429. stmt.close();
```

[Table of Contents] [Description]

Path 248:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
109. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 249:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
117. try { connection.close(); } catch(Throwable tx) { }
```

[Table of Contents] [Description]

Path 250:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
139. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 251:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
153. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 252:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
282. public void run() //sockettest.java
      ...
294. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 253:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
282. public void run() //sockettest.java
     ...
331. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 254:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
53. public static void main (String args[]) //trapd.java
     ...
64. testTrapSession.close();
```

[Table of Contents] [Description]

Path 255:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
     ...
112. p_output.close();
```

[Table of Contents] [Description]

Path 256:

Query Name - **Unclosed_Objects**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
     ...
118. p_err.close();
```

[Table of Contents] [Description]

Catch Without General Exception Handling

CWE ID	10009
Description	Not handling general exceptions may cause the program to crash.
Likelihood of Exploit	Low
Common Consequences	Program crashing.

Potential Mitigations	Handle the case of general exception.

Demonstrative Examples	The following code catches specific exception but not the general one.
	
	...
	try
	{
	File.Copy(path, path2);
	}
	catch(IOException e)
	{
	Console.WriteLine(e.Message);
	}
	...

Applicable Platforms	All

Path 1:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      33. try {
```

[Table of Contents] [Description]

Path 2:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      66. try {
```

[Table of Contents] [Description]

Path 3:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
124. static final public SimpleNode Start() throws ParseException //parsemib.java
      |...
      129. try {
```

[Table of Contents] [Description]

Path 4:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
190. static final public void MibBegin() throws ParseException //parsemib.java
      |...
      |195. try {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
287. static final public void ModuleIdentity() throws ParseException //parsemib.java
      |...
      |292. try {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
348. static final public void Declaration() throws ParseException //parsemib.java
      |...
      |353. try {
```

[Table of Contents] [Description]

Path 7:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
808. static final public void Assignment() throws ParseException //parsemib.java
      |...
      |813. try {
```

[Table of Contents] [Description]

Path 8:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
890. static final public void SequenceOfVarsOrTextConvention() throws
ParseException //parsemib.java
      |...
      |895. try {
```

[Table of Contents] [Description]

Path 9:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
927. static final public void SequenceOfVars() throws ParseException //parsemib.java
      ...
932. try {
```

[Table of Contents] [Description]

Path 10:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1020. static final public void TextConvention() throws
ParseException //parsemib.java
      ...
1025. try {
```

[Table of Contents] [Description]

Path 11:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1218. static final public void Expression() throws ParseException //parsemib.java
      ...
1223. try {
```

[Table of Contents] [Description]

Path 12:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1335. static final public void MultipleOids() throws ParseException //parsemib.java
      ...
1340. try {
```

[Table of Contents] [Description]

Path 13:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1507. static final private boolean jj_2_1(int xla) //parsemib.java
      ...
1509. try { return !jj_3_1(); }
```

[Table of Contents] [Description]

Path 14:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1514. static final private boolean jj_2_2(int xla) //parsemib.java
        ...
1516. try { return !jj_3_2(); }
```

[Table of Contents] [Description]

Path 15:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1521. static final private boolean jj_2_3(int xla) //parsemib.java
        ...
1523. try { return !jj_3_3(); }
```

[Table of Contents] [Description]

Path 16:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1528. static final private boolean jj_2_4(int xla) //parsemib.java
        ...
1530. try { return !jj_3_4(); }
```

[Table of Contents] [Description]

Path 17:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1535. static final private boolean jj_2_5(int xla) //parsemib.java
        ...
1537. try { return !jj_3_5(); }
```

[Table of Contents] [Description]

Path 18:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1542. static final private boolean jj_2_6(int xla) //parsemib.java
        ...
1544. try { return !jj_3_6(); }
```

[Table of Contents] [Description]

Path 19:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1549. static final private boolean jj_2_7(int xla) //parsemib.java
      ...
1551. try { return !jj_3_7(); }
```

[Table of Contents] [Description]

Path 20:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1556. static final private boolean jj_2_8(int xla) //parsemib.java
      ...
1558. try { return !jj_3_8(); }
```

[Table of Contents] [Description]

Path 21:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1563. static final private boolean jj_2_9(int xla) //parsemib.java
      ...
1565. try { return !jj_3_9(); }
```

[Table of Contents] [Description]

Path 22:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1570. static final private boolean jj_2_10(int xla) //parsemib.java
      ...
1572. try { return !jj_3_10(); }
```

[Table of Contents] [Description]

Path 23:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1577. static final private boolean jj_2_11(int xla) //parsemib.java
      ...
1579. try { return !jj_3_11(); }
```

[Table of Contents] [Description]

Path 24:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1584. static final private boolean jj_2_12(int xla) //parsemib.java
        ...
1586. try { return !jj_3_12(); }
```

[Table of Contents] [Description]

Path 25:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1591. static final private boolean jj_2_13(int xla) //parsemib.java
        ...
1593. try { return !jj_3_13(); }
```

[Table of Contents] [Description]

Path 26:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1598. static final private boolean jj_2_14(int xla) //parsemib.java
        ...
1600. try { return !jj_3_14(); }
```

[Table of Contents] [Description]

Path 27:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1605. static final private boolean jj_2_15(int xla) //parsemib.java
        ...
1607. try { return !jj_3_15(); }
```

[Table of Contents] [Description]

Path 28:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1612. static final private boolean jj_2_16(int xla) //parsemib.java
        ...
1614. try { return !jj_3_16(); }
```

[Table of Contents] [Description]

Path 29:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1619. static final private boolean jj_2_17(int xla) //parsemib.java
      ...
1621. try { return !jj_3_17(); }
```

[Table of Contents] [Description]

Path 30:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1626. static final private boolean jj_2_18(int xla) //parsemib.java
      ...
1628. try { return !jj_3_18(); }
```

[Table of Contents] [Description]

Path 31:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1633. static final private boolean jj_2_19(int xla) //parsemib.java
      ...
1635. try { return !jj_3_19(); }
```

[Table of Contents] [Description]

Path 32:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1640. static final private boolean jj_2_20(int xla) //parsemib.java
      ...
1642. try { return !jj_3_20(); }
```

[Table of Contents] [Description]

Path 33:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1647. static final private boolean jj_2_21(int xla) //parsemib.java
      ...
1649. try { return !jj_3_21(); }
```

[Table of Contents] [Description]

Path 34:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1654. static final private boolean jj_2_22(int xla) //parsemib.java
      ...
1656. try { return !jj_3_22(); }
```

[Table of Contents] [Description]

Path 35:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1661. static final private boolean jj_2_23(int xla) //parsemib.java
      ...
1663. try { return !jj_3_23(); }
```

[Table of Contents] [Description]

Path 36:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1668. static final private boolean jj_2_24(int xla) //parsemib.java
      ...
1670. try { return !jj_3_24(); }
```

[Table of Contents] [Description]

Path 37:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1675. static final private boolean jj_2_25(int xla) //parsemib.java
      ...
1677. try { return !jj_3_25(); }
```

[Table of Contents] [Description]

Path 38:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1682. static final private boolean jj_2_26(int xla) //parsemib.java
      ...
1684. try { return !jj_3_26(); }
```

[Table of Contents] [Description]

Path 39:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1689. static final private boolean jj_2_27(int xla) //parsemib.java
      ...
1691. try { return !jj_3_27(); }
```

[Table of Contents] [Description]

Path 40:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1696. static final private boolean jj_2_28(int xla) //parsemib.java
      ...
1698. try { return !jj_3_28(); }
```

[Table of Contents] [Description]

Path 41:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1703. static final private boolean jj_2_29(int xla) //parsemib.java
      ...
1705. try { return !jj_3_29(); }
```

[Table of Contents] [Description]

Path 42:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1710. static final private boolean jj_2_30(int xla) //parsemib.java
      ...
1712. try { return !jj_3_30(); }
```

[Table of Contents] [Description]

Path 43:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1717. static final private boolean jj_2_31(int xla) //parsemib.java
      ...
1719. try { return !jj_3_31(); }
```

[Table of Contents] [Description]

Path 44:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1724. static final private boolean jj_2_32(int xla) //parsemib.java
      ...
1726. try { return !jj_3_32(); }
```

[Table of Contents] [Description]

Path 45:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1731. static final private boolean jj_2_33(int xla) //parsemib.java
      ...
1733. try { return !jj_3_33(); }
```

[Table of Contents] [Description]

Path 46:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1738. static final private boolean jj_2_34(int xla) //parsemib.java
      ...
1740. try { return !jj_3_34(); }
```

[Table of Contents] [Description]

Path 47:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1745. static final private boolean jj_2_35(int xla) //parsemib.java
      ...
1747. try { return !jj_3_35(); }
```

[Table of Contents] [Description]

Path 48:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1752. static final private boolean jj_2_36(int xla) //parsemib.java
      ...
1754. try { return !jj_3_36(); }
```

[Table of Contents] [Description]

Path 49:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1759. static final private boolean jj_2_37(int xla) //parsemib.java
      ...
1761. try { return !jj_3_37(); }
```

[Table of Contents] [Description]

Path 50:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1766. static final private boolean jj_2_38(int xla) //parsemib.java
      ...
1768. try { return !jj_3_38(); }
```

[Table of Contents] [Description]

Path 51:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1773. static final private boolean jj_2_39(int xla) //parsemib.java
      ...
1775. try { return !jj_3_39(); }
```

[Table of Contents] [Description]

Path 52:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1780. static final private boolean jj_2_40(int xla) //parsemib.java
      ...
1782. try { return !jj_3_40(); }
```

[Table of Contents] [Description]

Path 53:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1787. static final private boolean jj_2_41(int xla) //parsemib.java
      ...
1789. try { return !jj_3_41(); }
```

[Table of Contents] [Description]

Path 54:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1794. static final private boolean jj_2_42(int xla) //parsemib.java
        ...
1796. try { return !jj_3_42(); }
```

[Table of Contents] [Description]

Path 55:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1801. static final private boolean jj_2_43(int xla) //parsemib.java
        ...
1803. try { return !jj_3_43(); }
```

[Table of Contents] [Description]

Path 56:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1808. static final private boolean jj_2_44(int xla) //parsemib.java
        ...
1810. try { return !jj_3_44(); }
```

[Table of Contents] [Description]

Path 57:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1815. static final private boolean jj_2_45(int xla) //parsemib.java
        ...
1817. try { return !jj_3_45(); }
```

[Table of Contents] [Description]

Path 58:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1822. static final private boolean jj_2_46(int xla) //parsemib.java
        ...
1824. try { return !jj_3_46(); }
```

[Table of Contents] [Description]

Path 59:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1829. static final private boolean jj_2_47(int xla) //parsemib.java
      ...
1831. try { return !jj_3_47(); }
```

[Table of Contents] [Description]

Path 60:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1836. static final private boolean jj_2_48(int xla) //parsemib.java
      ...
1838. try { return !jj_3_48(); }
```

[Table of Contents] [Description]

Path 61:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1843. static final private boolean jj_2_49(int xla) //parsemib.java
      ...
1845. try { return !jj_3_49(); }
```

[Table of Contents] [Description]

Path 62:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1850. static final private boolean jj_2_50(int xla) //parsemib.java
      ...
1852. try { return !jj_3_50(); }
```

[Table of Contents] [Description]

Path 63:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1857. static final private boolean jj_2_51(int xla) //parsemib.java
      ...
1859. try { return !jj_3_51(); }
```

[Table of Contents] [Description]

Path 64:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1864. static final private boolean jj_2_52(int xla) //parsemib.java
      ...
1866. try { return !jj_3_52(); }
```

[Table of Contents] [Description]

Path 65:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1871. static final private boolean jj_2_53(int xla) //parsemib.java
      ...
1873. try { return !jj_3_53(); }
```

[Table of Contents] [Description]

Path 66:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1878. static final private boolean jj_2_54(int xla) //parsemib.java
      ...
1880. try { return !jj_3_54(); }
```

[Table of Contents] [Description]

Path 67:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1885. static final private boolean jj_2_55(int xla) //parsemib.java
      ...
1887. try { return !jj_3_55(); }
```

[Table of Contents] [Description]

Path 68:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1892. static final private boolean jj_2_56(int xla) //parsemib.java
      ...
1894. try { return !jj_3_56(); }
```

[Table of Contents] [Description]

Path 69:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1899. static final private boolean jj_2_57(int xla) //parsemib.java
      ...
1901. try { return !jj_3_57(); }
```

[Table of Contents] [Description]

Path 70:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1906. static final private boolean jj_2_58(int xla) //parsemib.java
      ...
1908. try { return !jj_3_58(); }
```

[Table of Contents] [Description]

Path 71:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1913. static final private boolean jj_2_59(int xla) //parsemib.java
      ...
1915. try { return !jj_3_59(); }
```

[Table of Contents] [Description]

Path 72:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1920. static final private boolean jj_2_60(int xla) //parsemib.java
      ...
1922. try { return !jj_3_60(); }
```

[Table of Contents] [Description]

Path 73:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1927. static final private boolean jj_2_61(int xla) //parsemib.java
      ...
1929. try { return !jj_3_61(); }
```

[Table of Contents] [Description]

Path 74:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1934. static final private boolean jj_2_62(int xla) //parsemib.java
      ...
1936. try { return !jj_3_62(); }
```

[Table of Contents] [Description]

Path 75:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1941. static final private boolean jj_2_63(int xla) //parsemib.java
      ...
1943. try { return !jj_3_63(); }
```

[Table of Contents] [Description]

Path 76:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1948. static final private boolean jj_2_64(int xla) //parsemib.java
      ...
1950. try { return !jj_3_64(); }
```

[Table of Contents] [Description]

Path 77:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1955. static final private boolean jj_2_65(int xla) //parsemib.java
      ...
1957. try { return !jj_3_65(); }
```

[Table of Contents] [Description]

Path 78:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1962. static final private boolean jj_2_66(int xla) //parsemib.java
      ...
1964. try { return !jj_3_66(); }
```

[Table of Contents] [Description]

Path 79:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1969. static final private boolean jj_2_67(int xla) //parsemib.java
      ...
1971. try { return !jj_3_67(); }
```

[Table of Contents] [Description]

Path 80:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1976. static final private boolean jj_2_68(int xla) //parsemib.java
      ...
1978. try { return !jj_3_68(); }
```

[Table of Contents] [Description]

Path 81:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1983. static final private boolean jj_2_69(int xla) //parsemib.java
      ...
1985. try { return !jj_3_69(); }
```

[Table of Contents] [Description]

Path 82:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1990. static final private boolean jj_2_70(int xla) //parsemib.java
      ...
1992. try { return !jj_3_70(); }
```

[Table of Contents] [Description]

Path 83:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1997. static final private boolean jj_2_71(int xla) //parsemib.java
      ...
1999. try { return !jj_3_71(); }
```

[Table of Contents] [Description]

Path 84:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2004. static final private boolean jj_2_72(int xla) //parsemib.java
      ...
2006. try { return !jj_3_72(); }
```

[Table of Contents] [Description]

Path 85:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2011. static final private boolean jj_2_73(int xla) //parsemib.java
      ...
2013. try { return !jj_3_73(); }
```

[Table of Contents] [Description]

Path 86:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2018. static final private boolean jj_2_74(int xla) //parsemib.java
      ...
2020. try { return !jj_3_74(); }
```

[Table of Contents] [Description]

Path 87:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2025. static final private boolean jj_2_75(int xla) //parsemib.java
      ...
2027. try { return !jj_3_75(); }
```

[Table of Contents] [Description]

Path 88:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2032. static final private boolean jj_2_76(int xla) //parsemib.java
      ...
2034. try { return !jj_3_76(); }
```

[Table of Contents] [Description]

Path 89:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2039. static final private boolean jj_2_77(int xla) //parsemib.java
        ...
2041. try { return !jj_3_77(); }
```

[Table of Contents] [Description]

Path 90:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2046. static final private boolean jj_2_78(int xla) //parsemib.java
        ...
2048. try { return !jj_3_78(); }
```

[Table of Contents] [Description]

Path 91:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2053. static final private boolean jj_2_79(int xla) //parsemib.java
        ...
2055. try { return !jj_3_79(); }
```

[Table of Contents] [Description]

Path 92:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2060. static final private boolean jj_2_80(int xla) //parsemib.java
        ...
2062. try { return !jj_3_80(); }
```

[Table of Contents] [Description]

Path 93:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2067. static final private boolean jj_2_81(int xla) //parsemib.java
        ...
2069. try { return !jj_3_81(); }
```

[Table of Contents] [Description]

Path 94:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2074. static final private boolean jj_2_82(int xla) //parsemib.java
      ...
2076. try { return !jj_3_82(); }
```

[Table of Contents] [Description]

Path 95:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2081. static final private boolean jj_2_83(int xla) //parsemib.java
      ...
2083. try { return !jj_3_83(); }
```

[Table of Contents] [Description]

Path 96:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2088. static final private boolean jj_2_84(int xla) //parsemib.java
      ...
2090. try { return !jj_3_84(); }
```

[Table of Contents] [Description]

Path 97:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2095. static final private boolean jj_2_85(int xla) //parsemib.java
      ...
2097. try { return !jj_3_85(); }
```

[Table of Contents] [Description]

Path 98:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2102. static final private boolean jj_2_86(int xla) //parsemib.java
      ...
2104. try { return !jj_3_86(); }
```

[Table of Contents] [Description]

Path 99:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2109. static final private boolean jj_2_87(int xla) //parsemib.java
        ...
2111. try { return !jj_3_87(); }
```

[Table of Contents] [Description]

Path 100:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2116. static final private boolean jj_2_88(int xla) //parsemib.java
        ...
2118. try { return !jj_3_88(); }
```

[Table of Contents] [Description]

Path 101:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2123. static final private boolean jj_2_89(int xla) //parsemib.java
        ...
2125. try { return !jj_3_89(); }
```

[Table of Contents] [Description]

Path 102:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2130. static final private boolean jj_2_90(int xla) //parsemib.java
        ...
2132. try { return !jj_3_90(); }
```

[Table of Contents] [Description]

Path 103:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2137. static final private boolean jj_2_91(int xla) //parsemib.java
        ...
2139. try { return !jj_3_91(); }
```

[Table of Contents] [Description]

Path 104:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2144. static final private boolean jj_2_92(int xla) //parsemib.java
        ...
2146. try { return !jj_3_92(); }
```

[Table of Contents] [Description]

Path 105:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2151. static final private boolean jj_2_93(int xla) //parsemib.java
        ...
2153. try { return !jj_3_93(); }
```

[Table of Contents] [Description]

Path 106:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2158. static final private boolean jj_2_94(int xla) //parsemib.java
        ...
2160. try { return !jj_3_94(); }
```

[Table of Contents] [Description]

Path 107:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2165. static final private boolean jj_2_95(int xla) //parsemib.java
        ...
2167. try { return !jj_3_95(); }
```

[Table of Contents] [Description]

Path 108:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2172. static final private boolean jj_2_96(int xla) //parsemib.java
        ...
2174. try { return !jj_3_96(); }
```

[Table of Contents] [Description]

Path 109:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2179. static final private boolean jj_2_97(int xla) //parsemib.java
        ...
2181. try { return !jj_3_97(); }
```

[Table of Contents] [Description]

Path 110:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2186. static final private boolean jj_2_98(int xla) //parsemib.java
        ...
2188. try { return !jj_3_98(); }
```

[Table of Contents] [Description]

Path 111:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2193. static final private boolean jj_2_99(int xla) //parsemib.java
        ...
2195. try { return !jj_3_99(); }
```

[Table of Contents] [Description]

Path 112:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2200. static final private boolean jj_2_100(int xla) //parsemib.java
        ...
2202. try { return !jj_3_100(); }
```

[Table of Contents] [Description]

Path 113:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2207. static final private boolean jj_2_101(int xla) //parsemib.java
        ...
2209. try { return !jj_3_101(); }
```

[Table of Contents] [Description]

Path 114:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2214. static final private boolean jj_2_102(int xla) //parsemib.java
        ...
2216. try { return !jj_3_102(); }
```

[Table of Contents] [Description]

Path 115:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2221. static final private boolean jj_2_103(int xla) //parsemib.java
        ...
2223. try { return !jj_3_103(); }
```

[Table of Contents] [Description]

Path 116:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2228. static final private boolean jj_2_104(int xla) //parsemib.java
        ...
2230. try { return !jj_3_104(); }
```

[Table of Contents] [Description]

Path 117:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2235. static final private boolean jj_2_105(int xla) //parsemib.java
        ...
2237. try { return !jj_3_105(); }
```

[Table of Contents] [Description]

Path 118:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2242. static final private boolean jj_2_106(int xla) //parsemib.java
        ...
2244. try { return !jj_3_106(); }
```

[Table of Contents] [Description]

Path 119:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2249. static final private boolean jj_2_107(int xla) //parsemib.java
        ...
2251. try { return !jj_3_107(); }
```

[Table of Contents] [Description]

Path 120:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2256. static final private boolean jj_2_108(int xla) //parsemib.java
        ...
2258. try { return !jj_3_108(); }
```

[Table of Contents] [Description]

Path 121:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2263. static final private boolean jj_2_109(int xla) //parsemib.java
        ...
2265. try { return !jj_3_109(); }
```

[Table of Contents] [Description]

Path 122:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2270. static final private boolean jj_2_110(int xla) //parsemib.java
        ...
2272. try { return !jj_3_110(); }
```

[Table of Contents] [Description]

Path 123:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2277. static final private boolean jj_2_111(int xla) //parsemib.java
        ...
2279. try { return !jj_3_111(); }
```

[Table of Contents] [Description]

Path 124:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2284. static final private boolean jj_2_112(int xla) //parsemib.java
        ...
2286. try { return !jj_3_112(); }
```

[Table of Contents] [Description]

Path 125:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2291. static final private boolean jj_2_113(int xla) //parsemib.java
        ...
2293. try { return !jj_3_113(); }
```

[Table of Contents] [Description]

Path 126:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2298. static final private boolean jj_2_114(int xla) //parsemib.java
        ...
2300. try { return !jj_3_114(); }
```

[Table of Contents] [Description]

Path 127:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2305. static final private boolean jj_2_115(int xla) //parsemib.java
        ...
2307. try { return !jj_3_115(); }
```

[Table of Contents] [Description]

Path 128:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2312. static final private boolean jj_2_116(int xla) //parsemib.java
        ...
2314. try { return !jj_3_116(); }
```

[Table of Contents] [Description]

Path 129:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2319. static final private boolean jj_2_117(int xla) //parsemib.java
      ...
2321. try { return !jj_3_117(); }
```

[Table of Contents] [Description]

Path 130:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2326. static final private boolean jj_2_118(int xla) //parsemib.java
      ...
2328. try { return !jj_3_118(); }
```

[Table of Contents] [Description]

Path 131:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2333. static final private boolean jj_2_119(int xla) //parsemib.java
      ...
2335. try { return !jj_3_119(); }
```

[Table of Contents] [Description]

Path 132:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2340. static final private boolean jj_2_120(int xla) //parsemib.java
      ...
2342. try { return !jj_3_120(); }
```

[Table of Contents] [Description]

Path 133:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2347. static final private boolean jj_2_121(int xla) //parsemib.java
      ...
2349. try { return !jj_3_121(); }
```

[Table of Contents] [Description]

Path 134:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2354. static final private boolean jj_2_122(int xla) //parsemib.java
        ...
2356. try { return !jj_3_122(); }
```

[Table of Contents] [Description]

Path 135:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2361. static final private boolean jj_2_123(int xla) //parsemib.java
        ...
2363. try { return !jj_3_123(); }
```

[Table of Contents] [Description]

Path 136:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2368. static final private boolean jj_2_124(int xla) //parsemib.java
        ...
2370. try { return !jj_3_124(); }
```

[Table of Contents] [Description]

Path 137:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2375. static final private boolean jj_2_125(int xla) //parsemib.java
        ...
2377. try { return !jj_3_125(); }
```

[Table of Contents] [Description]

Path 138:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2382. static final private boolean jj_2_126(int xla) //parsemib.java
        ...
2384. try { return !jj_3_126(); }
```

[Table of Contents] [Description]

Path 139:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2389. static final private boolean jj_2_127(int xla) //parsemib.java
        ...
2391. try { return !jj_3_127(); }
```

[Table of Contents] [Description]

Path 140:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2396. static final private boolean jj_2_128(int xla) //parsemib.java
        ...
2398. try { return !jj_3_128(); }
```

[Table of Contents] [Description]

Path 141:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2403. static final private boolean jj_2_129(int xla) //parsemib.java
        ...
2405. try { return !jj_3_129(); }
```

[Table of Contents] [Description]

Path 142:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2410. static final private boolean jj_2_130(int xla) //parsemib.java
        ...
2412. try { return !jj_3_130(); }
```

[Table of Contents] [Description]

Path 143:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2417. static final private boolean jj_2_131(int xla) //parsemib.java
        ...
2419. try { return !jj_3_131(); }
```

[Table of Contents] [Description]

Path 144:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2424. static final private boolean jj_2_132(int xla) //parsemib.java
        ...
2426. try { return !jj_3_132(); }
```

[Table of Contents] [Description]

Path 145:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2431. static final private boolean jj_2_133(int xla) //parsemib.java
        ...
2433. try { return !jj_3_133(); }
```

[Table of Contents] [Description]

Path 146:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2438. static final private boolean jj_2_134(int xla) //parsemib.java
        ...
2440. try { return !jj_3_134(); }
```

[Table of Contents] [Description]

Path 147:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2445. static final private boolean jj_2_135(int xla) //parsemib.java
        ...
2447. try { return !jj_3_135(); }
```

[Table of Contents] [Description]

Path 148:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2452. static final private boolean jj_2_136(int xla) //parsemib.java
        ...
2454. try { return !jj_3_136(); }
```

[Table of Contents] [Description]

Path 149:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2459. static final private boolean jj_2_137(int xla) //parsemib.java
      ...
2461. try { return !jj_3_137(); }
```

[Table of Contents] [Description]

Path 150:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2466. static final private boolean jj_2_138(int xla) //parsemib.java
      ...
2468. try { return !jj_3_138(); }
```

[Table of Contents] [Description]

Path 151:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2473. static final private boolean jj_2_139(int xla) //parsemib.java
      ...
2475. try { return !jj_3_139(); }
```

[Table of Contents] [Description]

Path 152:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2480. static final private boolean jj_2_140(int xla) //parsemib.java
      ...
2482. try { return !jj_3_140(); }
```

[Table of Contents] [Description]

Path 153:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2487. static final private boolean jj_2_141(int xla) //parsemib.java
      ...
2489. try { return !jj_3_141(); }
```

[Table of Contents] [Description]

Path 154:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2494. static final private boolean jj_2_142(int xla) //parsemib.java
        ...
2496. try { return !jj_3_142(); }
```

[Table of Contents] [Description]

Path 155:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2501. static final private boolean jj_2_143(int xla) //parsemib.java
        ...
2503. try { return !jj_3_143(); }
```

[Table of Contents] [Description]

Path 156:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2508. static final private boolean jj_2_144(int xla) //parsemib.java
        ...
2510. try { return !jj_3_144(); }
```

[Table of Contents] [Description]

Path 157:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2515. static final private boolean jj_2_145(int xla) //parsemib.java
        ...
2517. try { return !jj_3_145(); }
```

[Table of Contents] [Description]

Path 158:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2522. static final private boolean jj_2_146(int xla) //parsemib.java
        ...
2524. try { return !jj_3_146(); }
```

[Table of Contents] [Description]

Path 159:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2529. static final private boolean jj_2_147(int xla) //parsemib.java
      ...
2531. try { return !jj_3_147(); }
```

[Table of Contents] [Description]

Path 160:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2536. static final private boolean jj_2_148(int xla) //parsemib.java
      ...
2538. try { return !jj_3_148(); }
```

[Table of Contents] [Description]

Path 161:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2543. static final private boolean jj_2_149(int xla) //parsemib.java
      ...
2545. try { return !jj_3_149(); }
```

[Table of Contents] [Description]

Path 162:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2550. static final private boolean jj_2_150(int xla) //parsemib.java
      ...
2552. try { return !jj_3_150(); }
```

[Table of Contents] [Description]

Path 163:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2557. static final private boolean jj_2_151(int xla) //parsemib.java
      ...
2559. try { return !jj_3_151(); }
```

[Table of Contents] [Description]

Path 164:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2564. static final private boolean jj_2_152(int xla) //parsemib.java
        ...
2566. try { return !jj_3_152(); }
```

[Table of Contents] [Description]

Path 165:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2571. static final private boolean jj_2_153(int xla) //parsemib.java
        ...
2573. try { return !jj_3_153(); }
```

[Table of Contents] [Description]

Path 166:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2578. static final private boolean jj_2_154(int xla) //parsemib.java
        ...
2580. try { return !jj_3_154(); }
```

[Table of Contents] [Description]

Path 167:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
219. static private final int jjStartNfaWithStates_0(int pos, int kind, int
state) //parsemibtokenmanager.java
        ...
223. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 168:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
293. static private final int jjMoveStringLiteralDfa1_0(long active0, long
active1) //parsemibtokenmanager.java
        ...
295. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 169:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
339. static private final int jjMoveStringLiteralDfa2_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
      ...
      343. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 170:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. static private final int jjMoveStringLiteralDfa3_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
      ...
      401. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 171:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      447. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 172:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      399. if (((active0 &= old0) | (active1 &= old1)) == 0L)
```

[Table of Contents] [Description]

Path 173:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. static private final int jjMoveStringLiteralDfa6_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      561. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 174:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
601. static private final int jjMoveStringLiteralDfa7_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
605. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 175:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
647. static private final int jjMoveStringLiteralDfa8_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
651. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 176:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
695. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 177:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
735. static private final int jjMoveStringLiteralDfa10_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
739. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 178:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
```

```
active0) //parsemibtokenmanager.java
    ...
773. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 179:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
811. static private final int jjMoveStringLiteralDfa12_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
815. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 180:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
839. static private final int jjMoveStringLiteralDfa13_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
843. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 181:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
865. static private final int jjMoveStringLiteralDfa14_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
869. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 182:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
895. static private final int jjMoveStringLiteralDfa15_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
899. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 183:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
923. static private final int jjMoveStringLiteralDfa16_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |927. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 184:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
949. static private final int jjMoveStringLiteralDfa17_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |953. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 185:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1241. try { curChar = input_stream.readChar(); }
```

[Table of Contents] [Description]

Path 186:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1339. public static Token getNextToken() //parsemibtokenmanager.java
      |...
      |1349. try
```

[Table of Contents] [Description]

Path 187:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1339. public static Token getNextToken() //parsemibtokenmanager.java
      |...
      |1360. try { input_stream.backup(0);
```

[Table of Contents] [Description]

Path 188:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1339. public static Token getNextToken() //parsemibtokenmanager.java
      |...
      1386. try { input_stream.readChar(); input_stream.backup(1); }
```

[Table of Contents] [Description]

Path 189:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
52. static protected void ExpandBuff(boolean wrapAround) //simplecharstream.java
      |...
      58. try
```

[Table of Contents] [Description]

Path 190:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |...
      127. try {
```

[Table of Contents] [Description]

Path 191:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void collectOids() //simplenode.java
      |...
      724. try {
```

[Table of Contents] [Description]

Path 192:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
265. protected void formatPad(int numBlanks) //calloutemitter.java
      |...
```

```
271. try {
```

[Table of Contents] [Description]

Path 193:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
293. protected void addCallout (int coNum, //calloutemitter.java
     ...
332. try {
```

[Table of Contents] [Description]

Path 194:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. public static String localTime (String cvsDate) //cvs.java
     ...
75. try {
```

[Table of Contents] [Description]

Path 195:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
86. public void formatTextCallout(Emitter rtfEmitter, //formatcallout.java
     ...
99. try {
```

[Table of Contents] [Description]

Path 196:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
47. public void formatCallout(Emitter rtfEmitter, //formatgraphiccallout.java
     ...
58. try {
```

[Table of Contents] [Description]

Path 197:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
51. public void formatCallout(Emitter rtfEmitter, //formatunicodecallout.java
```

```
...
62. try {
```

[Table of Contents] [Description]

Path 198:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
110. public static int convertLength(String length) //table.java
...
146. try {
```

[Table of Contents] [Description]

Path 199:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
186. protected static String getVariable(Context context, String
varName) //table.java
...
191. try {
```

[Table of Contents] [Description]

Path 200:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
224. private static void setupColumnWidths(Context context) //table.java
...
231. try {
```

[Table of Contents] [Description]

Path 201:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
295. public static NodeSetValue adjustColumnWidths (Context context, //table.java
...
302. try {
```

[Table of Contents] [Description]

Path 202:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
295. public static NodeSetValue adjustColumnWidths (Context context, //table.java
     ...
326. try {
```

[Table of Contents] [Description]

Path 203:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
442. protected static String[] correctRoundingError(String widths[]) //table.java
     ...
446. try {
```

[Table of Contents] [Description]

Path 204:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
442. protected static String[] correctRoundingError(String widths[]) //table.java
     ...
459. try {
```

[Table of Contents] [Description]

Path 205:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
130. try {
```

[Table of Contents] [Description]

Path 206:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
138. try {
```

[Table of Contents] [Description]

Path 207:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
       ...
141. try {
```

[Table of Contents] [Description]

Path 208:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
52. public Class getExtensionClass(String localname) //textfactory.java
       ...
54. try {
```

[Table of Contents] [Description]

Path 209:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
66. protected static String getVariable(Context context, String
varName) //unwraplinks.java
       ...
70. try {
```

[Table of Contents] [Description]

Path 210:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
102. public static NodeSetValue unwrapLinks (Context context, //unwraplinks.java
       ...
110. try {
```

[Table of Contents] [Description]

Path 211:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
105. protected static String getVariable(Context context, String
varName) //verbatim.java
       ...
109. try {
```

[Table of Contents] [Description]

Path 212:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. private static void setupLineNumbering(Context context) //verbatim.java
     ...
168. try {
```

[Table of Contents] [Description]

Path 213:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. private static void setupLineNumbering(Context context) //verbatim.java
     ...
176. try {
```

[Table of Contents] [Description]

Path 214:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. private static void setupLineNumbering(Context context) //verbatim.java
     ...
184. try {
```

[Table of Contents] [Description]

Path 215:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public static NodeSetValue numberLines (Context context, //verbatim.java
     ...
245. try {
```

[Table of Contents] [Description]

Path 216:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
311. private static void setupCallouts(Context context) //verbatim.java
     ...
339. try {
```

[Table of Contents] [Description]

Path 217:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
311. private static void setupCallouts(Context context) //verbatim.java
      |...
      365. try {
```

[Table of Contents] [Description]

Path 218:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
311. private static void setupCallouts(Context context) //verbatim.java
      |...
      381. try {
```

[Table of Contents] [Description]

Path 219:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
311. private static void setupCallouts(Context context) //verbatim.java
      |...
      391. try {
```

[Table of Contents] [Description]

Path 220:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
466. public static NodeSetValue insertCallouts (Context context, //verbatim.java
      |...
      474. try {
```

[Table of Contents] [Description]

Path 221:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. public String localTime (String cvsDate) //cvs.java
      |...
      75. try {
```

[Table of Contents] [Description]

Path 222:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
76. public void formatTextCallout(DOMBuilder rtf, //formatcallout.java
      ...
89. try {
```

[Table of Contents] [Description]

Path 223:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
39. public void formatCallout(DOMBuilder rtf, //formatdingbatcallout.java
      ...
45. try {
```

[Table of Contents] [Description]

Path 224:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
43. public void formatCallout(DOMBuilder rtf, //formatgraphiccallout.java
      ...
49. try {
```

[Table of Contents] [Description]

Path 225:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
43. public void formatCallout(DOMBuilder rtf, //formatunicodecallout.java
      ...
49. try {
```

[Table of Contents] [Description]

Path 226:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
14. public static String getString(ExpressionContext context, //params.java
      ...
16. try {
```

[Table of Contents] [Description]

Path 227:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
31. public static int getInt(ExpressionContext context, //params.java
      ...
35. try {
```

[Table of Contents] [Description]

Path 228:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public static int convertLength(String length) //table.java
      ...
147. try {
```

[Table of Contents] [Description]

Path 229:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public DocumentFragment adjustColumnWidths (ExpressionContext
context, //table.java
      ...
305. try {
```

[Table of Contents] [Description]

Path 230:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public DocumentFragment adjustColumnWidths (ExpressionContext
context, //table.java
      ...
409. try {
```

[Table of Contents] [Description]

Path 231:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public DocumentFragment adjustColumnWidths (ExpressionContext
context, //table.java
      ...
419. try {
```

[Table of Contents] [Description]

Path 232:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
496. protected String[] correctRoundingError(String widths[]) //table.java
     ...
500. try {
```

[Table of Contents] [Description]

Path 233:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
496. protected String[] correctRoundingError(String widths[]) //table.java
     ...
513. try {
```

[Table of Contents] [Description]

Path 234:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
     ...
87. try {
```

[Table of Contents] [Description]

Path 235:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
     ...
90. try {
```

[Table of Contents] [Description]

Path 236:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
     ...
101. try {
```

[Table of Contents] [Description]

Path 237:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      ...
104. try {
```

[Table of Contents] [Description]

Path 238:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
158. public DocumentFragment numberLines (ExpressionContext context, //verbatim.java
       ...
171. try {
```

[Table of Contents] [Description]

Path 239:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
249. private void lineNumberFragment(DOMBuilder rtf, //verbatim.java
       ...
251. try {
```

[Table of Contents] [Description]

Path 240:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
346. private void formatLineNumber(DOMBuilder rtf, //verbatim.java
       ...
362. try {
```

[Table of Contents] [Description]

Path 241:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
511. public DocumentFragment insertCallouts (NodeIterator
areaspecNodeSet, //verbatim.java
       ...
573. try {
```

[Table of Contents] [Description]

Path 242:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
604. private void calloutFragment(DOMBuilder rtf, //verbatim.java
       ...
607. try {
```

[Table of Contents] [Description]

Path 243:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
735. private void addCallout (int coNum, //verbatim.java
       ...
781. try {
```

[Table of Contents] [Description]

Path 244:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
831. private void formatPad(DOMBuilder rtf, //verbatim.java
       ...
838. try {
```

[Table of Contents] [Description]

Path 245:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
393. public void run() //runnableconsumerthreadpool.java
       ...
398. try {
```

[Table of Contents] [Description]

Path 246:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
393. public void run() //runnableconsumerthreadpool.java
       ...
410. try {
```

[Table of Contents] [Description]

Path 247:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
612. public void start() //runnableconsumerthreadpool.java
     |613. try {
```

[Table of Contents] [Description]

Path 248:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
625. public void stop() //runnableconsumerthreadpool.java
     |...
     |634. try {
```

[Table of Contents] [Description]

Path 249:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
201. public void run() //vmtaskfiber.java
     |...
     |213. try {
```

[Table of Contents] [Description]

Path 250:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
     |...
     |243. try {
```

[Table of Contents] [Description]

Path 251:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
     |...
     |249. try {
```

[Table of Contents] [Description]

Path 252:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
      |...
      |254. try {
```

[Table of Contents] [Description]

Path 253:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
276. public static EventProxy createEventProxy() //vault.java
      |...
      |285. try {
```

[Table of Contents] [Description]

Path 254:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      |...
      |101. try {
```

[Table of Contents] [Description]

Path 255:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      |...
      |131. try {
```

[Table of Contents] [Description]

Path 256:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
115. public static InputStream post(URL url, InputStream dataStream, String
username, String password, int bufSize) throws IOException //httputils.java
      |...
      |134. try {
```

[Table of Contents] [Description]

Path 257:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
      |
      ...
      |
208. try {
```

[Table of Contents] [Description]

Path 258:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
      |
      ...
      |
235. try {
```

[Table of Contents] [Description]

Path 259:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
      |
      ...
      |
284. try {
```

[Table of Contents] [Description]

Path 260:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
      |
      ...
      |
298. try {
```

[Table of Contents] [Description]

Path 261:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
78. public static String getProperty(String name, String
defaultVal) //javamailerconfig.java
      |
      ...
      |
81. try {
```

[Table of Contents] [Description]

Path 262:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. public static int getProperty(String name, int
defaultVal) //javamailerconfig.java
      |...
      |117. try {
```

[Table of Contents] [Description]

Path 263:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
135. public static long getProperty(String name, long
defaultVal) //javamailerconfig.java
      |...
      |138. try {
```

[Table of Contents] [Description]

Path 264:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
661. public void databaseCheckLanguage() throws Exception //installer.java
      |...
      |682. try {
```

[Table of Contents] [Description]

Path 265:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
      |...
      |970. try {
```

[Table of Contents] [Description]

Path 266:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1178. void insertData() throws Exception //installer.java
      |...
```

```
1191. try {
```

[Table of Contents] [Description]

Path 267:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      ...
1656. try {
```

[Table of Contents] [Description]

Path 268:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      ...
1675. try {
```

[Table of Contents] [Description]

Path 269:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      ...
2179. try {
```

[Table of Contents] [Description]

Path 270:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
      ...
2350. try {
```

[Table of Contents] [Description]

Path 271:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
83. public void run() //processexec.java
```

```
84. try {
```

[Table of Contents] [Description]

Path 272:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
83. public void run() //processexec.java
    ...
95. try {
```

[Table of Contents] [Description]

Path 273:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public SnmpIPAddress(String inetAddr) throws
SnmpBadConversionException //snmpipaddress.java
150. try {
```

[Table of Contents] [Description]

Path 274:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
300. public InetAddress convertToIpAddress() throws
SnmpBadConversionException //snmpipaddress.java
209. if (data == null || data.length < 4)
```

[Table of Contents] [Description]

Path 275:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
161. SnmpPortal(SnmpPacketHandler handler, AsnEncoder encoder, int port) throws
SocketException //snmpportal.java
    ...
196. try {
```

[Table of Contents] [Description]

Path 276:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
239. private int setup() //snmpportal.java
```

```
...
245. try {
```

[Table of Contents] [Description]

Path 277:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
239. private int setup() //snmpportal.java
...
252. try {
```

[Table of Contents] [Description]

Path 278:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
280. public void run() //snmpportal.java
...
306. try {
```

[Table of Contents] [Description]

Path 279:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
271. public void run() //snmpportal.java
...
356. try {
```

[Table of Contents] [Description]

Path 280:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
271. public void run() //snmpportal.java
...
398. try {
```

[Table of Contents] [Description]

Path 281:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
271. public void run() //snmpportal.java
```

```
...
441. try {
```

[Table of Contents] [Description]

Path 282:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
726. void close() //snmpportal.java
...
730. try {
```

[Table of Contents] [Description]

Path 283:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
150. public void run() //snmprequest.java
...
156. try {
```

[Table of Contents] [Description]

Path 284:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public void run() //snmptimer.java
...
134. try {
```

[Table of Contents] [Description]

Path 285:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public void run() //snmptimer.java
...
157. try {
```

[Table of Contents] [Description]

Path 286:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public void run() //snmptimer.java
```

```
...
190. try {
```

[Table of Contents] [Description]

Path 287:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
119. public void run() //snmptimer.java
...
205. try {
```

[Table of Contents] [Description]

Path 288:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
264. void cancel() //snmptimer.java
...
270. try {
```

[Table of Contents] [Description]

Path 289:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
949. public int buildString(byte[] buf, int startOffset, byte asnType, byte[]
opaque) throws AsnEncodingException //berencoder.java
...
967. try {
```

[Table of Contents] [Description]

Path 290:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1002. public Object[] parseString(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
...
1022. try {
```

[Table of Contents] [Description]

Path 291:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      ...
1175. try {
```

[Table of Contents] [Description]

Path 292:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
157. try {
```

[Table of Contents] [Description]

Path 293:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
      ...
430. try {
```

[Table of Contents] [Description]

Path 294:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
459. public void closeConnection() //availabilitydata.java
      ...
462. try {
```

[Table of Contents] [Description]

Path 295:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
670. try {
```

[Table of Contents] [Description]

Path 296:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     ...
222. try {
```

[Table of Contents] [Description]

Path 297:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. public static String constructUrl(String url_, String hostname_) throws
IllegalArgumentException, NullPointerException //dbtools.java
     ...
165. try {
```

[Table of Contents] [Description]

Path 298:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
99. public synchronized void init() //actiond.java
     ...
106. try {
```

[Table of Contents] [Description]

Path 299:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. public void onEvent(Event event) //broadcasteventprocessor.java
     ...
107. try {
```

[Table of Contents] [Description]

Path 300:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. public void onEvent(Event event) //broadcasteventprocessor.java
     ...
128. try {
```

[Table of Contents] [Description]

Path 301:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
195. public void run() //executor.java
       ...
219. try {
```

[Table of Contents] [Description]

Path 302:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
195. public void run() //executor.java
       ...
250. try {
```

[Table of Contents] [Description]

Path 303:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //executor.java
       ...
401. try {
```

[Table of Contents] [Description]

Path 304:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //executor.java
       ...
425. try {
```

[Table of Contents] [Description]

Path 305:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //executor.java
       ...
438. try {
```

[Table of Contents] [Description]

Path 306:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //executor.java
      |
      ...
      459. try {
```

[Table of Contents] [Description]

Path 307:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //executor.java
      |
      ...
      467. try {
```

[Table of Contents] [Description]

Path 308:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. private void init() throws ArchiverException //eventsarchiver.java
      |
      ...
      203. try {
```

[Table of Contents] [Description]

Path 309:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. private void init() throws ArchiverException //eventsarchiver.java
      |
      ...
      210. try {
```

[Table of Contents] [Description]

Path 310:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. private void init() throws ArchiverException //eventsarchiver.java
      |
      ...
      252. try {
```

[Table of Contents] [Description]

Path 311:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
273. private boolean removeEvent(String eventID) //eventsarchiver.java
      ...
276. try {
```

[Table of Contents] [Description]

Path 312:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
432. private void close() //eventsarchiver.java
      ...
436. try {
```

[Table of Contents] [Description]

Path 313:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
460. public EventsArchiver() throws ArchiverException //eventsarchiver.java
      ...
465. try {
```

[Table of Contents] [Description]

Path 314:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
475. public static void main(String[] args) //eventsarchiver.java
      476. try {
```

[Table of Contents] [Description]

Path 315:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
107. final protected boolean checkConnection(ConnectionConfig
config) //abstracttcpplugin.java
      ...
125. try {
```

[Table of Contents] [Description]

Path 316:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
171. protected void closeSocket(Socket socket, ConnectionConfig
config) //abstracttcpplugin.java
     172. try {
```

[Table of Contents] [Description]

Path 317:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
     363. try {
```

[Table of Contents] [Description]

Path 318:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
646. private List doAddServiceToInterface(Connection dbConn, String ipaddr, String
serviceName, int serviceId, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
     650. try {
```

[Table of Contents] [Description]

Path 319:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1160. private void handleAddInterface(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
     ...
     1178. try {
```

[Table of Contents] [Description]

Path 320:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1222. private void handleAddNode(Event event) throws
```

```
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1239. try {
```

[Table of Contents] [Description]

Path 321:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1283. private void handleChangeService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1301. try {
```

[Table of Contents] [Description]

Path 322:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1346. private void handleDeleteInterface(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1365. try {
```

[Table of Contents] [Description]

Path 323:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1411. private void handleDeleteNode(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1430. try {
```

[Table of Contents] [Description]

Path 324:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1478. private void handleDeleteService(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1497. try {
```

[Table of Contents] [Description]

Path 325:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      ...
1570. try {
```

[Table of Contents] [Description]

Path 326:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      ...
1583. try {
```

[Table of Contents] [Description]

Path 327:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      ...
1588. try {
```

[Table of Contents] [Description]

Path 328:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      ...
1593. try {
```

[Table of Contents] [Description]

Path 329:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1639. private void handleNodeAdded(Event event) throws
```

```
InsufficientInformationException //broadcasteventprocessor.java
    ...
1645. try {
```

[Table of Contents] [Description]

Path 330:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1670. private void handleUpdateServer(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1692. try {
```

[Table of Contents] [Description]

Path 331:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1742. private void handleUpdateService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
    ...
1761. try {
```

[Table of Contents] [Description]

Path 332:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2182. public void onEvent(Event event) //broadcasteventprocessor.java
    ...
2185. try {
```

[Table of Contents] [Description]

Path 333:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
76. public class Capsd implements PausableFiber //capsd.java
    ...
136. try {
```

[Table of Contents] [Description]

Path 334:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
182. try {
```

[Table of Contents] [Description]

Path 335:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
197. try {
```

[Table of Contents] [Description]

Path 336:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
212. try {
```

[Table of Contents] [Description]

Path 337:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
227. try {
```

[Table of Contents] [Description]

Path 338:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
245. try {
```

[Table of Contents] [Description]

Path 339:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
      |
      ...
      274. try {
```

[Table of Contents] [Description]

Path 340:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
      |
      ...
      323. try {
```

[Table of Contents] [Description]

Path 341:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void init() //capsd.java
      |
      ...
      337. try {
```

[Table of Contents] [Description]

Path 342:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
862. void store() throws SQLException //dbifserviceentry.java
      |
      ...
      871. try {
```

[Table of Contents] [Description]

Path 343:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
933. static DbIfServiceEntry get(int nid, InetAddress addr, int sid) throws
SQLException //dbifserviceentry.java
      |
      ...
      939. try {
```

[Table of Contents] [Description]

Path 344:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
998. public static void main(String[] args) //dbifserviceentry.java
     |999. try {
```

[Table of Contents] [Description]

Path 345:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
847. void store() throws SQLException //dbipinterfaceentry.java
     |...
     |856. try {
```

[Table of Contents] [Description]

Path 346:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
885. DbIfServiceEntry[] getServices() throws SQLException //dbipinterfaceentry.java
     |...
     |893. try {
```

[Table of Contents] [Description]

Path 347:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
991. static DbIpInterfaceEntry get(int nid, InetAddress addr) throws
SQLException //dbipinterfaceentry.java
     |...
     |997. try {
```

[Table of Contents] [Description]

Path 348:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1021. static DbIpInterfaceEntry get(int nid, InetAddress addr, int ifIndex) throws
SQLException //dbipinterfaceentry.java
     |...
     |1027. try {
```

[Table of Contents] [Description]

Path 349:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1107. public static void main(String[] args) //dbipinterfaceentry.java
      |1108. try {
```

[Table of Contents] [Description]

Path 350:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1502. void store() throws SQLException //dbnodeentry.java
      |...
      |1511. try {
```

[Table of Contents] [Description]

Path 351:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1542. DbIpInterfaceEntry[] getInterfaces() throws SQLException //dbnodeentry.java
      |...
      |1550. try {
```

[Table of Contents] [Description]

Path 352:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      |...
      |1575. try {
```

[Table of Contents] [Description]

Path 353:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1599. DbIpInterfaceEntry[] getManagedInterfaces() throws
SQLException //dbnodeentry.java
      |...
      |1607. try {
```

[Table of Contents] [Description]

Path 354:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1632. try {
```

[Table of Contents] [Description]

Path 355:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1667. DbSnmpInterfaceEntry[] getSnmpInterfaces() throws
SQLException //dbnodeentry.java
    ...
1675. try {
```

[Table of Contents] [Description]

Path 356:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
    ...
1700. try {
```

[Table of Contents] [Description]

Path 357:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1805. static DbNodeEntry get(int nid, String dpName) throws
SQLException //dbnodeentry.java
    ...
1811. try {
```

[Table of Contents] [Description]

Path 358:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
```

```
496. try {
```

[Table of Contents] [Description]

Path 359:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
507. try {
```

[Table of Contents] [Description]

Path 360:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
...
933. try {
```

[Table of Contents] [Description]

Path 361:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
990. static DbSnmpInterfaceEntry get(int nid, int ifIndex) throws
SQLException //dbsnmpinterfaceentry.java
...
996. try {
```

[Table of Contents] [Description]

Path 362:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1055. public static void main(String[] args) //dbsnmpinterfaceentry.java
1056. try {
```

[Table of Contents] [Description]

Path 363:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
102. private boolean isServer(InetAddress host, int retries, int
```

```
timeout) //dhcpplugin.java
    ...
110. try {
```

[Table of Contents] [Description]

Path 364:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
    ...
110. try {
```

[Table of Contents] [Description]

Path 365:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
    ...
121. try {
```

[Table of Contents] [Description]

Path 366:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
    ...
141. try {
```

[Table of Contents] [Description]

Path 367:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
420. public static long getLongParm(Event e, String parmName, long
defaultValue) //eventutils.java
    ...
426. try {
```

[Table of Contents] [Description]

Path 368:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
538. public static void sendEvent(Event newEvent, String callerUei, long txNo,
boolean isXmlRpcEnabled) //eventutils.java
     ...
541. try {
```

[Table of Contents] [Description]

Path 369:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public final class FtpPlugin extends AbstractTcpPlugin //ftpplugin.java
     ...
105. try {
```

[Table of Contents] [Description]

Path 370:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
130. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //ftpplugin.java
     ...
136. try {
```

[Table of Contents] [Description]

Path 371:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
230. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //gpplugin.java
     ...
261. try {
```

[Table of Contents] [Description]

Path 372:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
196. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //httpplugin.java
     ...
204. try {
```

[Table of Contents] [Description]

Path 373:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
196. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //httpplugin.java
      |...
      |211. try
```

[Table of Contents] [Description]

Path 374:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
232. private boolean isPingable(InetAddress ipv4Addr, int retries, long
timeout) //icmpplugin.java
      |...
      |252. try {
```

[Table of Contents] [Description]

Path 375:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
232. private boolean isPingable(InetAddress ipv4Addr, int retries, long
timeout) //icmpplugin.java
      |...
      |262. try {
```

[Table of Contents] [Description]

Path 376:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. private static void probe(InetAddress target, List supports) //ifcollector.java
      |...
      |197. try {
```

[Table of Contents] [Description]

Path 377:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |...
      |390. try {
```

[Table of Contents] [Description]

Path 378:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
408. try {
```

[Table of Contents] [Description]

Path 379:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
439. try {
```

[Table of Contents] [Description]

Path 380:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
293. try {
```

[Table of Contents] [Description]

Path 381:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
319. try {
```

[Table of Contents] [Description]

Path 382:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
335. try {
```

[Table of Contents] [Description]

Path 383:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
378. try {
```

[Table of Contents] [Description]

Path 384:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
217. InetAddress getMask(int ifIndex) //ifsnmpcollector.java
      ...
232. try {
```

[Table of Contents] [Description]

Path 385:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
255. InetAddress[] getIfAddressAndMask(int ifIndex) //ifsnmpcollector.java
      ...
271. try {
```

[Table of Contents] [Description]

Path 386:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
328. int getIfIndex(InetAddress address) //ifsnmpcollector.java
      ...
345. try {
```

[Table of Contents] [Description]

Path 387:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
      ...
466. try {
```

[Table of Contents] [Description]

Path 388:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
      ...
476. try {
```

[Table of Contents] [Description]

Path 389:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
      ...
506. try {
```

[Table of Contents] [Description]

Path 390:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
109. private boolean isServer( String hostname, Map qualifiers ) //jdbcplugin.java
      ...
127. try
```

[Table of Contents] [Description]

Path 391:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
109. private boolean isServer( String hostname, Map qualifiers ) //jdbcplugin.java
      ...
180. try
```

[Table of Contents] [Description]

Path 392:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
193. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcplugin.java
      ...
```

```
197. try
```

[Table of Contents] [Description]

Path 393:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
193. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcplugin.java
...
221. try
```

[Table of Contents] [Description]

Path 394:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
...
88. try
```

[Table of Contents] [Description]

Path 395:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
...
109. try
```

[Table of Contents] [Description]

Path 396:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
294. static synchronized void dataSourceSync() throws SQLException //knownipmgr.java
...
319. try {
```

[Table of Contents] [Description]

Path 397:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
294. static synchronized void dataSourceSync() throws SQLException //knownipmgr.java
        ...
340. try {
```

[Table of Contents] [Description]

Path 398:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
        ...
153. try {
```

[Table of Contents] [Description]

Path 399:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
        ...
165. try {
```

[Table of Contents] [Description]

Path 400:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
        ...
172. try {
```

[Table of Contents] [Description]

Path 401:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
        ...
194. try {
```

[Table of Contents] [Description]

Path 402:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //msexchangeplugin.java
     |...
     |135. try {
```

[Table of Contents] [Description]

Path 403:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //msexchangeplugin.java
     |...
     |172. try {
```

[Table of Contents] [Description]

Path 404:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, String command, int
padding, int retries, int timeout, RE regex, StringBuffer
bannerResult) //nrpeplugin.java
     |...
     |120. try {
```

[Table of Contents] [Description]

Path 405:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, String command, int
padding, int retries, int timeout, RE regex, StringBuffer
bannerResult) //nrpeplugin.java
     |...
     |203. try {
```

[Table of Contents] [Description]

Path 406:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
258. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //nrpeplugin.java
     |...
```

```
278. try {
```

[Table of Contents] [Description]

Path 407:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
214. private NsclientPacket isServer(InetAddress host, int
port, //nsclientplugin.java
     ...
223. try {
```

[Table of Contents] [Description]

Path 408:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout) //ntpplugin.java
     ...
104. try {
```

[Table of Contents] [Description]

Path 409:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout) //ntpplugin.java
     ...
115. try {
```

[Table of Contents] [Description]

Path 410:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
     ...
125. try {
```

[Table of Contents] [Description]

Path 411:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
      |...
      |179. try {
```

[Table of Contents] [Description]

Path 412:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
134. private boolean isRadius(InetAddress host, int authport, int acctport, String
authType, //radiusauthplugin.java
      |...
      |143. try {
```

[Table of Contents] [Description]

Path 413:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
134. private boolean isRadius(InetAddress host, int authport, int acctport, String
authType, //radiusauthplugin.java
      |...
      |154. try {
```

[Table of Contents] [Description]

Path 414:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
      |...
      |370. try {
```

[Table of Contents] [Description]

Path 415:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      |...
      |532. try {
```

[Table of Contents] [Description]

Path 416:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      ...
546. try {
```

[Table of Contents] [Description]

Path 417:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      ...
618. try {
```

[Table of Contents] [Description]

Path 418:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
708. private synchronized void sendInterfaceReparentedEvent(String ipAddr, String
ipHostName, int newNodeId, int oldNodeId, DbNodeEntry
reparentNodeEntry) //reparentviasmb.java
      ...
798. try {
```

[Table of Contents] [Description]

Path 419:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
      ...
572. try {
```

[Table of Contents] [Description]

Path 420:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
```

```
845. try {
```

[Table of Contents] [Description]

Path 421:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
905. try {
```

[Table of Contents] [Description]

Path 422:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
963. private void deleteDuplicateNode(Connection dbc, DbNodeEntry duplicateNode)
throws SQLException //rescanprocessor.java
    ...
971. try {
```

[Table of Contents] [Description]

Path 423:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
963. private void deleteDuplicateNode(Connection dbc, DbNodeEntry duplicateNode)
throws SQLException //rescanprocessor.java
    ...
986. try {
```

[Table of Contents] [Description]

Path 424:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1009. private boolean isDuplicateInterface(Connection dbc, InetAddress ifaddr, int
nodeId) throws SQLException //rescanprocessor.java
    ...
1015. try {
```

[Table of Contents] [Description]

Path 425:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1009. private boolean isDuplicateInterface(Connection dbc, InetAddress ifaddr, int
nodeId) throws SQLException //rescanprocessor.java
      ...
1029. try {
```

[Table of Contents] [Description]

Path 426:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1221. try {
```

[Table of Contents] [Description]

Path 427:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1330. try {
```

[Table of Contents] [Description]

Path 428:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1408. try {
```

[Table of Contents] [Description]

Path 429:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
```

```
...
1425. try {
```

[Table of Contents] [Description]

Path 430:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
...
1847. try {
```

[Table of Contents] [Description]

Path 431:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
...
1941. try {
```

[Table of Contents] [Description]

Path 432:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
1992. try {
```

[Table of Contents] [Description]

Path 433:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
2123. try {
```

[Table of Contents] [Description]

Path 434:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      2900. try {
```

[Table of Contents] [Description]

Path 435:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      2945. try {
```

[Table of Contents] [Description]

Path 436:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      2960. try {
```

[Table of Contents] [Description]

Path 437:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
3009. private InetAddress updatePrimarySnmpInterface(Connection dbc, DbNodeEntry
dbNodeEntry, Map collectorMap, InetAddress oldPriIf) throws
SQLException //rescanprocessor.java
      |...
      3041. try {
```

[Table of Contents] [Description]

Path 438:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
3113. private DbIpInterfaceEntry[] getInterfaces(DbNodeEntry
dbNodeEntry) //rescanprocessor.java
      |...
      3127. try {
```

[Table of Contents] [Description]

Path 439:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
3139. private DbNodeEntry getNode() //rescanprocessor.java
      ...
3146. try {
```

[Table of Contents] [Description]

Path 440:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
373. void forceRescan(int nodeId) //scheduler.java
      ...
375. try {
```

[Table of Contents] [Description]

Path 441:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void run() //scheduler.java
      ...
524. try {
```

[Table of Contents] [Description]

Path 442:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void run() //scheduler.java
      ...
559. try {
```

[Table of Contents] [Description]

Path 443:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void run() //scheduler.java
      ...
570. try {
```

[Table of Contents] [Description]

Path 444:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void run() //scheduler.java
       ...
574. try {
```

[Table of Contents] [Description]

Path 445:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
492. public void run() //scheduler.java
       ...
626. try {
```

[Table of Contents] [Description]

Path 446:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
74. private boolean isSmb(InetAddress host) //smbplugin.java
       ...
77. try {
```

[Table of Contents] [Description]

Path 447:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
79. public final class SmtpPlugin extends AbstractPlugin //smtpplugin.java
       ...
119. try {
```

[Table of Contents] [Description]

Path 448:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
79. public final class SmtpPlugin extends AbstractPlugin //smtpplugin.java
       ...
126. try {
```

[Table of Contents] [Description]

Path 449:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      ...
157. try {
```

[Table of Contents] [Description]

Path 450:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      ...
285. try {
```

[Table of Contents] [Description]

Path 451:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. public boolean isProtocolSupported(InetAddress address) //snmpplugin.java
      ...
100. try {
```

[Table of Contents] [Description]

Path 452:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. public boolean isProtocolSupported(InetAddress address) //snmpplugin.java
      ...
110. try {
```

[Table of Contents] [Description]

Path 453:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
      ...
148. try {
```

[Table of Contents] [Description]

Path 454:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |...
208. try {
```

[Table of Contents] [Description]

Path 455:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     |...
120. try {
```

[Table of Contents] [Description]

Path 456:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     |...
160. try {
```

[Table of Contents] [Description]

Path 457:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     |...
197. try {
```

[Table of Contents] [Description]

Path 458:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
252. public boolean isProtocolSupported(InetAddress address, Map
```

```
qualifiers) //sshplugin.java
    ...
273. try {
```

[Table of Contents] [Description]

Path 459:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
267. try {
```

[Table of Contents] [Description]

Path 460:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
292. try {
```

[Table of Contents] [Description]

Path 461:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
    ...
699. try {
```

[Table of Contents] [Description]

Path 462:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
    ...
751. try {
```

[Table of Contents] [Description]

Path 463:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
     ...
928. try {
```

[Table of Contents] [Description]

Path 464:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
     ...
1257. try {
```

[Table of Contents] [Description]

Path 465:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
     ...
1281. try {
```

[Table of Contents] [Description]

Path 466:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
     ...
1343. try {
```

[Table of Contents] [Description]

Path 467:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
     ...
1408. try {
```

[Table of Contents] [Description]

Path 468:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
throws SQLException //suspecteventprocessor.java
      ...
1470. try {
```

[Table of Contents] [Description]

Path 469:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
throws SQLException //suspecteventprocessor.java
      ...
1476. try {
```

[Table of Contents] [Description]

Path 470:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1542. try {
```

[Table of Contents] [Description]

Path 471:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1571. try {
```

[Table of Contents] [Description]

Path 472:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
      ...
1721. try {
```

[Table of Contents] [Description]

Path 473:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1740. private void createAndSendReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //suspecteventprocessor.java
      ...
1761. try {
```

[Table of Contents] [Description]

Path 474:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1848. private void createAndSendNodeAddedEvent(DbNodeEntry
nodeEntry) //suspecteventprocessor.java
      ...
1898. try {
```

[Table of Contents] [Description]

Path 475:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1918. private void createAndSendDuplicateIpaddressEvent(int nodeId, String
ipAddr) //suspecteventprocessor.java
      ...
1941. try {
```

[Table of Contents] [Description]

Path 476:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1918. private void createAndSendDuplicateIpaddressEvent(int nodeId, String
ipAddr) //suspecteventprocessor.java
      ...
1962. try {
```

[Table of Contents] [Description]

Path 477:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1981. private void createAndSendNodeGainedInterfaceEvent(int nodeId, InetAddress
ipAddr) //suspecteventprocessor.java
      |...
      2024. try {
```

[Table of Contents] [Description]

Path 478:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2048. private void createAndSendNodeGainedServiceEvent(DbNodeEntry nodeEntry,
InetAddress ipAddr, String svcName, String qualifier) //suspecteventprocessor.java
      |...
      2132. try {
```

[Table of Contents] [Description]

Path 479:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      |...
      121. try {
```

[Table of Contents] [Description]

Path 480:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      |...
      185. try {
```

[Table of Contents] [Description]

Path 481:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
240. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //tcpplugin.java
      |...
```

```
261. try {
```

[Table of Contents] [Description]

Path 482:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
...
451. try {
```

[Table of Contents] [Description]

Path 483:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public void update(SnmpVarBind[] vars) //iftableentry.java
...
269. try {
```

[Table of Contents] [Description]

Path 484:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public void update(SnmpVarBind[] vars) //ifxtableentry.java
...
271. try {
```

[Table of Contents] [Description]

Path 485:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
469. public static InetAddress getIpAddress(List ipAddrEntries, int
ifIndex) //ipaddrtable.java
...
496. try {
```

[Table of Contents] [Description]

Path 486:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
525. public static List getIpAddresses(List ipAddrEntries, int
ifIndex) //ipaddrtable.java
      ...
      542. try {
```

[Table of Contents] [Description]

Path 487:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
564. public static List getIpAddresses(List ipAddrEntries) //ipaddrtable.java
      ...
      581. try {
```

[Table of Contents] [Description]

Path 488:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
185. public void update(SnmpVarBind[] vars) //ipaddrtableentry.java
      ...
      201. try {
```

[Table of Contents] [Description]

Path 489:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
284. public void update(SnmpVarBind[] vars) //systemgroup.java
      ...
      293. try {
```

[Table of Contents] [Description]

Path 490:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
288. private void sendEvent(String uei) //collectableservice.java
      ...
      296. try {
```

[Table of Contents] [Description]

Path 491:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
351. try {
```

[Table of Contents] [Description]

Path 492:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
476. try {
```

[Table of Contents] [Description]

Path 493:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
535. try {
```

[Table of Contents] [Description]

Path 494:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
562. try {
```

[Table of Contents] [Description]

Path 495:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
578. try {
```

[Table of Contents] [Description]

Path 496:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
592. try {
```

[Table of Contents] [Description]

Path 497:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. private void createEventProcessor(final Category log) //collectd.java
       ...
181. try {
```

[Table of Contents] [Description]

Path 498:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
203. public void run() //collectd.java
       ...
205. try {
```

[Table of Contents] [Description]

Path 499:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
219. private void createScheduler(final Category log, CollectdConfiguration
config) //collectd.java
       ...
222. try {
```

[Table of Contents] [Description]

Path 500:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
       ...
244. try {
```

[Table of Contents] [Description]

Path 501:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
299. private void instantiateCollectors(final Category log, CollectdConfiguration
config) //collectd.java
      |...
      |307. try {
```

[Table of Contents] [Description]

Path 502:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
      |...
      |331. try {
```

[Table of Contents] [Description]

Path 503:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      |...
      |351. try {
```

[Table of Contents] [Description]

Path 504:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      |...
      |385. try {
```

[Table of Contents] [Description]

Path 505:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
      |...
      |525. try {
```

[Table of Contents] [Description]

Path 506:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
       ...
557. try {
```

[Table of Contents] [Description]

Path 507:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
       ...
630. try {
```

[Table of Contents] [Description]

Path 508:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
255. public boolean equals(Object object) //mibobject.java
       ...
261. try {
```

[Table of Contents] [Description]

Path 509:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
       ...
258. try {
```

[Table of Contents] [Description]

Path 510:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
       ...
268. try {
```

[Table of Contents] [Description]

Path 511:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |...
      |286. try {
```

[Table of Contents] [Description]

Path 512:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |...
      |306. try {
```

[Table of Contents] [Description]

Path 513:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |...
      |271. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 514:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |...
      |362. try {
```

[Table of Contents] [Description]

Path 515:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |457. try {
```

[Table of Contents] [Description]

Path 516:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     |...
     |486. try {
```

[Table of Contents] [Description]

Path 517:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     |...
     |542. try {
```

[Table of Contents] [Description]

Path 518:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     |...
     |604. try {
```

[Table of Contents] [Description]

Path 519:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     |...
     |647. try {
```

[Table of Contents] [Description]

Path 520:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     |...
```

```
821. try {
```

[Table of Contents] [Description]

Path 521:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
932. try {
```

[Table of Contents] [Description]

Path 522:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1011. try {
```

[Table of Contents] [Description]

Path 523:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1020. try {
```

[Table of Contents] [Description]

Path 524:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1076. try {
```

[Table of Contents] [Description]

Path 525:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
1090. try {
```

[Table of Contents] [Description]

Path 526:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
1149. try {
```

[Table of Contents] [Description]

Path 527:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
1163. try {
```

[Table of Contents] [Description]

Path 528:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1324. try {
```

[Table of Contents] [Description]

Path 529:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1421. try {
```

[Table of Contents] [Description]

Path 530:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
      ...
1569. try {
```

[Table of Contents] [Description]

Path 531:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1607. try {
```

[Table of Contents] [Description]

Path 532:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1627. try {
```

[Table of Contents] [Description]

Path 533:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1637. try {
```

[Table of Contents] [Description]

Path 534:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1684. try {
```

[Table of Contents] [Description]

Path 535:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1711. try {
```

[Table of Contents] [Description]

Path 536:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1717. try {
```

[Table of Contents] [Description]

Path 537:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1726. try {
```

[Table of Contents] [Description]

Path 538:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
      ...
159. try {
```

[Table of Contents] [Description]

Path 539:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
      ...
183. try {
```

[Table of Contents] [Description]

Path 540:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
431. try {
```

[Table of Contents] [Description]

Path 541:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
543. try {
```

[Table of Contents] [Description]

Path 542:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
614. try {
```

[Table of Contents] [Description]

Path 543:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
      ...
328. try {
```

[Table of Contents] [Description]

Path 544:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      ...
407. try {
```

[Table of Contents] [Description]

Path 545:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
      |...
      94. try {
```

[Table of Contents] [Description]

Path 546:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
      |...
      111. try {
```

[Table of Contents] [Description]

Path 547:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
13. public C3P0DbConnectionFactory(DbConfiguration dbConfig) throws
ClassNotFoundException //c3p0dbconnectionfactory.java
      |...
      15. try {
```

[Table of Contents] [Description]

Path 548:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
497. private CapsdConfigFactory(String configFile) throws IOException,
MarshalException, ValidationException //capsdconfigfactory.java
      |...
      509. try {
```

[Table of Contents] [Description]

Path 549:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
```

```
880. try {
```

[Table of Contents] [Description]

Path 550:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1164. try {
```

[Table of Contents] [Description]

Path 551:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
...
1319. try {
```

[Table of Contents] [Description]

Path 552:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
...
1337. try {
```

[Table of Contents] [Description]

Path 553:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
...
1345. try {
```

[Table of Contents] [Description]

Path 554:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |...
      1494. try {
```

[Table of Contents] [Description]

Path 555:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |...
      1514. try {
```

[Table of Contents] [Description]

Path 556:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |...
      1548. try {
```

[Table of Contents] [Description]

Path 557:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1589. private List getAddressesFromURL(String url) //capsdconfigfactory.java
      |...
      1592. try {
```

[Table of Contents] [Description]

Path 558:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
      |...
      1661. try {
```

[Table of Contents] [Description]

Path 559:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
164. private void createPackageIpListMap() //collectdconfigfactory.java
     ...
182. try {
```

[Table of Contents] [Description]

Path 560:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
     ...
215. try {
```

[Table of Contents] [Description]

Path 561:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
272. public synchronized boolean isClosed() throws
SQLException //legacydbconnectionfactory.java
273. try {
```

[Table of Contents] [Description]

Path 562:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
292. public Statement createStatement() throws
SQLException //legacydbconnectionfactory.java
     ...
294. try {
```

[Table of Contents] [Description]

Path 563:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
310. public PreparedStatement prepareStatement(String sql) throws
SQLException //legacydbconnectionfactory.java
212. if (!m_inUse)
```

[Table of Contents] [Description]

Path 564:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
328. public PreparedStatement prepareStatement(String sql, int autoGeneratedKeys)
throws SQLException //legacydbconnectionfactory.java
    ...
330. try {
```

[Table of Contents] [Description]

Path 565:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
346. public PreparedStatement prepareStatement(String sql, int[] columnIndexes)
throws SQLException //legacydbconnectionfactory.java
    ...
348. try {
```

[Table of Contents] [Description]

Path 566:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
364. public PreparedStatement prepareStatement(String sql, String[] columnNames)
throws SQLException //legacydbconnectionfactory.java
    ...
366. try {
```

[Table of Contents] [Description]

Path 567:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public CallableStatement prepareCall(String sql) throws
SQLException //legacydbconnectionfactory.java
    ...
384. try {
```

[Table of Contents] [Description]

Path 568:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
400. public String nativeSQL(String sql) throws
SQLException //legacydbconnectionfactory.java
    ...
```

```
402. try {
```

[Table of Contents] [Description]

Path 569:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
418. public void setAutoCommit(boolean autoCommit) throws
SQLException //legacydbconnectionfactory.java
      ...
420. try {
```

[Table of Contents] [Description]

Path 570:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
436. public boolean getAutoCommit() throws
SQLException //legacydbconnectionfactory.java
      ...
438. try {
```

[Table of Contents] [Description]

Path 571:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
454. public void commit() throws SQLException //legacydbconnectionfactory.java
      ...
456. try {
```

[Table of Contents] [Description]

Path 572:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
472. public void rollback() throws SQLException //legacydbconnectionfactory.java
      ...
474. try {
```

[Table of Contents] [Description]

Path 573:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
483. public void rollback(Savepoint savepoint) throws
SQLException //legacydbconnectionfactory.java
      ...
      485. try {
```

[Table of Contents] [Description]

Path 574:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. public DatabaseMetaData getMetaData() throws
SQLException //legacydbconnectionfactory.java
      ...
      503. try {
```

[Table of Contents] [Description]

Path 575:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
519. public void setReadOnly(boolean readOnly) throws
SQLException //legacydbconnectionfactory.java
      ...
      521. try {
```

[Table of Contents] [Description]

Path 576:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
537. public boolean isReadOnly() throws
SQLException //legacydbconnectionfactory.java
      ...
      539. try {
```

[Table of Contents] [Description]

Path 577:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
555. public void setCatalog(String catalog) throws
SQLException //legacydbconnectionfactory.java
      ...
      557. try {
```

[Table of Contents] [Description]

Path 578:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
573. public String getCatalog() throws SQLException //legacydbconnectionfactory.java
       ...
575. try {
```

[Table of Contents] [Description]

Path 579:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
591. public void setTransactionIsolation(int level) throws
SQLException //legacydbconnectionfactory.java
       ...
593. try {
```

[Table of Contents] [Description]

Path 580:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
609. public int getTransactionIsolation() throws
SQLException //legacydbconnectionfactory.java
       ...
611. try {
```

[Table of Contents] [Description]

Path 581:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
627. public SQLWarning getWarnings() throws
SQLException //legacydbconnectionfactory.java
       ...
629. try {
```

[Table of Contents] [Description]

Path 582:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
645. public void clearWarnings() throws
SQLException //legacydbconnectionfactory.java
       ...
647. try {
```

[Table of Contents] [Description]

Path 583:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
663. public Statement createStatement(int resultSetType, int resultSetConcurrency)
throws SQLException //legacydbconnectionfactory.java
     |...
     665. try {
```

[Table of Contents] [Description]

Path 584:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
681. public Statement createStatement(int resultSetType, int resultSetConcurrency,
int resultSetHoldability) throws SQLException //legacydbconnectionfactory.java
     |...
     683. try {
```

[Table of Contents] [Description]

Path 585:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
699. public PreparedStatement prepareStatement(String sql, int resultSetType, int
resultSetConcurrency) throws SQLException //legacydbconnectionfactory.java
     |...
     701. try {
```

[Table of Contents] [Description]

Path 586:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
717. public PreparedStatement prepareStatement(String sql, int resultSetType, int
resultSetConcurrency, int resultSetHoldability) throws
SQLException //legacydbconnectionfactory.java
     |...
     719. try {
```

[Table of Contents] [Description]

Path 587:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
735. public CallableStatement prepareCall(String sql, int resultSetType, int
resultSetConcurrency) throws SQLException //legacydbconnectionfactory.java
     ...
737. try {
```

[Table of Contents] [Description]

Path 588:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
753. public CallableStatement prepareCall(String sql, int resultSetType, int
resultSetConcurrency, int resultSetHoldability) throws
SQLException //legacydbconnectionfactory.java
     ...
755. try {
```

[Table of Contents] [Description]

Path 589:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
771. public Map getTypeMap() throws SQLException //legacydbconnectionfactory.java
     ...
773. try {
```

[Table of Contents] [Description]

Path 590:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
789. public void setTypeMap(Map map) throws
SQLException //legacydbconnectionfactory.java
     ...
791. try {
```

[Table of Contents] [Description]

Path 591:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
804. public int getHoldability() throws
SQLException //legacydbconnectionfactory.java
805. try {
```

[Table of Contents] [Description]

Path 592:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
814. public void setHoldability(int holdability) throws
SQLException //legacydbconnectionfactory.java
     |815. try {
```

[Table of Contents] [Description]

Path 593:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
824. public Savepoint setSavepoint() throws
SQLException //legacydbconnectionfactory.java
     |825. try {
```

[Table of Contents] [Description]

Path 594:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
834. public Savepoint setSavepoint(String name) throws
SQLException //legacydbconnectionfactory.java
     |835. try {
```

[Table of Contents] [Description]

Path 595:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
844. public void releaseSavepoint(Savepoint savepoint) throws
SQLException //legacydbconnectionfactory.java
     |845. try {
```

[Table of Contents] [Description]

Path 596:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     |...
     |915. try {
```

[Table of Contents] [Description]

Path 597:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
54. public MergeableSpecific(String specific) //mergeablespecific.java
      ...
57. try {
```

[Table of Contents] [Description]

Path 598:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
165. protected void sendEvent(String uei) //notifdconfigmanager.java
      ...
172. try {
```

[Table of Contents] [Description]

Path 599:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      ...
177. try {
```

[Table of Contents] [Description]

Path 600:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      ...
236. try {
```

[Table of Contents] [Description]

Path 601:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      ...
241. try {
```

[Table of Contents] [Description]

Path 602:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      |...
      |255. try {
```

[Table of Contents] [Description]

Path 603:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
292. public synchronized int getNoticeId() throws SQLException, IOException,
MarshalException, ValidationException //notificationmanager.java
      |...
      |310. try {
```

[Table of Contents] [Description]

Path 604:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
322. public boolean noticeOutstanding(int noticeId) throws IOException,
MarshalException, ValidationException //notificationmanager.java
      |...
      |326. try {
```

[Table of Contents] [Description]

Path 605:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
322. public boolean noticeOutstanding(int noticeId) throws IOException,
MarshalException, ValidationException //notificationmanager.java
      |...
      |352. try {
```

[Table of Contents] [Description]

Path 606:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
364. public Collection acknowledgeNotice(Event event, String uei, String[]
matchList) throws SQLException, IOException, MarshalException,
ValidationException //notificationmanager.java
```

```
...
450. try {
```

[Table of Contents] [Description]

Path 607:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
460. public List getActiveNodes() throws SQLException //notificationmanager.java
...
483. try {
```

[Table of Contents] [Description]

Path 608:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
493. public String getServiceNoticeStatus(String nodeID, String ipaddr, String
service) throws SQLException //notificationmanager.java
...
517. try {
```

[Table of Contents] [Description]

Path 609:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
527. public void updateNoticeWithUserInfo(String userId, int noticeId, String media,
String contactInfo, String autoNotify) throws
SQLException //notificationmanager.java
...
549. try {
```

[Table of Contents] [Description]

Path 610:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
561. public void insertNotice(int notifyId, Map params, String queueID) throws
SQLException //notificationmanager.java
...
619. try {
```

[Table of Contents] [Description]

Path 611:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
634. private int getServiceId(String service) throws
SQLException //notificationmanager.java
      ...
655. try {
```

[Table of Contents] [Description]

Path 612:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java
      ...
696. try {
```

[Table of Contents] [Description]

Path 613:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
      ...
415. try {
```

[Table of Contents] [Description]

Path 614:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
785. private void createServiceMonitors() //pollerconfigfactory.java
      ...
797. try {
```

[Table of Contents] [Description]

Path 615:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. private void setOutCalTime(Calendar outCal, String
timeStr) //polloutagesconfigmanager.java
      ...
123. try {
```

[Table of Contents] [Description]

Path 616:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. private void setOutCalTime(Calendar outCal, String
timeStr) //polloutagesconfigmanager.java
      |...
      |146. try {
```

[Table of Contents] [Description]

Path 617:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
52. public int compare(Object rng1, Object rng2) //rangecomparator.java
      |...
      |54. try {
```

[Table of Contents] [Description]

Path 618:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
      |...
      |129. try {
```

[Table of Contents] [Description]

Path 619:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
84. public SnmpEventInfo(Event event) //snmpeventinfo.java
      |...
      |111. try {
```

[Table of Contents] [Description]

Path 620:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
251. private int computeIntValue(String parmContent) throws
IllegalArgumentException //snmpeventinfo.java
      |...
      |253. try {
```

[Table of Contents] [Description]

Path 621:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public void testInvalidRange() throws MarshalException, ValidationException,
UnknownHostException //snmpeventinfotest.java
     ...
444. try {
```

[Table of Contents] [Description]

Path 622:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
207. public static synchronized String marshallConfig() //snmppeerfactory.java
     ...
212. try {
```

[Table of Contents] [Description]

Path 623:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
207. public static synchronized String marshallConfig() //snmppeerfactory.java
     ...
221. try {
```

[Table of Contents] [Description]

Path 624:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
     ...
451. try {
```

[Table of Contents] [Description]

Path 625:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
     ...
470. try {
```

[Table of Contents] [Description]

Path 626:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
56. public int compare(Object spec1, Object spec2) //specificcomparator.java
      ...
58. try {
```

[Table of Contents] [Description]

Path 627:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
150. private void createPackageIpListMap() //threshdconfigfactory.java
       ...
166. try {
```

[Table of Contents] [Description]

Path 628:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
603. public String encryptedPassword(String aPassword) //usermanager.java
       ...
606. try {
```

[Table of Contents] [Description]

Path 629:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. private String stripWhiteSpace(String comment) //viewfactory.java
       ...
406. try {
```

[Table of Contents] [Description]

Path 630:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public static synchronized void init() throws IOException, MarshalException,
ValidationException //vulnscandconfigfactory.java
       ...
189. try {
```

[Table of Contents] [Description]

Path 631:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
419. public Set getAllIpAddresses(ScanLevel level) //vulnscandconfigfactory.java
      ...
426. try {
```

[Table of Contents] [Description]

Path 632:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
419. public Set getAllIpAddresses(ScanLevel level) //vulnscandconfigfactory.java
      ...
441. try {
```

[Table of Contents] [Description]

Path 633:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
451. public Set getAllExcludes() //vulnscandconfigfactory.java
      ...
464. try {
```

[Table of Contents] [Description]

Path 634:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
534. public InetAddress getServerAddress() //vulnscandconfigfactory.java
535. try {
```

[Table of Contents] [Description]

Path 635:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
546. public String[] getPluginLists() //vulnscandconfigfactory.java
      ...
553. try {
```

[Table of Contents] [Description]

Path 636:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
577. public boolean[] getSafeChecks() //vulnscandconfigfactory.java
       ...
584. try {
```

[Table of Contents] [Description]

Path 637:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
58. final class Client extends Observable implements Runnable, Fiber //client.java
       ...
80. try {
```

[Table of Contents] [Description]

Path 638:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public void run() //client.java
       ...
133. try {
```

[Table of Contents] [Description]

Path 639:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public void run() //client.java
       ...
160. try {
```

[Table of Contents] [Description]

Path 640:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public synchronized void stop() //client.java
       ...
227. try {
```

[Table of Contents] [Description]

Path 641:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public void run() //client.java
       |...
252. try {
```

[Table of Contents] [Description]

Path 642:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public void run() //client.java
       |...
270. try {
```

[Table of Contents] [Description]

Path 643:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public void run() //client.java
       |...
282. try {
```

[Table of Contents] [Description]

Path 644:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public void run() //client.java
       |...
313. try {
```

[Table of Contents] [Description]

Path 645:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public void run() //client.java
       |...
327. try {
```

[Table of Contents] [Description]

Path 646:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
205. try {
```

[Table of Contents] [Description]

Path 647:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
221. try {
```

[Table of Contents] [Description]

Path 648:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
243. try {
```

[Table of Contents] [Description]

Path 649:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
249. try {
```

[Table of Contents] [Description]

Path 650:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
258. try {
```

[Table of Contents] [Description]

Path 651:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
264. try {
```

[Table of Contents] [Description]

Path 652:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
281. public synchronized void stop() //dhcpd.java
       ...
295. try {
```

[Table of Contents] [Description]

Path 653:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
353. public void run() //dhcpd.java
       ...
374. try {
```

[Table of Contents] [Description]

Path 654:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
353. public void run() //dhcpd.java
       ...
381. try {
```

[Table of Contents] [Description]

Path 655:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
353. public void run() //dhcpd.java
       ...
391. try {
```

[Table of Contents] [Description]

Path 656:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
353. public void run() //dhcpd.java
      |...
      |402. try {
```

[Table of Contents] [Description]

Path 657:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
      |...
      |235. try {
```

[Table of Contents] [Description]

Path 658:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
      |...
      |252. try {
```

[Table of Contents] [Description]

Path 659:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
      |...
      |263. try {
```

[Table of Contents] [Description]

Path 660:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
280. public void close() //poller.java
      |...
      |282. try {
```

[Table of Contents] [Description]

Path 661:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      403. try {
```

[Table of Contents] [Description]

Path 662:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      420. try {
```

[Table of Contents] [Description]

Path 663:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
501. private static void setHwAddress(String hwAddressStr) //poller.java
      |...
      515. try {
```

[Table of Contents] [Description]

Path 664:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public void run() //receiver.java
      |...
      114. try {
```

[Table of Contents] [Description]

Path 665:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public void run() //receiver.java
      |...
      128. try {
```

[Table of Contents] [Description]

Path 666:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public void run() //receiver2.java
       ...
114. try {
```

[Table of Contents] [Description]

Path 667:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public void run() //receiver2.java
       ...
128. try {
```

[Table of Contents] [Description]

Path 668:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
110. try {
```

[Table of Contents] [Description]

Path 669:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
119. try {
```

[Table of Contents] [Description]

Path 670:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
124. try {
```

[Table of Contents] [Description]

Path 671:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
130. try {
```

[Table of Contents] [Description]

Path 672:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
179. try {
```

[Table of Contents] [Description]

Path 673:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
190. try {
```

[Table of Contents] [Description]

Path 674:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
266. static synchronized boolean isExcluded(InetAddress addr) //discoveredipmgr.java
      ...
275. try {
```

[Table of Contents] [Description]

Path 675:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
164. private boolean addToSpecificsFromURL(List specifics, String url, long timeout,
int retries) //discovery.java
      ...
```

```
167. try {
```

[Table of Contents] [Description]

Path 676:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
164. private boolean addToSpecificsFromURL(List specifics, String url, long timeout,
int retries) //discovery.java
...
197. try {
```

[Table of Contents] [Description]

Path 677:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
...
237. try {
```

[Table of Contents] [Description]

Path 678:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
...
265. try {
```

[Table of Contents] [Description]

Path 679:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
...
305. try {
```

[Table of Contents] [Description]

Path 680:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
```

```
...
328. try {
```

[Table of Contents] [Description]

Path 681:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
101. static InetAddress make(long addr) //ipaddressrange.java
...
103. try {
```

[Table of Contents] [Description]

Path 682:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
197. private void sync() //ipgenerator.java
198. try {
```

[Table of Contents] [Description]

Path 683:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
351. public void run() //pingmanager.java
...
358. try {
```

[Table of Contents] [Description]

Path 684:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
351. public void run() //pingmanager.java
...
393. try {
```

[Table of Contents] [Description]

Path 685:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
```

```
...
211. try {
```

[Table of Contents] [Description]

Path 686:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
...
236. try {
```

[Table of Contents] [Description]

Path 687:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
...
256. try {
```

[Table of Contents] [Description]

Path 688:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
150. public static void loadConfiguration(Reader rdr) throws IOException,
MarshalException, ValidationException //eventconfigurationmanager.java
...
187. try {
```

[Table of Contents] [Description]

Path 689:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
232. public static void mergeConfiguration(Reader rdr) throws MarshalException,
ValidationException //eventconfigurationmanager.java
...
245. try {
```

[Table of Contents] [Description]

Path 690:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
150. public Eventd() //eventd.java
     |151. try {
```

[Table of Contents] [Description]

Path 691:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void init() //eventd.java
     |...
     |211. try {
```

[Table of Contents] [Description]

Path 692:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void init() //eventd.java
     |...
     |231. try {
```

[Table of Contents] [Description]

Path 693:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void init() //eventd.java
     |...
     |262. try {
```

[Table of Contents] [Description]

Path 694:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void init() //eventd.java
     |...
     |272. try {
```

[Table of Contents] [Description]

Path 695:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void init() //eventd.java
```

```
...
292. try {
```

[Table of Contents] [Description]

Path 696:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
89. public void run() //eventhandler.java
...
109. try {
```

[Table of Contents] [Description]

Path 697:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
89. public void run() //eventhandler.java
...
146. try {
```

[Table of Contents] [Description]

Path 698:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
...
151. try {
```

[Table of Contents] [Description]

Path 699:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
...
163. try {
```

[Table of Contents] [Description]

Path 700:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
200. public EventIpcManagerDefaultImpl() //eventipcmanagerdefaultimpl.java
```

```
...
209. try {
```

[Table of Contents] [Description]

Path 701:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
258. public synchronized void sendNow(Log
eventLog) //eventipcmanagerdefaultimpl.java
...
262. try {
```

[Table of Contents] [Description]

Path 702:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
...
297. try {
```

[Table of Contents] [Description]

Path 703:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
...
322. try {
```

[Table of Contents] [Description]

Path 704:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
...
441. try {
```

[Table of Contents] [Description]

Path 705:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
      ...
453. try {
```

[Table of Contents] [Description]

Path 706:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
      ...
473. try {
```

[Table of Contents] [Description]

Path 707:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
      ...
483. try {
```

[Table of Contents] [Description]

Path 708:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
      ...
571. try {
```

[Table of Contents] [Description]

Path 709:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
      ...
585. try {
```

[Table of Contents] [Description]

Path 710:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     |...
     696. try {
```

[Table of Contents] [Description]

Path 711:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
834. private static String getNodeLabel(long nodeId) throws
SQLException //eventutil.java
     |...
     863. try {
```

[Table of Contents] [Description]

Path 712:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
888. private static String getIfAlias(long nodeId, String ipaddr) throws
SQLException //eventutil.java
     |...
     918. try {
```

[Table of Contents] [Description]

Path 713:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
943. private static String getAssetInfo(long nodeId, String assetField) throws
SQLException //eventutil.java
     |...
     972. try {
```

[Table of Contents] [Description]

Path 714:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
     |...
     402. try {
```

[Table of Contents] [Description]

Path 715:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      ...
      419. try {
```

[Table of Contents] [Description]

Path 716:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      ...
      445. try {
```

[Table of Contents] [Description]

Path 717:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
607. public EventWriter(String getNextEventIdStr) throws
SQLException //eventwriter.java
      ...
      611. try {
```

[Table of Contents] [Description]

Path 718:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
607. public EventWriter(String getNextEventIdStr) throws
SQLException //eventwriter.java
      ...
      637. try {
```

[Table of Contents] [Description]

Path 719:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
647. public void close() //eventwriter.java
      648. try {
```

[Table of Contents] [Description]

Path 720:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
647. public void close() //eventwriter.java
       ...
656. try {
```

[Table of Contents] [Description]

Path 721:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
672. public void persistEvent(Header eventHeader, Event event) throws
SQLException //eventwriter.java
       ...
689. try {
```

[Table of Contents] [Description]

Path 722:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
66. private void findServer() throws
InstanceNotFoundException //eventhandlermbeanproxy.java
       ...
70. try {
```

[Table of Contents] [Description]

Path 723:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
99. public boolean processEvent(Event event) //eventhandlermbeanproxy.java
       ...
101. try {
```

[Table of Contents] [Description]

Path 724:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public void receiptSent(EventReceipt receipt) //eventhandlermbeanproxy.java
       112. try {
```

[Table of Contents] [Description]

Path 725:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. public synchronized void start() //tcpeventreceiver.java
       ...
162. try {
```

[Table of Contents] [Description]

Path 726:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. public synchronized void start() //tcpeventreceiver.java
       ...
175. try {
```

[Table of Contents] [Description]

Path 727:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. public synchronized void stop() //tcpeventreceiver.java
       ...
205. try {
```

[Table of Contents] [Description]

Path 728:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
170. try {
```

[Table of Contents] [Description]

Path 729:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
199. private void forward(char ch) throws IOException //tcprecordhandler.java
       200. try {
```

[Table of Contents] [Description]

Path 730:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
375. try {
```

[Table of Contents] [Description]

Path 731:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
401. try {
```

[Table of Contents] [Description]

Path 732:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
423. try {
```

[Table of Contents] [Description]

Path 733:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
429. try {
```

[Table of Contents] [Description]

Path 734:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
445. try {
```

[Table of Contents] [Description]

Path 735:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
286. try {
```

[Table of Contents] [Description]

Path 736:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
308. try {
```

[Table of Contents] [Description]

Path 737:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
334. try {
```

[Table of Contents] [Description]

Path 738:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
374. try {
```

[Table of Contents] [Description]

Path 739:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
192. try {
```

[Table of Contents] [Description]

Path 740:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
225. try {
```

[Table of Contents] [Description]

Path 741:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
239. try {
```

[Table of Contents] [Description]

Path 742:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
259. try {
```

[Table of Contents] [Description]

Path 743:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
287. try {
```

[Table of Contents] [Description]

Path 744:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
307. try {
```

[Table of Contents] [Description]

Path 745:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
     ...
343. try {
```

[Table of Contents] [Description]

Path 746:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
     ...
349. try {
```

[Table of Contents] [Description]

Path 747:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
     ...
394. try {
```

[Table of Contents] [Description]

Path 748:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
     ...
432. try {
```

[Table of Contents] [Description]

Path 749:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
     ...
446. try {
```

[Table of Contents] [Description]

Path 750:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
484. try {
```

[Table of Contents] [Description]

Path 751:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
498. try {
```

[Table of Contents] [Description]

Path 752:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. public synchronized void start() //udpeventreceiver.java
       ...
162. try {
```

[Table of Contents] [Description]

Path 753:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. public synchronized void start() //udpeventreceiver.java
       ...
181. try {
```

[Table of Contents] [Description]

Path 754:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
197. public synchronized void stop() //udpeventreceiver.java
       ...
207. try {
```

[Table of Contents] [Description]

Path 755:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      ...
163. try {
```

[Table of Contents] [Description]

Path 756:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      ...
186. try {
```

[Table of Contents] [Description]

Path 757:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      ...
201. if (isTracing)
```

[Table of Contents] [Description]

Path 758:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
147. try {
```

[Table of Contents] [Description]

Path 759:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
159. try {
```

[Table of Contents] [Description]

Path 760:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      |...
182. try {
```

[Table of Contents] [Description]

Path 761:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      |...
196. try {
```

[Table of Contents] [Description]

Path 762:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
170. try {
```

[Table of Contents] [Description]

Path 763:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
223. try {
```

[Table of Contents] [Description]

Path 764:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
232. try {
```

[Table of Contents] [Description]

Path 765:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
244. try {
```

[Table of Contents] [Description]

Path 766:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public static String format(org.opennms.netmgt.xml.event.Correlation ec, int
sz) //correlation.java
      ...
72. try {
```

[Table of Contents] [Description]

Path 767:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. public void parseRule(String rule) throws FilterParseException //filter.java
      ...
95. try {
```

[Table of Contents] [Description]

Path 768:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
      ...
175. try {
```

[Table of Contents] [Description]

Path 769:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
      ...
202. try {
```

[Table of Contents] [Description]

Path 770:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
207. try {
```

[Table of Contents] [Description]

Path 771:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
233. try {
```

[Table of Contents] [Description]

Path 772:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
266. try {
```

[Table of Contents] [Description]

Path 773:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
290. try {
```

[Table of Contents] [Description]

Path 774:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
295. try {
```

[Table of Contents] [Description]

Path 775:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
     ...
317. try {
```

[Table of Contents] [Description]

Path 776:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
90. public class SQLTranslation extends DepthFirstAdapter //sqltranslation.java
    ...
143. try {
```

[Table of Contents] [Description]

Path 777:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
307. public void checkIPNum(String octet) //sqltranslation.java
     308. try {
```

[Table of Contents] [Description]

Path 778:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public synchronized Collection waitForAnticipated(long
millis) //eventanticipator.java
     ...
113. try {
```

[Table of Contents] [Description]

Path 779:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
423. public Timestamp convertEventTimeToTimeStamp(String time) //mockdatabase.java
     424. try {
```

[Table of Contents] [Description]

Path 780:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. public void run() //mockeventipcmanager.java
      ...
159. try { Thread.sleep(2000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 781:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
189. public synchronized void finishProcessingEvents() //mockeventipcmanager.java
      ...
193. try { wait(); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 782:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. public void visitService(MockService service) //mocknetworktest.java
      157. try {
```

[Table of Contents] [Description]

Path 783:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
534. public void testScheduledOutages() //mocknetworktest.java
      ...
541. try { Thread.sleep(1000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 784:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
624. public void run() //mocknetworktest.java
      625. try {
```

[Table of Contents] [Description]

Path 785:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
624. public void run() //mocknetworktest.java
      ...
626. try {
```

[Table of Contents] [Description]

Path 786:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
624. public void run() //mocknetworktest.java
      ...
632. try {
```

[Table of Contents] [Description]

Path 787:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
711. public void run() //mocknetworktest.java
      712. try {
```

[Table of Contents] [Description]

Path 788:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
711. public void run() //mocknetworktest.java
      ...
715. try {
```

[Table of Contents] [Description]

Path 789:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
180. public static Timestamp convertEventTimeIntoTimestamp(String
eventTime) //mockutil.java
      ...
182. try {
```

[Table of Contents] [Description]

Path 790:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
105. private void waitFor(long millis) //pollanticipator.java
     |106. try {
```

[Table of Contents] [Description]

Path 791:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. public final class BroadcastEventProcessor implements
EventListener //broadcasteventprocessor.java
     |...
     |112. try {
```

[Table of Contents] [Description]

Path 792:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
185. private void automaticAcknowledge(Event event) //broadcasteventprocessor.java
     |...
     |194. try {
```

[Table of Contents] [Description]

Path 793:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
     |...
     |451. try {
```

[Table of Contents] [Description]

Path 794:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
554. private void sendNotifEvent(String uei, String logMessage, String
description) //broadcasteventprocessor.java
     |555. try {
```

[Table of Contents] [Description]

Path 795:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
728. private NotificationTask makeUserTask(long sendTime, Map parameters, int
noticeId, String targetName, String[] commandList, List siblings, String autoNotify)
throws IOException, MarshalException,
ValidationException //broadcasteventprocessor.java
      ...
731. try {
```

[Table of Contents] [Description]

Path 796:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
762. private NotificationTask makeEmailTask(long sendTime, Map parameters, int
noticeId, String address, String[] commandList, List siblings, String autoNotify)
throws IOException, MarshalException,
ValidationException //broadcasteventprocessor.java
      ...
765. try {
```

[Table of Contents] [Description]

Path 797:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
118. public void run() //defaultqueuehandler.java
      ...
134. try {
```

[Table of Contents] [Description]

Path 798:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
118. public void run() //defaultqueuehandler.java
      ...
152. try {
```

[Table of Contents] [Description]

Path 799:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
39. public int send(List arguments) //javamailnotificationstrategy.java
      ...
46. try {
```

[Table of Contents] [Description]

Path 800:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
126. public synchronized void init() //notifd.java
       ...
132. try {
```

[Table of Contents] [Description]

Path 801:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
810. private void sleep(long millis) //notifdtest.java
       811. try {
```

[Table of Contents] [Description]

Path 802:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
200. public void run() //notificationtask.java
       ...
212. try {
```

[Table of Contents] [Description]

Path 803:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
200. public void run() //notificationtask.java
       ...
225. try {
```

[Table of Contents] [Description]

Path 804:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
177. public boolean sendMessage(String xmppTo, String
xmppMessage) //xmppnotificationmanager.java
       ...
179. try {
```

[Table of Contents] [Description]

Path 805:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
200. public boolean sendGroupChat(String xmppChatRoom, String
xmppMessage) //xmppnotificationmanager.java
      ...
216. try {
```

[Table of Contents] [Description]

Path 806:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
200. public boolean sendGroupChat(String xmppChatRoom, String
xmppMessage) //xmppnotificationmanager.java
      ...
224. try {
```

[Table of Contents] [Description]

Path 807:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
49. public void init() //notifd.java
      ...
52. try {
```

[Table of Contents] [Description]

Path 808:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
49. public void init() //notifd.java
      ...
58. try {
```

[Table of Contents] [Description]

Path 809:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
140. public synchronized Collection waitForAnticipated(long
millis) //notificationanticipator.java
      ...
147. try {
```

[Table of Contents] [Description]

Path 810:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. public void verifyAnticipated(long lastNotifyTime, long
waitTime, //notificationanticipator.java
     ...
167. try {
```

[Table of Contents] [Description]

Path 811:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public static Timestamp convertEventTimeIntoTimestamp(String
eventTime) //basicnetwork.java
     ...
122. try {
```

[Table of Contents] [Description]

Path 812:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. public static void close(Connection dbConn) //dbutil.java
     ...
76. try {
```

[Table of Contents] [Description]

Path 813:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
84. public static void rollback(Connection dbConn, String msg, SQLException
se) //dbutil.java
     ...
88. try {
```

[Table of Contents] [Description]

Path 814:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
95. public static void commit(Connection dbConn, String successMsg, String
failMsg) //dbutil.java
```

```
...
97. try {
```

[Table of Contents] [Description]

Path 815:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
...
107. try {
```

[Table of Contents] [Description]

Path 816:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
...
152. try {
```

[Table of Contents] [Description]

Path 817:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
...
193. try {
```

[Table of Contents] [Description]

Path 818:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
...
233. try {
```

[Table of Contents] [Description]

Path 819:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
211. public void init() //outagemanager.java
      |
      ...
      234. try {
```

[Table of Contents] [Description]

Path 820:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
298. public void start() //outagemanager.java
      |
      ...
      317. try {
```

[Table of Contents] [Description]

Path 821:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
      |
      ...
      152. try {
```

[Table of Contents] [Description]

Path 822:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
325. private void sleep(long millis) //outagetest.java
      |
      326. try { Thread.sleep(millis); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 823:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
      |
      ...
      173. try {
```

[Table of Contents] [Description]

Path 824:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
197. private void handleNodeLostService(long eventID, String eventTime, BasicService
svc) //outagewriter.java
      |...
      |208. try {
```

[Table of Contents] [Description]

Path 825:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
230. private void handleInterfaceDown(long eventID, String eventTime, BasicInterface
iface) //outagewriter.java
      |...
      |237. try {
```

[Table of Contents] [Description]

Path 826:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
257. private void handleNodeDown(long eventID, String eventTime, BasicNode
node) //outagewriter.java
      |...
      |264. try {
```

[Table of Contents] [Description]

Path 827:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
283. private void handleNodeUp(long eventID, String eventTime, BasicNode
node) //outagewriter.java
      |...
      |292. try {
```

[Table of Contents] [Description]

Path 828:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
319. private void handleInterfaceUp(long eventID, String eventTime, BasicInterface
iface) //outagewriter.java
      |...
      |328. try {
```

[Table of Contents] [Description]

Path 829:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
352. private void handleNodeRegainedService(long eventID, String eventTime,
BasicService svc) //outagewriter.java
      |
      ...
      361. try {
```

[Table of Contents] [Description]

Path 830:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |
      ...
      423. try {
```

[Table of Contents] [Description]

Path 831:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |
      ...
      434. try {
```

[Table of Contents] [Description]

Path 832:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |
      ...
      450. try {
```

[Table of Contents] [Description]

Path 833:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
      |
      ...
      575. try {
```

[Table of Contents] [Description]

Path 834:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
643. public void run() //outagewriter.java
     644. try {
```

[Table of Contents] [Description]

Path 835:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
51. public void init() //outaged.java
     ...
     60. try {
```

[Table of Contents] [Description]

Path 836:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. protected void process(DatagramPacket pkt) throws InterruptedException,
FifoQueueException //replyreceiver.java
     ...
     165. try {
```

[Table of Contents] [Description]

Path 837:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public boolean activeServiceExists(String whichEvent, int nodeId, String ipAddr,
String serviceName) //defaultquerymanager.java
     ...
     102. try {
```

[Table of Contents] [Description]

Path 838:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
     ...
```

```
153. try {
```

[Table of Contents] [Description]

Path 839:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
...
254. try {
```

[Table of Contents] [Description]

Path 840:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
...
302. try {
```

[Table of Contents] [Description]

Path 841:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
379. public Date getServiceLostDate(int nodeId, String ipAddr, String svcName, int
serviceId) //defaultquerymanager.java
...
396. try {
```

[Table of Contents] [Description]

Path 842:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
379. public Date getServiceLostDate(int nodeId, String ipAddr, String svcName, int
serviceId) //defaultquerymanager.java
...
430. try {
```

[Table of Contents] [Description]

Path 843:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
463. public Timestamp convertEventTimeToTimeStamp(String
time) //defaultquerymanager.java
     |464. try {
```

[Table of Contents] [Description]

Path 844:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public synchronized void init() //poller.java
     |...
     |152. try {
```

[Table of Contents] [Description]

Path 845:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. private void createScheduler() //poller.java
     |...
     |210. try {
```

[Table of Contents] [Description]

Path 846:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
228. public synchronized void start() //poller.java
     |...
     |236. try {
```

[Table of Contents] [Description]

Path 847:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
416. private void scheduleService(int nodeId, String ipAddr, String serviceName,
Number svcLostEventId, Date date, String svcLostUei) //poller.java
     |...
     |432. try {
```

[Table of Contents] [Description]

Path 848:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
        ...
611. try {
```

[Table of Contents] [Description]

Path 849:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
        ...
625. try {
```

[Table of Contents] [Description]

Path 850:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. private void interfaceReparentedHandler(Event
event) //pollereventprocessor.java
        ...
264. try {
```

[Table of Contents] [Description]

Path 851:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
299. private void nodeRemovePollableServiceHandler(Event
event) //pollereventprocessor.java
        ...
307. try {
```

[Table of Contents] [Description]

Path 852:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
330. private void nodeDeletedHandler(Event event) //pollereventprocessor.java
        ...
359. try {
```

[Table of Contents] [Description]

Path 853:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
330. private void nodeDeletedHandler(Event event) //pollereventprocessor.java
      |...
      |370. try {
```

[Table of Contents] [Description]

Path 854:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      |...
      |424. try {
```

[Table of Contents] [Description]

Path 855:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      |...
      |435. try {
```

[Table of Contents] [Description]

Path 856:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      |...
      |444. try {
```

[Table of Contents] [Description]

Path 857:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
472. private void serviceDeletedHandler(Event event) //pollereventprocessor.java
      |...
      |480. try {
```

[Table of Contents] [Description]

Path 858:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
472. private void serviceDeletedHandler(Event event) //pollereventprocessor.java
       ...
488. try {
```

[Table of Contents] [Description]

Path 859:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
748. private void sleep(long millis) //pollertest.java
       749. try {
```

[Table of Contents] [Description]

Path 860:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
51. public void init() //pollerd.java
       ...
57. try {
```

[Table of Contents] [Description]

Path 861:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
       ...
157. try {
```

[Table of Contents] [Description]

Path 862:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
       ...
184. try {
```

[Table of Contents] [Description]

Path 863:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
     ...
216. try {
```

[Table of Contents] [Description]

Path 864:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dhcpmonitor.java
     ...
127. try {
```

[Table of Contents] [Description]

Path 865:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dhcpmonitor.java
     ...
146. try {
```

[Table of Contents] [Description]

Path 866:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
145. try {
```

[Table of Contents] [Description]

Path 867:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
163. try {
```

[Table of Contents] [Description]

Path 868:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     |
     ...
     169. try {
```

[Table of Contents] [Description]

Path 869:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     |
     ...
     216. try {
```

[Table of Contents] [Description]

Path 870:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
     |
     ...
     196. try {
```

[Table of Contents] [Description]

Path 871:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
     |
     ...
     226. try {
```

[Table of Contents] [Description]

Path 872:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
77. final public class FtpMonitor extends IPv4LatencyMonitor //ftpmonitor.java
     |
```

```
...
122. try {
```

[Table of Contents] [Description]

Path 873:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
187. try {
```

[Table of Contents] [Description]

Path 874:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
220. try {
```

[Table of Contents] [Description]

Path 875:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
238. try {
```

[Table of Contents] [Description]

Path 876:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
297. try {
```

[Table of Contents] [Description]

Path 877:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
352. try {
```

[Table of Contents] [Description]

Path 878:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
378. try {
```

[Table of Contents] [Description]

Path 879:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
399. try {
```

[Table of Contents] [Description]

Path 880:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
418. try {
```

[Table of Contents] [Description]

Path 881:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
454. try {
```

[Table of Contents] [Description]

Path 882:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     ...
180. try {
```

[Table of Contents] [Description]

Path 883:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     ...
220. try {
```

[Table of Contents] [Description]

Path 884:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     ...
290. try {
```

[Table of Contents] [Description]

Path 885:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     ...
327. try {
```

[Table of Contents] [Description]

Path 886:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
```

```
...
192. try {
```

[Table of Contents] [Description]

Path 887:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
231. try {
```

[Table of Contents] [Description]

Path 888:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
241. try {
```

[Table of Contents] [Description]

Path 889:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
251. try {
```

[Table of Contents] [Description]

Path 890:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
261. try {
```

[Table of Contents] [Description]

Path 891:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     ...
327. try {
```

[Table of Contents] [Description]

Path 892:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
     ...
310. try {
```

[Table of Contents] [Description]

Path 893:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
     ...
320. try {
```

[Table of Contents] [Description]

Path 894:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
     ...
348. try {
```

[Table of Contents] [Description]

Path 895:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     ...
180. try {
```

[Table of Contents] [Description]

Path 896:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     ...
222. try {
```

[Table of Contents] [Description]

Path 897:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     ...
255. try {
```

[Table of Contents] [Description]

Path 898:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. public void initialize(PollerConfig pollerConfig, Map
parameters) //ipv4latencymonitor.java
     ...
105. try {
```

[Table of Contents] [Description]

Path 899:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
246. public void updateRRD(String repository, InetAddress addr, String dsName, long
value, org.opennms.netmgt.config.poller.Package pkg) //ipv4latencymonitor.java
     ...
249. try {
```

[Table of Contents] [Description]

Path 900:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
     |
```

```
...
256. try
```

[Table of Contents] [Description]

Path 901:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
...
279. try
```

[Table of Contents] [Description]

Path 902:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
...
308. try
```

[Table of Contents] [Description]

Path 903:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
...
326. try
```

[Table of Contents] [Description]

Path 904:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
...
356. try
```

[Table of Contents] [Description]

Path 905:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
      ...
      72. try
```

[Table of Contents] [Description]

Path 906:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
      ...
      107. try
```

[Table of Contents] [Description]

Path 907:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
      ...
      168. try {
```

[Table of Contents] [Description]

Path 908:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
      ...
      188. try {
```

[Table of Contents] [Description]

Path 909:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
      ...
      198. try {
```

[Table of Contents] [Description]

Path 910:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
202. try {
```

[Table of Contents] [Description]

Path 911:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
221. try {
```

[Table of Contents] [Description]

Path 912:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
232. try {
```

[Table of Contents] [Description]

Path 913:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
242. try {
```

[Table of Contents] [Description]

Path 914:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
```

```
org.opennms.netmgt.config.poller.Package pkg) //nrpemonitor.java
    ...
174. try {
```

[Table of Contents] [Description]

Path 915:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //nrpemonitor.java
    ...
199. try {
```

[Table of Contents] [Description]

Path 916:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //nrpemonitor.java
    ...
242. try {
```

[Table of Contents] [Description]

Path 917:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
116. public int poll(NetworkInterface iface, Map parameters, //nsclientmonitor.java
    ...
180. try {
```

[Table of Contents] [Description]

Path 918:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
116. public int poll(NetworkInterface iface, Map parameters, //nsclientmonitor.java
    ...
230. try {
```

[Table of Contents] [Description]

Path 919:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
     ...
137. try {
```

[Table of Contents] [Description]

Path 920:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
     ...
143. try {
```

[Table of Contents] [Description]

Path 921:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
     ...
193. try {
```

[Table of Contents] [Description]

Path 922:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     ...
154. try {
```

[Table of Contents] [Description]

Path 923:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     ...
199. try {
```

[Table of Contents] [Description]

Path 924:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     ...
231. try {
```

[Table of Contents] [Description]

Path 925:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
146. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //radiusauthmonitor.java
     ...
182. try {
```

[Table of Contents] [Description]

Path 926:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
146. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //radiusauthmonitor.java
     ...
194. try {
```

[Table of Contents] [Description]

Path 927:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
146. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //radiusauthmonitor.java
     ...
220. try {
```

[Table of Contents] [Description]

Path 928:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. final public class SmtpMonitor extends IPv4LatencyMonitor //smtpmonitor.java
     ...
```

```
119. try {
```

[Table of Contents] [Description]

Path 929:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. final public class SmtpMonitor extends IPv4LatencyMonitor //smtpmonitor.java
    ...
126. try {
```

[Table of Contents] [Description]

Path 930:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
    ...
194. try {
```

[Table of Contents] [Description]

Path 931:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
    ...
227. try {
```

[Table of Contents] [Description]

Path 932:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
    ...
272. try {
```

[Table of Contents] [Description]

Path 933:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
309. try {
```

[Table of Contents] [Description]

Path 934:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
331. try {
```

[Table of Contents] [Description]

Path 935:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
371. try {
```

[Table of Contents] [Description]

Path 936:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
     ...
160. try {
```

[Table of Contents] [Description]

Path 937:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
     ...
279. try {
```

[Table of Contents] [Description]

Path 938:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
     ...
304. try {
```

[Table of Contents] [Description]

Path 939:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
     ...
313. try {
```

[Table of Contents] [Description]

Path 940:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
     ...
324. try {
```

[Table of Contents] [Description]

Path 941:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public void testErrorConditions() //snmpmonitortest.java
     ...
108. try {
```

[Table of Contents] [Description]

Path 942:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public void testErrorConditions() //snmpmonitortest.java
     ...
114. try {
```

[Table of Contents] [Description]

Path 943:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
160. try {
```

[Table of Contents] [Description]

Path 944:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
204. try {
```

[Table of Contents] [Description]

Path 945:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
211. try {
```

[Table of Contents] [Description]

Path 946:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
238. try {
```

[Table of Contents] [Description]

Path 947:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
```

```
...
166. try {
```

[Table of Contents] [Description]

Path 948:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
...
185. try {
```

[Table of Contents] [Description]

Path 949:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
...
214. try {
```

[Table of Contents] [Description]

Path 950:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
...
241. try {
```

[Table of Contents] [Description]

Path 951:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
139. public byte[] buildPacket(int padding) //nrpepacket.java
...
142. try {
```

[Table of Contents] [Description]

Path 952:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
379. public void init() throws NsclientException //nsclientmanager.java
     |380. try {
```

[Table of Contents] [Description]

Path 953:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
470. public NsclientPacket processCheckCommand(short type, //nsclientmanager.java
     |...
     |472. try {
```

[Table of Contents] [Description]

Path 954:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
516. private NsclientPacket checkClientVersion(NsclientCheckParams
param) //nsclientmanager.java
     |...
     |521. try {
```

[Table of Contents] [Description]

Path 955:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
595. private NsclientPacket checkCpuLoad(NsclientCheckParams
param) //nsclientmanager.java
     |...
     |598. try {
```

[Table of Contents] [Description]

Path 956:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
642. private NsclientPacket checkUptime(NsclientCheckParams
param) //nsclientmanager.java
     |...
     |645. try {
```

[Table of Contents] [Description]

Path 957:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
678. private NsclientPacket checkServiceState(NsclientCheckParams
param) //nsclientmanager.java
      ...
681. try {
```

[Table of Contents] [Description]

Path 958:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
720. private NsclientPacket checkProcState(NsclientCheckParams
param) //nsclientmanager.java
      ...
723. try {
```

[Table of Contents] [Description]

Path 959:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
763. private NsclientPacket checkUsedDiskSpace(NsclientCheckParams
param) //nsclientmanager.java
      ...
767. try {
```

[Table of Contents] [Description]

Path 960:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
822. private NsclientPacket checkMemoryUsage(NsclientCheckParams
param) //nsclientmanager.java
      ...
825. try {
```

[Table of Contents] [Description]

Path 961:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
876. private NsclientPacket checkPerfCounter(NsclientCheckParams
param) //nsclientmanager.java
```

```
...
880. try {
```

[Table of Contents] [Description]

Path 962:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
929. private NsclientPacket checkFileAge(NsclientCheckParams
param) //nsclientmanager.java
...
932. try {
```

[Table of Contents] [Description]

Path 963:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. public Date getDate() //pendingpollevent.java
60. try {
```

[Table of Contents] [Description]

Path 964:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
58. public synchronized void obtain() //pollablenode.java
...
64. try { wait();} catch (InterruptedException e) { throw new
ThreadInterrupted("Lock for "+PollableNode.this+" is unavailable", e);}
```

[Table of Contents] [Description]

Path 965:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
72. public synchronized void obtain(long timeout) //pollablenode.java
...
80. try { wait(endTime-now);} catch (InterruptedException e) { throw new
ThreadInterrupted("Lock for "+PollableNode.this+" is unavailable", e);}
```

[Table of Contents] [Description]

Path 966:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
273. public void run() //pollableservice.java
       ...
       284. try {
```

[Table of Contents] [Description]

Path 967:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
       ...
       297. try {
```

[Table of Contents] [Description]

Path 968:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1998. public void run() //pollablestest.java
       ...
       2001. try { Thread.sleep(3000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 969:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2027. public void run() //pollablestest.java
       ...
       2030. try { Thread.sleep(5000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 970:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
2043. public void run() //pollablestest.java
       2044. try {
```

[Table of Contents] [Description]

Path 971:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. public static void main(String[] argv) //interface.java
```

```
174. try {
```

[Table of Contents] [Description]

Path 972:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
...
258. try {
```

[Table of Contents] [Description]

Path 973:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
...
358. try {
```

[Table of Contents] [Description]

Path 974:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
156. public Double fetchLastValue(String fileName, int interval) throws
NumberFormatException, org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
...
158. try {
```

[Table of Contents] [Description]

Path 975:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
193. public Double fetchLastValueInRange(String fileName, int interval, int range)
throws NumberFormatException,
org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
...
195. try {
```

[Table of Contents] [Description]

Path 976:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. void addToPendingList(LinkedList pendingOperations) //queuingrrdstrategy.java
      ...
406. try {
```

[Table of Contents] [Description]

Path 977:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
422. public Operation makeUpdateOperation(String fileName, String
update) //queuingrrdstrategy.java
      423. try {
```

[Table of Contents] [Description]

Path 978:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
518. public LinkedList getNext() //queuingrrdstrategy.java
      468. if (op.isSignificant())
```

[Table of Contents] [Description]

Path 979:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
794. public void run() //queuingrrdstrategy.java
      ...
808. try {
```

[Table of Contents] [Description]

Path 980:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
85. public static String getProperty(String name, String
defaultVal) //rrdconfig.java
      ...
88. try {
```

[Table of Contents] [Description]

Path 981:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
121. public static int getProperty(String name, int defaultVal) //rrdconfig.java
       ...
124. try {
```

[Table of Contents] [Description]

Path 982:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
142. public static long getProperty(String name, long defaultVal) //rrdconfig.java
       ...
145. try {
```

[Table of Contents] [Description]

Path 983:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
333. public boolean performUpdate( //rrddatasource.java
       ...
343. try {
```

[Table of Contents] [Description]

Path 984:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
354. public String getStorableValue(SnmpSyntax snmpVar) //rrddatasource.java
       ...
380. try {
```

[Table of Contents] [Description]

Path 985:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
157. private static String makeZeroUpdate(int fileNum, String
update) //rrdstresser.java
       158. try {
```

[Table of Contents] [Description]

Path 986:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
314. public void execute(String[] args) throws Exception //rrdstresser.java
     |...
     327. try {
```

[Table of Contents] [Description]

Path 987:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
     |...
     150. try {
```

[Table of Contents] [Description]

Path 988:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
     |...
     135. try {
```

[Table of Contents] [Description]

Path 989:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
     |...
     168. try {
```

[Table of Contents] [Description]

Path 990:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
406. private void createCategoriesMap() //datamanager.java
     |...
     410. try {
```

[Table of Contents] [Description]

Path 991:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |...
      |602. try {
```

[Table of Contents] [Description]

Path 992:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
      |...
      |637. try {
```

[Table of Contents] [Description]

Path 993:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
      |...
      |689. try {
```

[Table of Contents] [Description]

Path 994:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
710. public synchronized void nodeGainedService(long nodeid, String ip, String
svcName) //datamanager.java
      |...
      |747. try {
```

[Table of Contents] [Description]

Path 995:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |...
      |1017. try {
```

[Table of Contents] [Description]

Path 996:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1075. try {
```

[Table of Contents] [Description]

Path 997:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1247. try {
```

[Table of Contents] [Description]

Path 998:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      ...
250. try {
```

[Table of Contents] [Description]

Path 999:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      ...
282. try {
```

[Table of Contents] [Description]

Path 1000:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. public synchronized void unsubscribe(String urlStr) //datasender.java
      |
```

```
...
336. try {
```

[Table of Contents] [Description]

Path 1001:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
...
407. try {
```

[Table of Contents] [Description]

Path 1002:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
...
424. try {
```

[Table of Contents] [Description]

Path 1003:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
483. public synchronized void notifyToSend() //datasender.java
484. try {
```

[Table of Contents] [Description]

Path 1004:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
218. private void handleInterfaceReparented(String ip, Parms
eventParms) //dataupdater.java
...
249. try {
```

[Table of Contents] [Description]

Path 1005:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
218. private void handleInterfaceReparented(String ip, Parms
```

```
eventParms) //dataupdater.java
    ...
260. try {
```

[Table of Contents] [Description]

Path 1006:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
    ...
436. try {
```

[Table of Contents] [Description]

Path 1007:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
509. public void run() //dataupdater.java
    ...
512. try {
```

[Table of Contents] [Description]

Path 1008:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
    ...
356. try {
```

[Table of Contents] [Description]

Path 1009:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
    ...
368. try {
```

[Table of Contents] [Description]

Path 1010:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
418. try {
```

[Table of Contents] [Description]

Path 1011:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
432. public synchronized void resetUserTimer() //rtcmanager.java
       ...
439. try {
```

[Table of Contents] [Description]

Path 1012:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
463. public void init() //rtcmanager.java
       ...
470. try {
```

[Table of Contents] [Description]

Path 1013:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
607. try {
```

[Table of Contents] [Description]

Path 1014:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
71. public void run() //pipedmarshaller.java
      72. try {
```

[Table of Contents] [Description]

Path 1015:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. public Reader getReader() throws IOException //pipedmarshaller.java
```

```
...
99. try {
```

[Table of Contents] [Description]

Path 1016:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
74. public void run() //schedule.java
...
81. try {
```

[Table of Contents] [Description]

Path 1017:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
...
268. try {
```

[Table of Contents] [Description]

Path 1018:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
433. public void run() //scheduler.java
...
459. try {
```

[Table of Contents] [Description]

Path 1019:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
433. public void run() //scheduler.java
...
493. try {
```

[Table of Contents] [Description]

Path 1020:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
526. try {
```

[Table of Contents] [Description]

Path 1021:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
99. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
106. try {
```

[Table of Contents] [Description]

Path 1022:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
       ...
193. try {
```

[Table of Contents] [Description]

Path 1023:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
       ...
211. try {
```

[Table of Contents] [Description]

Path 1024:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
       ...
234. try {
```

[Table of Contents] [Description]

Path 1025:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
      ...
271. try {
```

[Table of Contents] [Description]

Path 1026:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
      ...
286. try {
```

[Table of Contents] [Description]

Path 1027:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
174. public void run() //executor.java
      ...
300. try {
```

[Table of Contents] [Description]

Path 1028:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
335. public synchronized void start() //executor.java
      ...
376. try {
```

[Table of Contents] [Description]

Path 1029:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
397. public synchronized void stop() //executor.java
      ...
417. try {
```

[Table of Contents] [Description]

Path 1030:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
102. public synchronized void init() //scriptd.java
       ...
110. try {
```

[Table of Contents] [Description]

Path 1031:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
284. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
       ...
287. try {
```

[Table of Contents] [Description]

Path 1032:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
388. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
       ...
391. try {
```

[Table of Contents] [Description]

Path 1033:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
429. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
       ...
432. try {
```

[Table of Contents] [Description]

Path 1034:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
474. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
       ...
477. try {
```

[Table of Contents] [Description]

Path 1035:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
519. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
     ...
522. try {
```

[Table of Contents] [Description]

Path 1036:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
599. public SnmpVarBind getVarBind(String name, String encoding, String value)
throws SnmpTrapHelperException //snmptraphelper.java
     ...
602. try {
```

[Table of Contents] [Description]

Path 1037:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
636. public void sendTrap(String community, SnmpPduTrap trap, String destAddr, int
destPort) throws SnmpTrapHelperException //snmptraphelper.java
     ...
638. try {
```

[Table of Contents] [Description]

Path 1038:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
676. public void sendTrap(String community, SnmpPduPacket packet, String destAddr,
int destPort) throws SnmpTrapHelperException //snmptraphelper.java
     ...
678. try {
```

[Table of Contents] [Description]

Path 1039:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
880. public void forwardV1Trap(Event event, String destAddr, int destPort) throws
SnmpTrapHelperException //snmptraphelper.java
     ...
916. try {
```

[Table of Contents] [Description]

Path 1040:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
880. public void forwardV1Trap(Event event, String destAddr, int destPort) throws
SnmpTrapHelperException //snmptraphelper.java
     ...
965. try {
```

[Table of Contents] [Description]

Path 1041:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1001. public void forwardV2Trap(Event event, String destAddr, int destPort) throws
SnmpTrapHelperException //snmptraphelper.java
     ...
1045. try {
```

[Table of Contents] [Description]

Path 1042:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1001. public void forwardV2Trap(Event event, String destAddr, int destPort) throws
SnmpTrapHelperException //snmptraphelper.java
     ...
1081. try {
```

[Table of Contents] [Description]

Path 1043:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
165. try {
```

[Table of Contents] [Description]

Path 1044:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
```

```
173. try {
```

[Table of Contents] [Description]

Path 1045:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
222. try {
```

[Table of Contents] [Description]

Path 1046:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
237. try {
```

[Table of Contents] [Description]

Path 1047:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
258. try {
```

[Table of Contents] [Description]

Path 1048:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
287. try {
```

[Table of Contents] [Description]

Path 1049:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      ...
316. try {
```

[Table of Contents] [Description]

Path 1050:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      ...
344. try {
```

[Table of Contents] [Description]

Path 1051:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
      ...
455. try {
```

[Table of Contents] [Description]

Path 1052:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
      ...
470. try {
```

[Table of Contents] [Description]

Path 1053:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
      ...
545. try {
```

[Table of Contents] [Description]

Path 1054:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
609. private Event createEvent(int nodeId, InetAddress ipAddr, double dsValue,
Threshold threshold, String uei, java.util.Date date) //latencythresholder.java
     ...
631. try {
```

[Table of Contents] [Description]

Path 1055:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     ...
183. try {
```

[Table of Contents] [Description]

Path 1056:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     ...
191. try {
```

[Table of Contents] [Description]

Path 1057:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
240. try {
```

[Table of Contents] [Description]

Path 1058:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
265. try {
```

[Table of Contents] [Description]

Path 1059:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
287. try {
```

[Table of Contents] [Description]

Path 1060:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
228. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 1061:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
349. try {
```

[Table of Contents] [Description]

Path 1062:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
398. try {
```

[Table of Contents] [Description]

Path 1063:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
```

```
...
507. try {
```

[Table of Contents] [Description]

Path 1064:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
...
527. try {
```

[Table of Contents] [Description]

Path 1065:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
...
540. try {
```

[Table of Contents] [Description]

Path 1066:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
...
615. try {
```

[Table of Contents] [Description]

Path 1067:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
...
626. try {
```

[Table of Contents] [Description]

Path 1068:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
639. try {
```

[Table of Contents] [Description]

Path 1069:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     ...
793. try {
```

[Table of Contents] [Description]

Path 1070:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     ...
829. try {
```

[Table of Contents] [Description]

Path 1071:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     ...
839. try {
```

[Table of Contents] [Description]

Path 1072:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
```

```
events) throws IllegalArgumentException //snmpthresholder.java
      |...
      |857. try {
```

[Table of Contents] [Description]

Path 1073:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
922. private Event createEvent(Integer nodeId, InetAddress primary, Map ifDataMap,
double dsValue, Threshold threshold, String uei, java.util.Date date, String
label) //snmpthresholder.java
      |...
      |964. try {
```

[Table of Contents] [Description]

Path 1074:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      |...
      |172. try {
```

[Table of Contents] [Description]

Path 1075:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      |...
      |190. try {
```

[Table of Contents] [Description]

Path 1076:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      |...
      |209. try {
```

[Table of Contents] [Description]

Path 1077:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
       ...
234. try {
```

[Table of Contents] [Description]

Path 1078:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
       ...
303. try {
```

[Table of Contents] [Description]

Path 1079:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
       ...
229. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1080:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
352. public void run() //threshd.java
       ...
354. try {
```

[Table of Contents] [Description]

Path 1081:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
       ...
372. try {
```

[Table of Contents] [Description]

Path 1082:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
       |
       ...
409. try {
```

[Table of Contents] [Description]

Path 1083:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
       |
       ...
559. try {
```

[Table of Contents] [Description]

Path 1084:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
       |
       ...
594. try {
```

[Table of Contents] [Description]

Path 1085:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
       |
       ...
678. try {
```

[Table of Contents] [Description]

Path 1086:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
288. private void sendEvent(String uei) //thresholdableservice.java
       |
       ...
296. try {
```

[Table of Contents] [Description]

Path 1087:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
      |...
      |351. try {
```

[Table of Contents] [Description]

Path 1088:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
      |...
      |475. try {
```

[Table of Contents] [Description]

Path 1089:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
      |...
      |495. try {
```

[Table of Contents] [Description]

Path 1090:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
      |...
      |509. try {
```

[Table of Contents] [Description]

Path 1091:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
293. public void snmpReceivedTrap(SnmpTrapSession session, InetAddress agent, int
port, SnmpOctetString community, SnmpPduPacket pdu) //trapd.java
      |294. try {
```

[Table of Contents] [Description]

Path 1092:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
323. public void snmpReceivedTrap(SnmpTrapSession session, InetAddress agent, int
port, SnmpOctetString community, SnmpPduTrap pdu) //trapd.java
     |324. try {
```

[Table of Contents] [Description]

Path 1093:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     |...
     |352. try {
```

[Table of Contents] [Description]

Path 1094:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
     |...
     |500. try {
```

[Table of Contents] [Description]

Path 1095:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. static synchronized void dataSourceSync() throws SQLException //trapdipmgr.java
     |...
     |118. try {
```

[Table of Contents] [Description]

Path 1096:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     |...
     |412. try {
```

[Table of Contents] [Description]

Path 1097:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
760. private synchronized boolean statusOK() //trapqueueprocessor.java
       ...
789. try {
```

[Table of Contents] [Description]

Path 1098:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
807. TrapQueueProcessor(FifoQueue backlog, boolean
newSuspect) //trapqueueprocessor.java
       ...
810. try {
```

[Table of Contents] [Description]

Path 1099:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
909. public void run() //trapqueueprocessor.java
       ...
918. try {
```

[Table of Contents] [Description]

Path 1100:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
909. public void run() //trapqueueprocessor.java
       ...
938. try {
```

[Table of Contents] [Description]

Path 1101:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
909. public void run() //trapqueueprocessor.java
       ...
944. try {
```

[Table of Contents] [Description]

Path 1102:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
115. public ExecRunner(String command) throws
ExceptionInInitializerError //execrunner.java
     ...
117. try {
```

[Table of Contents] [Description]

Path 1103:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
233. public int exec(String command, PrintWriter stdoutWriter, PrintWriter
stderrWriter) throws IOException, InterruptedException //execrunner.java
     ...
292. try {
```

[Table of Contents] [Description]

Path 1104:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
87. public static java.util.List parse(String url) //iplistfromurl.java
     ...
90. try {
```

[Table of Contents] [Description]

Path 1105:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
     ...
56. try{
```

[Table of Contents] [Description]

Path 1106:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public void execute(Object values[]) //jdbctemplate.java
     81. try {
```

[Table of Contents] [Description]

Path 1107:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
277. try {
```

[Table of Contents] [Description]

Path 1108:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
    ...
370. try {
```

[Table of Contents] [Description]

Path 1109:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
440. try {
```

[Table of Contents] [Description]

Path 1110:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
498. try {
```

[Table of Contents] [Description]

Path 1111:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
```

```
532. try {
```

[Table of Contents] [Description]

Path 1112:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
580. try {
```

[Table of Contents] [Description]

Path 1113:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
61. public static int getKeyedInteger(Map map, String key, int
defValue) //parametermap.java
...
66. try {
```

[Table of Contents] [Description]

Path 1114:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
87. public final static int[] getKeyedIntegerArray(Map map, String key, int[]
defValues) //parametermap.java
...
100. try {
```

[Table of Contents] [Description]

Path 1115:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
...
73. try {
```

[Table of Contents] [Description]

Path 1116:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public final class TcpEventProxy implements EventProxy //tcpeventproxy.java
      |...
      |80. try {
```

[Table of Contents] [Description]

Path 1117:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
154. public Connection(InetAddress target, int port) throws
IOException //tcpeventproxy.java
      |...
      |157. try {
```

[Table of Contents] [Description]

Path 1118:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
169. public void run() //tcpeventproxy.java
      |...
      |172. try {
```

[Table of Contents] [Description]

Path 1119:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
191. public void close() //tcpeventproxy.java
      |...
      |193. try {
```

[Table of Contents] [Description]

Path 1120:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      |...
      |72. try {
```

[Table of Contents] [Description]

Path 1121:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      ...
      122. try {
```

[Table of Contents] [Description]

Path 1122:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      ...
      143. try {
```

[Table of Contents] [Description]

Path 1123:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
88. public void init() //vacuumd.java
      ...
      94. try {
```

[Table of Contents] [Description]

Path 1124:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
203. public void run() //vacuumd.java
      ...
      220. try {
```

[Table of Contents] [Description]

Path 1125:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
250. public void runUpdate(String sql) //vacuumd.java
      ...
      257. try {
```

[Table of Contents] [Description]

Path 1126:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
      ...
139. try {
```

[Table of Contents] [Description]

Path 1127:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
      ...
192. try {
```

[Table of Contents] [Description]

Path 1128:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
      ...
258. try {
```

[Table of Contents] [Description]

Path 1129:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
      ...
286. try {
```

[Table of Contents] [Description]

Path 1130:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
      ...
354. try {
```

[Table of Contents] [Description]

Path 1131:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     |...
     383. try {
```

[Table of Contents] [Description]

Path 1132:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
428. public void doTestLoadLibraries() //manager.java
     |...
     434. try {
```

[Table of Contents] [Description]

Path 1133:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
428. public void doTestLoadLibraries() //manager.java
     |...
     445. try {
```

[Table of Contents] [Description]

Path 1134:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
428. public void doTestLoadLibraries() //manager.java
     |...
     456. try {
```

[Table of Contents] [Description]

Path 1135:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
     |...
     480. try {
```

[Table of Contents] [Description]

Path 1136:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
      ...
493. try {
```

[Table of Contents] [Description]

Path 1137:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
44. static public Socket getConnection(InetAddress hostname, int
hostport) //nessusconnectionfactory.java
      ...
47. try {
```

[Table of Contents] [Description]

Path 1138:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. static public void releaseConnection(Socket
socket) //nessusconnectionfactory.java
      ...
62. try {
```

[Table of Contents] [Description]

Path 1139:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. private void initREs() //nessusparser.java
      ...
112. try {
```

[Table of Contents] [Description]

Path 1140:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
178. public PortValues parsePort(String portString) throws
IllegalArgumentException //nessusparser.java
      ...
185. try {
```

[Table of Contents] [Description]

Path 1141:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
202. public NessusScan(NessusScanConfiguration newConfig) throws
IllegalArgumentException //nessusscan.java
     ...
208. try {
```

[Table of Contents] [Description]

Path 1142:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
238. try {
```

[Table of Contents] [Description]

Path 1143:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
244. try {
```

[Table of Contents] [Description]

Path 1144:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
259. try {
```

[Table of Contents] [Description]

Path 1145:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
272. try {
```

[Table of Contents] [Description]

Path 1146:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
       |
       ...
452. try {
```

[Table of Contents] [Description]

Path 1147:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
       |
       ...
458. try {
```

[Table of Contents] [Description]

Path 1148:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
225. public void run() //nessusscan.java
       |
       ...
487. try {
```

[Table of Contents] [Description]

Path 1149:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       |
       ...
588. try {
```

[Table of Contents] [Description]

Path 1150:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       |
       ...
598. try {
```

[Table of Contents] [Description]

Path 1151:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
608. try {
```

[Table of Contents] [Description]

Path 1152:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
619. try {
```

[Table of Contents] [Description]

Path 1153:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
625. try {
```

[Table of Contents] [Description]

Path 1154:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
760. try {
```

[Table of Contents] [Description]

Path 1155:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
778. try {
```

[Table of Contents] [Description]

Path 1156:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
792. try {
```

[Table of Contents] [Description]

Path 1157:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
798. try {
```

[Table of Contents] [Description]

Path 1158:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
902. try {
```

[Table of Contents] [Description]

Path 1159:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
935. try {
```

[Table of Contents] [Description]

Path 1160:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
990. public FifoQueue readLines(InputStream in) //nessusscan.java
       ...
1004. try {
```

[Table of Contents] [Description]

Path 1161:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
       ...
199. try {
```

[Table of Contents] [Description]

Path 1162:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. private Set getAllManagedInterfaces() //scheduler.java
       ...
222. try {
```

[Table of Contents] [Description]

Path 1163:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
448. public void run() //scheduler.java
       ...
478. try {
```

[Table of Contents] [Description]

Path 1164:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
448. public void run() //scheduler.java
       ...
511. try {
```

[Table of Contents] [Description]

Path 1165:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
448. public void run() //scheduler.java
       ...
538. try {
```

[Table of Contents] [Description]

Path 1166:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
69. public class Vulnscand implements PausableFiber //vulnscand.java
      ...
133. try {
```

[Table of Contents] [Description]

Path 1167:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. public void start() //vulnscand.java
      ...
181. try {
```

[Table of Contents] [Description]

Path 1168:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. public void start() //vulnscand.java
      ...
195. try {
```

[Table of Contents] [Description]

Path 1169:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. public void start() //vulnscand.java
      ...
228. try {
```

[Table of Contents] [Description]

Path 1170:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
173. public void start() //vulnscand.java
      ...
243. try {
```

[Table of Contents] [Description]

Path 1171:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      |
116. try {
```

[Table of Contents] [Description]

Path 1172:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      |
193. try {
```

[Table of Contents] [Description]

Path 1173:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      |
212. try {
```

[Table of Contents] [Description]

Path 1174:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
252. private void pushBackEvent(Event event) //eventqueueprocessor.java
      |
      ...
      |
256. try {
```

[Table of Contents] [Description]

Path 1175:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
278. private synchronized boolean statusOK() //eventqueueprocessor.java
      |
      ...
      |
307. try {
```

[Table of Contents] [Description]

Path 1176:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
413. public void run() //eventqueueprocessor.java
       ...
422. try {
```

[Table of Contents] [Description]

Path 1177:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
413. public void run() //eventqueueprocessor.java
       ...
441. try {
```

[Table of Contents] [Description]

Path 1178:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
       ...
121. try {
```

[Table of Contents] [Description]

Path 1179:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
       ...
354. try {
```

[Table of Contents] [Description]

Path 1180:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
       ...
372. try {
```

[Table of Contents] [Description]

Path 1181:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      ...
403. try {
```

[Table of Contents] [Description]

Path 1182:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
      ...
456. try {
```

[Table of Contents] [Description]

Path 1183:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
      ...
464. try {
```

[Table of Contents] [Description]

Path 1184:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
      ...
480. try {
```

[Table of Contents] [Description]

Path 1185:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      ...
348. try {
```

[Table of Contents] [Description]

Path 1186:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. public boolean equals(Object obj) //ipv4address.java
      |
      ...
      |
390. try {
```

[Table of Contents] [Description]

Path 1187:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
69. public void contextInitialized(ServletContextEvent
event) //initializerservletcontextlistener.java
      |
      ...
      |
84. try {
```

[Table of Contents] [Description]

Path 1188:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
92. public void contextDestroyed(ServletContextEvent
event) //initializerservletcontextlistener.java
      |
      ...
      |
95. try {
```

[Table of Contents] [Description]

Path 1189:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
288. protected void mailFileToUser(String filename, String emailAddr) throws
IOException //reportmailer.java
      |
      ...
      |
293. try {
```

[Table of Contents] [Description]

Path 1190:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
108. public synchronized static void init(ServletContext context) throws
ServletException //servletinitializer.java
```

```
...
118. try {
```

[Table of Contents] [Description]

Path 1191:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
238. try {
```

[Table of Contents] [Description]

Path 1192:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
...
240. try {
```

[Table of Contents] [Description]

Path 1193:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
92. public Object clone() //group.java
93. try {
```

[Table of Contents] [Description]

Path 1194:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
85. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewinterfaceservlet.java
...
89. try {
```

[Table of Contents] [Description]

Path 1195:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
107. private void createAndSendNewSuspectInterfaceEvent(String ipaddr) throws
ServletException //addnewinterfaceservlet.java
     ...
113. try {
```

[Table of Contents] [Description]

Path 1196:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. private int getNodeId(String ipaddr) throws
SQLException //addnewinterfaceservlet.java
     ...
147. try {
```

[Table of Contents] [Description]

Path 1197:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
129. private int getNodeId(String ipaddr) throws
SQLException //addnewinterfaceservlet.java
     ...
153. try {
```

[Table of Contents] [Description]

Path 1198:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletegetnodesservlet.java
     ...
73. try {
```

[Table of Contents] [Description]

Path 1199:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
86. private List getAllNodes(HttpSession userSession) throws
SQLException //deletegetnodesservlet.java
     ...
112. try {
```

[Table of Contents] [Description]

Path 1200:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
     ...
132. try {
```

[Table of Contents] [Description]

Path 1201:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
69. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteserviceservlet.java
     70. try {
```

[Table of Contents] [Description]

Path 1202:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getinterfacesservlet.java
     ...
85. try {
```

[Table of Contents] [Description]

Path 1203:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getinterfacesservlet.java
     ...
94. try {
```

[Table of Contents] [Description]

Path 1204:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. private List getInterfaces(HttpSession userSession, int nodeId) throws
SQLException //getinterfacesservlet.java
     ...
156. try {
```

[Table of Contents] [Description]

Path 1205:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getnodesservlet.java
    ...
83. try {
```

[Table of Contents] [Description]

Path 1206:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. private List getAllNodes(HttpSession userSession) throws
SQLException //getnodesservlet.java
    ...
159. try {
```

[Table of Contents] [Description]

Path 1207:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
    ...
128. try {
```

[Table of Contents] [Description]

Path 1208:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
292. private void writeURLFile(java.util.List interfaceList) throws
ServletException //managenodeservlet.java
    ...
298. try {
```

[Table of Contents] [Description]

Path 1209:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //managenodesservlet.java
    ...
128. try {
```

[Table of Contents] [Description]

Path 1210:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
292. private void writeURLFile(java.util.List interfaceList) throws
ServletException //managenodesservlet.java
    ...
298. try {
```

[Table of Contents] [Description]

Path 1211:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
76. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpgetinterfacesservlet.java
    ...
87. try {
```

[Table of Contents] [Description]

Path 1212:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
98. private List getNodeInterfaces(HttpSession userSession, int nodeid) throws
SQLException //snmpgetinterfacesservlet.java
    ...
130. try {
```

[Table of Contents] [Description]

Path 1213:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpgetnodesservlet.java
    ...
80. try {
```

[Table of Contents] [Description]

Path 1214:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. private List getAllNodes(HttpSession userSession) throws
SQLException //snmpgetnodesservlet.java
      ...
      126. try {
```

[Table of Contents] [Description]

Path 1215:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
      ...
      113. try {
```

[Table of Contents] [Description]

Path 1216:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. private void updateService(int nodeID, String interfaceIP, int serviceID, String
notifyFlag) throws ServletException //servicenoticeupdateservlet.java
      ...
      94. try {
```

[Table of Contents] [Description]

Path 1217:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. private void updateService(int nodeID, String interfaceIP, int serviceID, String
notifyFlag) throws ServletException //servicenoticeupdateservlet.java
      ...
      108. try {
```

[Table of Contents] [Description]

Path 1218:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      ...
      184. try {
```

[Table of Contents] [Description]

Path 1219:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
240. try {
```

[Table of Contents] [Description]

Path 1220:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
     ...
245. try {
```

[Table of Contents] [Description]

Path 1221:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public Object clone() //notificationinfo.java
     107. try {
```

[Table of Contents] [Description]

Path 1222:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
100. public Object clone() //user.java
     101. try {
```

[Table of Contents] [Description]

Path 1223:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
237. public static String encryptPassword(String aPassword) throws
IllegalStateException //user.java
     ...
240. try {
```

[Table of Contents] [Description]

Path 1224:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
220. public static String encryptPassword(String aPassword) throws
IllegalStateException //userinfo.java
     ...
223. try {
```

[Table of Contents] [Description]

Path 1225:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
85. public Object clone() //category.java
     86. try {
```

[Table of Contents] [Description]

Path 1226:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
110. public Category(String aDataString) //category.java
     ...
123. try {
```

[Table of Contents] [Description]

Path 1227:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. public Object clone() //view.java
     83. try {
```

[Table of Contents] [Description]

Path 1228:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
     ...
62. try {
```

[Table of Contents] [Description]

Path 1229:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
    ...
121. try {
```

[Table of Contents] [Description]

Path 1230:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
91. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //importassetsservlet.java
    ...
98. try {
```

[Table of Contents] [Description]

Path 1231:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
124. public List decodeAssetsText(String text) //importassetsservlet.java
    ...
134. try {
```

[Table of Contents] [Description]

Path 1232:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyassetservlet.java
    ...
83. try {
```

[Table of Contents] [Description]

Path 1233:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
510. public synchronized void start() throws
LifecycleException //opennmstomcatrealm.java
    ...
```

```
512. try {
```

[Table of Contents] [Description]

Path 1234:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
50. public void init() throws ServletException //rtcdebugservlet.java
    51. try {
```

[Table of Contents] [Description]

Path 1235:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
62. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcdebugservlet.java
    ...
    77. try {
```

[Table of Contents] [Description]

Path 1236:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
56. public void init() throws ServletException //rtcpostservlet.java
    57. try {
```

[Table of Contents] [Description]

Path 1237:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcpostservlet.java
    ...
    91. try {
```

[Table of Contents] [Description]

Path 1238:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
77. private EventProxy createEventProxy() //rtcpostsubscriber.java
    ...
```

```
86. try {
```

[Table of Contents] [Description]

Path 1239:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
    ...
106. try {
```

[Table of Contents] [Description]

Path 1240:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
    ...
101. try {
```

[Table of Contents] [Description]

Path 1241:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
    ...
112. try {
```

[Table of Contents] [Description]

Path 1242:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
    ...
123. try {
```

[Table of Contents] [Description]

Path 1243:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
      |
      ...
      129. try {
```

[Table of Contents] [Description]

Path 1244:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventqueryservlet.java
      |
      ...
      136. try {
```

[Table of Contents] [Description]

Path 1245:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventqueryservlet.java
      |
      ...
      146. try {
```

[Table of Contents] [Description]

Path 1246:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventqueryservlet.java
      |
      ...
      158. try {
```

[Table of Contents] [Description]

Path 1247:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
      |
      ...
      59. try {
```

[Table of Contents] [Description]

Path 1248:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //negativeservicefilter.java
      ...
59. try {
```

[Table of Contents] [Description]

Path 1249:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
      ...
59. try {
```

[Table of Contents] [Description]

Path 1250:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //servicefilter.java
      ...
59. try {
```

[Table of Contents] [Description]

Path 1251:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
80. public Object clone() //autoaction.java
      81. try {
```

[Table of Contents] [Description]

Path 1252:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
113. public Object clone() //correlation.java
      114. try {
```

[Table of Contents] [Description]

Path 1253:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
181. public Object clone() //event.java
     |182. try {
```

[Table of Contents] [Description]

Path 1254:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
96. public Object clone() //forward.java
     |97. try {
```

[Table of Contents] [Description]

Path 1255:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
65. public Object clone() //global.java
     |66. try {
```

[Table of Contents] [Description]

Path 1256:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
73. public Object clone() //maskelement.java
     |74. try {
```

[Table of Contents] [Description]

Path 1257:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
83. public Object clone() //operatoraction.java
     |84. try {
```

[Table of Contents] [Description]

Path 1258:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
81. public Object clone() //snmp.java
     |82. try {
```

[Table of Contents] [Description]

Path 1259:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
93. public void init() throws ServletException //rrdadhocgraphservlet.java
      94. try {
```

[Table of Contents] [Description]

Path 1260:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
120. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdadhocgraphservlet.java
      ...
      147. try {
```

[Table of Contents] [Description]

Path 1261:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public void init() throws ServletException //rrdgraphservlet.java
      ...
      121. try {
```

[Table of Contents] [Description]

Path 1262:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
117. public void init() throws ServletException //rrdgraphservlet.java
      ...
      141. try {
```

[Table of Contents] [Description]

Path 1263:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
      ...
      186. try {
```

[Table of Contents] [Description]

Path 1264:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
     ...
257. try {
```

[Table of Contents] [Description]

Path 1265:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
     ...
298. try {
```

[Table of Contents] [Description]

Path 1266:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
453. private Element createHostElement(MapNode n, boolean
loadIcons) //documentgenerator.java
     ...
518. try {
```

[Table of Contents] [Description]

Path 1267:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. public Vector getOpenNMSNodes() //mapnodefactory.java
     ...
119. try {
```

[Table of Contents] [Description]

Path 1268:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
```

```
...
92. try {
```

[Table of Contents] [Description]

Path 1269:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
285. public String getTextDescription() //noticefactory.java
...
287. try {
```

[Table of Contents] [Description]

Path 1270:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
365. public String getTextDescription() //noticefactory.java
...
367. try {
```

[Table of Contents] [Description]

Path 1271:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. public static boolean canDisplayEvent(int eventId) //noticefactory.java
...
466. try {
```

[Table of Contents] [Description]

Path 1272:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
462. public static boolean canDisplayEvent(int eventId) //noticefactory.java
...
487. try {
```

[Table of Contents] [Description]

Path 1273:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //noticequeryservlet.java
    ...
105. try {
```

[Table of Contents] [Description]

Path 1274:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noticequeryservlet.java
    ...
115. try {
```

[Table of Contents] [Description]

Path 1275:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noticequeryservlet.java
    ...
121. try {
```

[Table of Contents] [Description]

Path 1276:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
111. try {
```

[Table of Contents] [Description]

Path 1277:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
202. public Notification[] allNotifications() throws
SQLException //notificationmodel.java
    ...
207. try {
```

[Table of Contents] [Description]

Path 1278:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
     ...
235. try {
```

[Table of Contents] [Description]

Path 1279:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
306. public Notification[] getOutstandingNotices() throws
SQLException //notificationmodel.java
     ...
311. try {
```

[Table of Contents] [Description]

Path 1280:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
332. public int getOutstandingNoticeCount() throws
SQLException //notificationmodel.java
     ...
337. try {
```

[Table of Contents] [Description]

Path 1281:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
361. public int getOutstandingNoticeCount(String username) throws
SQLException //notificationmodel.java
     ...
370. try {
```

[Table of Contents] [Description]

Path 1282:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
     ...
400. try {
```

[Table of Contents] [Description]

Path 1283:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public void acknowledged(String name, int noticeId) throws
SQLException //notificationmodel.java
      ...
431. try {
```

[Table of Contents] [Description]

Path 1284:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
451. public void insert(Notification nbean) throws
SQLException //notificationmodel.java
      ...
458. try {
```

[Table of Contents] [Description]

Path 1285:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
277. public int execute() //command.java
      ...
302. try {
```

[Table of Contents] [Description]

Path 1286:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
      ...
107. try {
```

[Table of Contents] [Description]

Path 1287:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
```

```
...
117. try {
```

[Table of Contents] [Description]

Path 1288:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
...
123. try {
```

[Table of Contents] [Description]

Path 1289:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
...
59. try {
```

[Table of Contents] [Description]

Path 1290:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //negativeservicefilter.java
...
60. try {
```

[Table of Contents] [Description]

Path 1291:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
...
59. try {
```

[Table of Contents] [Description]

Path 1292:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //servicefilter.java
      ...
60. try {
```

[Table of Contents] [Description]

Path 1293:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
112. public static Properties loadLocal(String aFileName) //propertyloader.java
      ...
116. try {
```

[Table of Contents] [Description]

Path 1294:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
139. public static Properties loadLocal(String aFileName, Properties
someProperties) //propertyloader.java
      ...
143. try {
```

[Table of Contents] [Description]

Path 1295:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
189. public void backup() throws XMLWriteException //xmlwriter.java
190. try {
```

[Table of Contents] [Description]

Path 1296:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public Element addCDataElement(Element parent, String elementName, String data)
throws XMLWriteException //xmlwriter.java
      ...
248. try {
```

[Table of Contents] [Description]

Path 1297:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
228. public static RRDInfo createRRDInfo(int nodeId, String
infoString) //rrdinfo.java
     ...
236. try {
```

[Table of Contents] [Description]

Path 1298:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
309. protected static double parseDouble(String s) //rrdinfo.java
     ...
312. try {
```

[Table of Contents] [Description]

Path 1299:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
66. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
     ...
71. try {
```

[Table of Contents] [Description]

Path 1300:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
232. public static RRDInfo createRRDInfo(int nodeId, String
infoString) //rrdinfo.java
     ...
240. try {
```

[Table of Contents] [Description]

Path 1301:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
313. protected static double parseDouble(String s) //rrdinfo.java
     ...
316. try {
```

[Table of Contents] [Description]

Path 1302:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
    ...
75. try {
```

[Table of Contents] [Description]

Path 1303:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
    ...
113. try {
```

[Table of Contents] [Description]

Path 1304:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
    ...
124. try {
```

[Table of Contents] [Description]

Path 1305:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
    ...
130. try {
```

[Table of Contents] [Description]

Path 1306:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
    ...
59. try {
```

[Table of Contents] [Description]

Path 1307:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
      ...
59. try {
```

[Table of Contents] [Description]

Path 1308:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //interfacedeleted.jsp
      ...
58. try {
```

[Table of Contents] [Description]

Path 1309:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //servicedeleted.jsp
      ...
62. try {
```

[Table of Contents] [Description]

Path 1310:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //deletenode.jsp
      ...
51. try {
```

[Table of Contents] [Description]

Path 1311:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //index.jsp
      ...
52. try {
```

[Table of Contents] [Description]

Path 1312:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
18. public void sendOutagesChangedEvent() throws ServletException //editoutage.jsp
    ...
22. try {
```

[Table of Contents] [Description]

Path 1313:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
    ...
300. try {
```

[Table of Contents] [Description]

Path 1314:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
    ...
387. try {
```

[Table of Contents] [Description]

Path 1315:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
    ...
397. try {
```

[Table of Contents] [Description]

Path 1316:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
    ...
406. try {
```

[Table of Contents] [Description]

Path 1317:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
13. public void sendOutagesChangedEvent() throws ServletException //index.jsp
      ...
17. try {
```

[Table of Contents] [Description]

Path 1318:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //interface.jsp
      ...
107. try {
```

[Table of Contents] [Description]

Path 1319:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //service.jsp
      ...
63. try {
```

[Table of Contents] [Description]

Path 1320:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //detail.jsp
      ...
53. try {
```

[Table of Contents] [Description]

Path 1321:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //eventlist.jsp
      ...
69. try {
```

[Table of Contents] [Description]

Path 1322:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
54. public void init() throws ServletException //interfaceavailability-box.jsp
      55. try {
```

[Table of Contents] [Description]

Path 1323:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
55. public void init() throws ServletException //nodeavailability-box.jsp
      56. try {
```

[Table of Contents] [Description]

Path 1324:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
54. public void init() throws ServletException //serviceavailability-box.jsp
      55. try {
```

[Table of Contents] [Description]

Path 1325:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
128. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //include_rrd.jsp
      ...
      135. try {
```

[Table of Contents] [Description]

Path 1326:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
128. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //include_rrd.jsp
      ...
      135. try {
```

[Table of Contents] [Description]

Path 1327:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
128. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //include_rrd.jsp
      ...
135. try {
```

[Table of Contents] [Description]

Path 1328:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
128. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //include_rrd.jsp
      ...
135. try {
```

[Table of Contents] [Description]

Path 1329:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //detail.jsp
      ...
53. try {
```

[Table of Contents] [Description]

Path 1330:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //detail.jsp
      ...
57. try {
```

[Table of Contents] [Description]

Path 1331:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
474. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //results.jsp
      ...
481. try {
```

[Table of Contents] [Description]

Path 1332:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
312. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //zoom.jsp
      ...
319. try {
```

[Table of Contents] [Description]

Path 1333:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
476. public String getIfSpeed(int nodeId, String intf) throws
java.sql.SQLException //results.jsp
      ...
483. try {
```

[Table of Contents] [Description]

Path 1334:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
48. public void init() throws ServletException //category.jsp
      49. try {
```

[Table of Contents] [Description]

Path 1335:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
1. <!-- //detail.jsp
      ...
57. try {
```

[Table of Contents] [Description]

Path 1336:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. public void run() //inetechoxchange.java
      ...
111. try
```

[Table of Contents] [Description]

Path 1337:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
97. public void run() //inetechoxchange.java
      ...
122. try
```

[Table of Contents] [Description]

Path 1338:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
165. public void run() //inetechoxchange.java
      ...
173. try
```

[Table of Contents] [Description]

Path 1339:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
165. public void run() //inetechoxchange.java
      ...
185. try
```

[Table of Contents] [Description]

Path 1340:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
245. public static void main(String[] args) //inetechoxchange.java
      ...
281. try
```

[Table of Contents] [Description]

Path 1341:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
98. try
```

[Table of Contents] [Description]

Path 1342:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
109. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 1343:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
117. try { connection.close(); } catch(Throwable tx) { }
```

[Table of Contents] [Description]

Path 1344:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
130. try
```

[Table of Contents] [Description]

Path 1345:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
139. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 1346:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
145. try
```

[Table of Contents] [Description]

Path 1347:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
153. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 1348:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public void run() //sockettest.java
      ...
184. try
```

[Table of Contents] [Description]

Path 1349:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public void run() //sockettest.java
      ...
200. try
```

[Table of Contents] [Description]

Path 1350:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
179. public void run() //sockettest.java
      ...
223. try
```

[Table of Contents] [Description]

Path 1351:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public void run() //sockettest.java
      ...
285. try
```

[Table of Contents] [Description]

Path 1352:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
294. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 1353:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
309. try
```

[Table of Contents] [Description]

Path 1354:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
331. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 1355:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
341. try
```

[Table of Contents] [Description]

Path 1356:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
382. public void run() //sockettest.java
      |
      ...
388. try
```

[Table of Contents] [Description]

Path 1357:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public static void main(String[] args) //sockettest.java
      |...
      |438. try
```

[Table of Contents] [Description]

Path 1358:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
424. public static void main(String[] args) //sockettest.java
      |...
      |453. try
```

[Table of Contents] [Description]

Path 1359:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. void parseOptions(String[] args) throws
IllegalArgumentException //snmpwalk.java
      |...
      |201. try
```

[Table of Contents] [Description]

Path 1360:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. void parseOptions(String[] args) throws
IllegalArgumentException //snmpwalk.java
      |...
      |212. try
```

[Table of Contents] [Description]

Path 1361:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
176. void parseOptions(String[] args) throws
IllegalArgumentException //snmpwalk.java
      |...
```

```
224. try
```

[Table of Contents] [Description]

Path 1362:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
     ...
377. try
```

[Table of Contents] [Description]

Path 1363:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
     ...
417. try
```

[Table of Contents] [Description]

Path 1364:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
     ...
445. try
```

[Table of Contents] [Description]

Path 1365:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
261. void parseOptions(String[] args) throws
IllegalArgumentException //snmpwalkmv.java
     ...
286. try
```

[Table of Contents] [Description]

Path 1366:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
261. void parseOptions(String[] args) throws
```

```
IllegalArgumentException //snmpwalkmv.java
    ...
297. try
```

[Table of Contents] [Description]

Path 1367:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
261. void parseOptions(String[] args) throws
IllegalArgumentException //snmpwalkmv.java
    ...
309. try
```

[Table of Contents] [Description]

Path 1368:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
398. public void snmpReceivedPdu(SnmpSession session, int cmd, SnmpPduPacket
pdu) //snmpwalkmv.java
    ...
448. try
```

[Table of Contents] [Description]

Path 1369:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
    ...
515. try
```

[Table of Contents] [Description]

Path 1370:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
    ...
555. try
```

[Table of Contents] [Description]

Path 1371:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
       ...
588. try
```

[Table of Contents] [Description]

Path 1372:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
15. public SimulatorController(InetAddress controlChannel) throws
RuntimeException //simulatorcontroller.java
       ...
18. try {
```

[Table of Contents] [Description]

Path 1373:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
40. protected void sendCommand(String command) //simulatorcontroller.java
       ...
43. try {
```

[Table of Contents] [Description]

Path 1374:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
40. protected void sendCommand(String command) //simulatorcontroller.java
       ...
47. try {
```

[Table of Contents] [Description]

Path 1375:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
40. protected void sendCommand(String command) //simulatorcontroller.java
       ...
56. try {
```

[Table of Contents] [Description]

Path 1376:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
      ...
91. try {
```

[Table of Contents] [Description]

Path 1377:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
      ...
95. try {
```

[Table of Contents] [Description]

Path 1378:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
      ...
104. try {
```

[Table of Contents] [Description]

Path 1379:

Query Name - **Catch_Without_General_Exception_Handling**

Severity - **Information**

```
11. public static void main(String[] args) throws UnknownHostException,
java.io.IOException //tester.java
      ...
67. try {
```

[Table of Contents] [Description]

Incorrect Block Delimitation

CWE ID	483
Description	In some languages, forgetting to explicitly delimit a block can result in a logic error that can, in turn, have security implications.
Likelihood of Exploit	Low

Common Consequences	This is a general logic error -- with all the potential consequences that this entails.
Potential Mitigations	Implementation: Always use explicit block delimitation and use static-analysis technologies to enforce this practice.
Demonstrative Examples	In this example, when the condition is true, the intention may be that both x and y run. if (condition==true) x; y;
Context Notes	In many languages, braces are optional for blocks, and -- in a case where braces are omitted -- it is possible to insert a logic error where a statement is thought to be in a block but is not. This is a common and well known reliability error.
Node Relationships	ChildOf - Code Quality (398)
Source Taxonomies	CLASP - Incorrect block delimitation
Applicable Platforms	All

Path 1:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public String getMessage() //parseexception.java
       |
       ...
       |
139. if (i != 0) retval += " ";
```

[Table of Contents] [Description]

Path 2:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
       |
       ...
       |
47. if (arg.equalsIgnoreCase("-debug"))
```

[Table of Contents] [Description]

Path 3:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
       |
       ...
       |
64. if (setDebug) System.out.println("JavaCC Parser: Reading from file " + arg
+ " . . .");
```

[Table of Contents] [Description]

Path 4:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |64. if (setDebug) System.out.println("JavaCC Parser: Reading from file " + arg
      |+ " . . .");
```

[Table of Contents] [Description]

Path 5:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |67. if (firstFile)
```

[Table of Contents] [Description]

Path 6:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |67. if (firstFile)
```

[Table of Contents] [Description]

Path 7:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |78. if (setDebug)
```

[Table of Contents] [Description]

Path 8:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |78. if (setDebug)
```

[Table of Contents] [Description]

Path 9:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |
      ...
      89. if (setDebug) System.out.println("collect table information");
```

[Table of Contents] [Description]

Path 10:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |
      ...
      89. if (setDebug) System.out.println("collect table information");
```

[Table of Contents] [Description]

Path 11:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |
      ...
      93. if (setDebug) System.out.println("collect table index information");
```

[Table of Contents] [Description]

Path 12:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |
      ...
      93. if (setDebug) System.out.println("collect table index information");
```

[Table of Contents] [Description]

Path 13:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |
      ...
      97. if (setDebug) System.out.println("collect sequence info information");
```

[Table of Contents] [Description]

Path 14:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      97. if (setDebug) System.out.println("collect sequence info information");
```

[Table of Contents] [Description]

Path 15:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      101. if (setDebug) System.out.println("collect oids");
```

[Table of Contents] [Description]

Path 16:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      101. if (setDebug) System.out.println("collect oids");
```

[Table of Contents] [Description]

Path 17:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      105. if (setDebug) System.out.println("collect type information");
```

[Table of Contents] [Description]

Path 18:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      105. if (setDebug) System.out.println("collect type information");
```

[Table of Contents] [Description]

Path 19:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |115. if (setDebug) System.out.println("Thank you.");
```

[Table of Contents] [Description]

Path 20:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      |...
      |115. if (setDebug) System.out.println("Thank you.");
```

[Table of Contents] [Description]

Path 21:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
124. static final public SimpleNode Start() throws ParseException //parsemib.java
      |...
      |167. {if (true) return jjtn000;}
```

[Table of Contents] [Description]

Path 22:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
124. static final public SimpleNode Start() throws ParseException //parsemib.java
      |...
      |176. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 23:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
124. static final public SimpleNode Start() throws ParseException //parsemib.java
      |...
      |179. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 24:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
124. static final public SimpleNode Start() throws ParseException //parsemib.java
       ...
181. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 25:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
190. static final public void MibBegin() throws ParseException //parsemib.java
       ...
208. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 26:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
190. static final public void MibBegin() throws ParseException //parsemib.java
       ...
211. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 27:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
190. static final public void MibBegin() throws ParseException //parsemib.java
       ...
213. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 28:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. static final public void ModuleIdentity() throws ParseException //parsemib.java
       ...
335. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 29:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. static final public void ModuleIdentity() throws ParseException //parsemib.java
      ...
338. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 30:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. static final public void ModuleIdentity() throws ParseException //parsemib.java
      ...
340. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 31:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. static final public void Declaration() throws ParseException //parsemib.java
      ...
592. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 32:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. static final public void Declaration() throws ParseException //parsemib.java
      ...
595. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 33:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. static final public void Declaration() throws ParseException //parsemib.java
      ...
597. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 34:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
808. static final public void Assignment() throws ParseException //parsemib.java
      ...
877. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 35:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
808. static final public void Assignment() throws ParseException //parsemib.java
      ...
880. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 36:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
808. static final public void Assignment() throws ParseException //parsemib.java
      ...
882. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 37:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
890. static final public void SequenceOfVarsOrTextConvention() throws
ParseException //parsemib.java
      ...
914. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 38:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
890. static final public void SequenceOfVarsOrTextConvention() throws
ParseException //parsemib.java
      ...
917. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 39:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
890. static final public void SequenceOfVarsOrTextConvention() throws
ParseException //parsemib.java
      ...
      919. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 40:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
927. static final public void SequenceOfVars() throws ParseException //parsemib.java
      ...
      1007. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 41:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
927. static final public void SequenceOfVars() throws ParseException //parsemib.java
      ...
      1010. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 42:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
927. static final public void SequenceOfVars() throws ParseException //parsemib.java
      ...
      1012. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 43:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. static final public void TextConvention() throws
ParseException //parsemib.java
      ...
      1171. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 44:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. static final public void TextConvention() throws
ParseException //parsemib.java
      ...
1174. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 45:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. static final public void TextConvention() throws
ParseException //parsemib.java
      ...
1176. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 46:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1218. static final public void Expression() throws ParseException //parsemib.java
      ...
1233. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 47:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1218. static final public void Expression() throws ParseException //parsemib.java
      ...
1236. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 48:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1218. static final public void Expression() throws ParseException //parsemib.java
      ...
1238. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 49:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1335. static final public void MultipleOids() throws ParseException //parsemib.java
      ...
1353. {if (true) throw (RuntimeException)jjte000;}
```

[Table of Contents] [Description]

Path 50:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1335. static final public void MultipleOids() throws ParseException //parsemib.java
      ...
1356. {if (true) throw (ParseException)jjte000;}
```

[Table of Contents] [Description]

Path 51:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1335. static final public void MultipleOids() throws ParseException //parsemib.java
      ...
1358. {if (true) throw (Error)jjte000;}
```

[Table of Contents] [Description]

Path 52:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2585. static final private boolean jj_3_30() //parsemib.java
      2586. if (jj_scan_token(MODULE_COMPLIANCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 53:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2590. static final private boolean jj_3_89() //parsemib.java
      2591. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 54:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2590. static final private boolean jj_3_89() //parsemib.java
      |...
      |2592. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 55:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2590. static final private boolean jj_3_89() //parsemib.java
      |...
      |2593. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 56:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2597. static final private boolean jj_3_88() //parsemib.java
      |2598. if (jj_scan_token(NOTIFICATIONS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 57:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2597. static final private boolean jj_3_88() //parsemib.java
      |...
      |2599. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 58:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2597. static final private boolean jj_3_88() //parsemib.java
      |...
      |2601. if (jj_3_84()) return true;
```

[Table of Contents] [Description]

Path 59:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2609. static final private boolean jj_3_87() //parsemib.java
      2610. if (jj_scan_token(OBJECTS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 60:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2609. static final private boolean jj_3_87() //parsemib.java
      ...
      2611. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 61:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2609. static final private boolean jj_3_87() //parsemib.java
      ...
      2613. if (jj_3_82()) return true;
```

[Table of Contents] [Description]

Path 62:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2621. static final private boolean jj_3_80() //parsemib.java
      ...
      2632. if (jj_3_89()) return true;
```

[Table of Contents] [Description]

Path 63:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2640. static final private boolean jj_3_85() //parsemib.java
      2641. if (jj_scan_token(STATUS)) return true;
```

[Table of Contents] [Description]

Path 64:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2640. static final private boolean jj_3_85() //parsemib.java
```

```
...
2642. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 65:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2646. static final private boolean jj_3_66() //parsemib.java
      2647. if (jj_scan_token(AUGMENTS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 66:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2646. static final private boolean jj_3_66() //parsemib.java
      ...
      2648. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 67:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2646. static final private boolean jj_3_66() //parsemib.java
      ...
      2649. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 68:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2653. static final private boolean jj_3_52() //parsemib.java
      2654. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 69:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2658. static final private boolean jj_3_49() //parsemib.java
      ...
```

```
2663. if (jj_3_53()) return true;
```

[Table of Contents] [Description]

Path 70:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2668. static final private boolean jj_3_47() //parsemib.java
2669. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 71:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2668. static final private boolean jj_3_47() //parsemib.java
...
2672. if (jj_3_46()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 72:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2676. static final private boolean jj_3_59() //parsemib.java
2677. if (jj_scan_token(MAX_ACCESS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 73:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2681. static final private boolean jj_3_33() //parsemib.java
2682. if (jj_scan_token(DESCRIPTION)) return true;
```

[Table of Contents] [Description]

Path 74:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2686. static final private boolean jj_3_62() //parsemib.java
...
2693. if (jj_3_35()) return true;
```

[Table of Contents] [Description]

Path 75:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2686. static final private boolean jj_3_62() //parsemib.java
      ...
2696. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 76:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2700. static final private boolean jj_3_54() //parsemib.java
2701. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 77:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2700. static final private boolean jj_3_54() //parsemib.java
      ...
2702. if (jj_scan_token(IDENTIFIER_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 78:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2706. static final private boolean jj_3_51() //parsemib.java
2707. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 79:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2706. static final private boolean jj_3_51() //parsemib.java
      ...
2708. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 80:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2706. static final private boolean jj_3_51() //parsemib.java
      ...
      2709. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 81:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2713. static final private boolean jj_3_29() //parsemib.java
      2714. if (jj_scan_token(OBJECT_TYPE)) return true;
```

[Table of Contents] [Description]

Path 82:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2718. static final private boolean jj_3R_33() //parsemib.java
      2719. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 83:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2718. static final private boolean jj_3R_33() //parsemib.java
      ...
      2724. if (jj_3_79()) return true;
```

[Table of Contents] [Description]

Path 84:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2733. static final private boolean jj_3_41() //parsemib.java
      2734. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 85:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2738. static final private boolean jj_3_76() //parsemib.java
      2739. if (jj_3R_41()) return true;
```

[Table of Contents] [Description]

Path 86:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2743. static final private boolean jj_3_77() //parsemib.java
      2744. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 87:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2743. static final private boolean jj_3_77() //parsemib.java
      ...
      2745. if (jj_scan_token(IDENTIFIER_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 88:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2743. static final private boolean jj_3_77() //parsemib.java
      ...
      2746. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 89:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2750. static final private boolean jj_3_71() //parsemib.java
      2751. if (jj_3R_41()) return true;
```

[Table of Contents] [Description]

Path 90:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2755. static final private boolean jj_3_58() //parsemib.java
      2756. if (jj_scan_token(ACCESS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 91:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2760. static final private boolean jj_3_53() //parsemib.java
      2761. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 92:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2760. static final private boolean jj_3_53() //parsemib.java
      ...
      2763. if (jj_3_51()) return true;
```

[Table of Contents] [Description]

Path 93:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2771. static final private boolean jj_3_72() //parsemib.java
      2772. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 94:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2771. static final private boolean jj_3_72() //parsemib.java
      ...
      2773. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 95:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2771. static final private boolean jj_3_72() //parsemib.java
      ...
      2774. if (jj_3R_42()) return true;
```

[Table of Contents] [Description]

Path 96:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2778. static final private boolean jj_3_70() //parsemib.java
      ...
2785. if (jj_3_60()) return true;
```

[Table of Contents] [Description]

Path 97:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2778. static final private boolean jj_3_70() //parsemib.java
      ...
2788. if (jj_3R_40()) return true;
```

[Table of Contents] [Description]

Path 98:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2792. static final private boolean jj_3_57() //parsemib.java
2793. if (jj_3R_39()) return true;
```

[Table of Contents] [Description]

Path 99:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2802. static final private boolean jj_3_56() //parsemib.java
2803. if (jj_3R_38()) return true;
```

[Table of Contents] [Description]

Path 100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2807. static final private boolean jj_3_69() //parsemib.java
2808. if (jj_scan_token(SYNTAX_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 101:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
2807. static final private boolean jj_3_69() //parsemib.java
      |...
      |2817. if (jj_3_57()) return true;
```

[Table of Contents] [Description]

Path 102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2824. static final private boolean jj_3_68() //parsemib.java
      |2825. if (jj_scan_token(OBJECTS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2824. static final private boolean jj_3_68() //parsemib.java
      |...
      |2826. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2824. static final private boolean jj_3_68() //parsemib.java
      |...
      |2828. if (jj_3_47()) return true;
```

[Table of Contents] [Description]

Path 105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2836. static final private boolean jj_3_67() //parsemib.java
      |2837. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2836. static final private boolean jj_3_67() //parsemib.java
      |...
      |2838. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2842. static final private boolean jj_3_65() //parsemib.java
      |2843. if (jj_scan_token(GROUP_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2842. static final private boolean jj_3_65() //parsemib.java
      |...
      |2844. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2848. static final private boolean jj_3_64() //parsemib.java
      |2849. if (jj_scan_token(MODULE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2848. static final private boolean jj_3_64() //parsemib.java
      |...
      |2850. if (jj_scan_token(MANDATORY_GROUPS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2848. static final private boolean jj_3_64() //parsemib.java
      |
```

```
...
2851. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2855. static final private boolean jj_3_63() //parsemib.java
      2856. if (jj_scan_token(DEFVAL_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2855. static final private boolean jj_3_63() //parsemib.java
      ...
      2863. if (jj_3_43()) return true;
```

[Table of Contents] [Description]

Path 114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2869. static final private boolean jj_3_28() //parsemib.java
      2870. if (jj_scan_token(OBJECT_IDENTITY)) return true;
```

[Table of Contents] [Description]

Path 115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2874. static final private boolean jj_3_61() //parsemib.java
      2875. if (jj_scan_token(STATUS)) return true;
```

[Table of Contents] [Description]

Path 116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2874. static final private boolean jj_3_61() //parsemib.java
      ...
      2876. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2880. static final private boolean jj_3_32() //parsemib.java
      ...
2905. if (jj_3_72()) return true;
```

[Table of Contents] [Description]

Path 118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2920. static final private boolean jj_3_75() //parsemib.java
      ...
2929. if (jj_3_31()) return true;
```

[Table of Contents] [Description]

Path 119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2940. static final private boolean jj_3_148() //parsemib.java
2941. if (jj_scan_token(69)) return true;
```

[Table of Contents] [Description]

Path 120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2945. static final private boolean jj_3_12() //parsemib.java
2946. if (jj_scan_token(MODULE_IDENTITY_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2950. static final private boolean jj_3_151() //parsemib.java
2951. if (jj_scan_token(69)) return true;
```

[Table of Contents] [Description]

Path 122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2955. static final private boolean jj_3_15() //parsemib.java
      |2956. if (jj_scan_token(NOTIFICATION_GROUP_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2960. static final private boolean jj_3_18() //parsemib.java
      |2961. if (jj_scan_token(OBJECT_IDENTITY)) return true;
```

[Table of Contents] [Description]

Path 124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2965. static final private boolean jj_3R_40() //parsemib.java
      |2966. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2970. static final private boolean jj_3R_31() //parsemib.java
      |2971. if (jj_3R_48()) return true;
```

[Table of Contents] [Description]

Path 126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2970. static final private boolean jj_3R_31() //parsemib.java
      |...
      |2978. if (jj_3_77()) return true;
```

[Table of Contents] [Description]

Path 127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2984. static final private boolean jj_3_26() //parsemib.java
```

```
2985. if (jj_scan_token(REVISION_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2984. static final private boolean jj_3_26() //parsemib.java
      ...
2986. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2990. static final private boolean jj_3R_39() //parsemib.java
2991. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2995. static final private boolean jj_3_25() //parsemib.java
2996. if (jj_scan_token(CONTACT_INFO_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2995. static final private boolean jj_3_25() //parsemib.java
      ...
2997. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3001. static final private boolean jj_3_24() //parsemib.java
3002. if (jj_scan_token(DESCRIPTION)) return true;
```

[Table of Contents] [Description]

Path 133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3001. static final private boolean jj_3_24() //parsemib.java
      ...
3003. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3007. static final private boolean jj_3_22() //parsemib.java
3008. if (jj_scan_token(LAST_UPDATED_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3007. static final private boolean jj_3_22() //parsemib.java
      ...
3009. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3013. static final private boolean jj_3_23() //parsemib.java
3014. if (jj_scan_token(ORGANIZATION_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3013. static final private boolean jj_3_23() //parsemib.java
      ...
3015. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3019. static final private boolean jj_3_27() //parsemib.java
      ...
3030. if (jj_3_26()) return true;
```

[Table of Contents] [Description]

Path 139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3038. static final private boolean jj_3_147() //parsemib.java
3039. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3038. static final private boolean jj_3_147() //parsemib.java
      ...
3040. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3038. static final private boolean jj_3_147() //parsemib.java
      ...
3041. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3045. static final private boolean jj_3_11() //parsemib.java
3046. if (jj_scan_token(NOTIFICATION_TYPE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3050. static final private boolean jj_3_5() //parsemib.java
      |3051. if (jj_3R_33()) return true;
```

[Table of Contents] [Description]

Path 144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3055. static final private boolean jj_3_150() //parsemib.java
      |3056. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3055. static final private boolean jj_3_150() //parsemib.java
      |...
      |3057. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3055. static final private boolean jj_3_150() //parsemib.java
      |...
      |3058. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3062. static final private boolean jj_3_14() //parsemib.java
      |3063. if (jj_scan_token(OBJECT_GROUP_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3067. static final private boolean jj_3_17() //parsemib.java
      |3068. if (jj_scan_token(TEXTUAL_CONVENTION_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3072. static final private boolean jj_3R_44() //parsemib.java
      3073. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3077. static final private boolean jj_3_10() //parsemib.java
      3078. if (jj_scan_token(OBJECT_TYPE)) return true;
```

[Table of Contents] [Description]

Path 151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3082. static final private boolean jj_3_149() //parsemib.java
      3083. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3082. static final private boolean jj_3_149() //parsemib.java
      ...
      3086. if (jj_3_147()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3082. static final private boolean jj_3_149() //parsemib.java
      ...
      3088. if (jj_3_148()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3092. static final private boolean jj_3_152() //parsemib.java
      3093. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3092. static final private boolean jj_3_152() //parsemib.java
      ...
      3096. if (jj_3_150()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3092. static final private boolean jj_3_152() //parsemib.java
      ...
      3098. if (jj_3_151()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3102. static final private boolean jj_3_8() //parsemib.java
      3103. if (jj_3R_36()) return true;
```

[Table of Contents] [Description]

Path 158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3107. static final private boolean jj_3_140() //parsemib.java
      3108. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 159:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3112. static final private boolean jj_3_154() //parsemib.java
      ...
3114. if (jj_3_152()) return true;
```

[Table of Contents] [Description]

Path 160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3122. static final private boolean jj_3_153() //parsemib.java
3123. if (jj_scan_token(SIZE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3122. static final private boolean jj_3_153() //parsemib.java
      ...
3124. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3122. static final private boolean jj_3_153() //parsemib.java
      ...
3126. if (jj_3_149()) return true;
```

[Table of Contents] [Description]

Path 163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3134. static final private boolean jj_3_21() //parsemib.java
      ...
3136. if (jj_3_20()) return true;
```

[Table of Contents] [Description]

Path 164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3134. static final private boolean jj_3_21() //parsemib.java
      |...
      |3141. if (jj_scan_token(FROM)) return true;
```

[Table of Contents] [Description]

Path 165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3134. static final private boolean jj_3_21() //parsemib.java
      |...
      |3142. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3146. static final private boolean jj_3R_36() //parsemib.java
      |3147. if (jj_3R_48()) return true;
```

[Table of Contents] [Description]

Path 167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3146. static final private boolean jj_3R_36() //parsemib.java
      |...
      |3148. if (jj_scan_token(MODULE_IDENTITY_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3146. static final private boolean jj_3R_36() //parsemib.java
      |...
      |3150. if (jj_3_27()) return true;
```

[Table of Contents] [Description]

Path 169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3158. static final private boolean jj_3_16() //parsemib.java
```

```
3159. if (jj_scan_token(TRAP_TYPE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3163. static final private boolean jj_3_19() //parsemib.java
      3164. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. static final private boolean jj_3R_37() //parsemib.java
      3169. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. static final private boolean jj_3R_37() //parsemib.java
      ...
      3174. if (jj_3_154()) return true;
```

[Table of Contents] [Description]

Path 173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. static final private boolean jj_3R_37() //parsemib.java
      ...
      3176. if (jj_scan_token(RPAREN)) return true;
```

[Table of Contents] [Description]

Path 174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3180. static final private boolean jj_3_13() //parsemib.java
      3181. if (jj_scan_token(MODULE_COMPLIANCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3185. static final private boolean jj_3_9() //parsemib.java
      3186. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3190. static final private boolean jj_3_20() //parsemib.java
      ...
      3211. if (jj_3_18()) return true;
```

[Table of Contents] [Description]

Path 177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3228. static final private boolean jj_3_4() //parsemib.java
      3229. if (jj_3R_32()) return true;
```

[Table of Contents] [Description]

Path 178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3233. static final private boolean jj_3_7() //parsemib.java
      3234. if (jj_3R_35()) return true;
```

[Table of Contents] [Description]

Path 179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3238. static final private boolean jj_3R_30() //parsemib.java
      3239. if (jj_scan_token(IMPORTS)) return true;
```

[Table of Contents] [Description]

Path 180:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3238. static final private boolean jj_3R_30() //parsemib.java
      |...
      |3241. if (jj_3_21()) return true;
```

[Table of Contents] [Description]

Path 181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3249. static final private boolean jj_3R_43() //parsemib.java
      |3250. if (jj_3R_49()) return true;
```

[Table of Contents] [Description]

Path 182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3249. static final private boolean jj_3R_43() //parsemib.java
      |...
      |3251. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3249. static final private boolean jj_3R_43() //parsemib.java
      |...
      |3252. if (jj_3R_50()) return true;
```

[Table of Contents] [Description]

Path 184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3256. static final private boolean jj_3_6() //parsemib.java
      |3257. if (jj_3R_34()) return true;
```

[Table of Contents] [Description]

Path 185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3261. static final private boolean jj_3R_50() //parsemib.java
      3262. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3266. static final private boolean jj_3_2() //parsemib.java
      ...
      3279. if (jj_3_8()) return true;
```

[Table of Contents] [Description]

Path 187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3288. static final private boolean jj_3_3() //parsemib.java
      3289. if (jj_3R_31()) return true;
```

[Table of Contents] [Description]

Path 188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3293. static final private boolean jj_3_125() //parsemib.java
      3294. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3298. static final private boolean jj_3_1() //parsemib.java
      3299. if (jj_3R_30()) return true;
```

[Table of Contents] [Description]

Path 190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3303. static final private boolean jj_3R_49() //parsemib.java
      3304. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3308. static final private boolean jj_3_141() //parsemib.java
      3309. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3308. static final private boolean jj_3_141() //parsemib.java
      ...
      3310. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3308. static final private boolean jj_3_141() //parsemib.java
      ...
      3311. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3315. static final private boolean jj_3_126() //parsemib.java
      3316. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3320. static final private boolean jj_3R_42() //parsemib.java
      3321. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3325. static final private boolean jj_3_143() //parsemib.java
      |3326. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3325. static final private boolean jj_3_143() //parsemib.java
      |...
      |3328. if (jj_3_141()) return true;
```

[Table of Contents] [Description]

Path 198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3336. static final private boolean jj_3_132() //parsemib.java
      |3337. if (jj_scan_token(OCTET_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3336. static final private boolean jj_3_132() //parsemib.java
      |...
      |3338. if (jj_scan_token(STRING_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3336. static final private boolean jj_3_132() //parsemib.java
      |...
      |3341. if (jj_3_125()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3345. static final private boolean jj_3R_48() //parsemib.java
      |3346. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3350. static final private boolean jj_3_123() //parsemib.java
      |3351. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3355. static final private boolean jj_3_113() //parsemib.java
      |3356. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3360. static final private boolean jj_3_144() //parsemib.java
      |...
      |3365. if (jj_3_143()) return true;
```

[Table of Contents] [Description]

Path 205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3370. static final private boolean jj_3_142() //parsemib.java
      |3371. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3375. static final private boolean jj_3_128() //parsemib.java
      |3376. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3380. static final private boolean jj_3_124() //parsemib.java
      3381. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3380. static final private boolean jj_3_124() //parsemib.java
      ...
      3384. if (jj_3_123()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3388. static final private boolean jj_3_130() //parsemib.java
      ...
      3393. if (jj_3_129()) return true;
```

[Table of Contents] [Description]

Path 210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3398. static final private boolean jj_3_131() //parsemib.java
      3399. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3398. static final private boolean jj_3_131() //parsemib.java
      ...
      3400. if (jj_scan_token(IDENTIFIER_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3404. static final private boolean jj_3R_47() //parsemib.java
      3405. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3409. static final private boolean jj_3_122() //parsemib.java
      3410. if (jj_scan_token(REFERENCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3414. static final private boolean jj_3_127() //parsemib.java
      3415. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3414. static final private boolean jj_3_127() //parsemib.java
      ...
      3416. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3414. static final private boolean jj_3_127() //parsemib.java
      ...
      3417. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 217:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3421. static final private boolean jj_3_118() //parsemib.java
      3422. if (jj_scan_token(OCTET_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3421. static final private boolean jj_3_118() //parsemib.java
      ...
      3423. if (jj_scan_token(STRING_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3421. static final private boolean jj_3_118() //parsemib.java
      ...
      3426. if (jj_3_113()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3430. static final private boolean jj_3_112() //parsemib.java
      3431. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3430. static final private boolean jj_3_112() //parsemib.java
      ...
      3432. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3430. static final private boolean jj_3_112() //parsemib.java
      ...
      3433. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3437. static final private boolean jj_3_133() //parsemib.java
      3438. if (jj_3R_39()) return true;
```

[Table of Contents] [Description]

Path 224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3437. static final private boolean jj_3_133() //parsemib.java
      ...
      3441. if (jj_3_130()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3445. static final private boolean jj_3_146() //parsemib.java
    3446. if (jj_scan_token(OCTET_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3445. static final private boolean jj_3_146() //parsemib.java
      ...
      3447. if (jj_scan_token(STRING_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3445. static final private boolean jj_3_146() //parsemib.java
      ...
      3450. if (jj_3_144()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3454. static final private boolean jj_3_129() //parsemib.java
      3455. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3454. static final private boolean jj_3_129() //parsemib.java
      ...
      3457. if (jj_3_127()) return true;
```

[Table of Contents] [Description]

Path 230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3465. static final private boolean jj_3_138() //parsemib.java
      3466. if (jj_scan_token(SYNTAX_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3465. static final private boolean jj_3_138() //parsemib.java
      ...
      3473. if (jj_3_133()) return true;
```

[Table of Contents] [Description]

Path 232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3479. static final private boolean jj_3_109() //parsemib.java
      3480. if (jj_3R_46()) return true;
```

[Table of Contents] [Description]

Path 233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3484. static final private boolean jj_3_137() //parsemib.java
      3485. if (jj_scan_token(OBJECTS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3484. static final private boolean jj_3_137() //parsemib.java
      ...
      3486. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3484. static final private boolean jj_3_137() //parsemib.java
      |...
      |3488. if (jj_3_124()) return true;
```

[Table of Contents] [Description]

Path 236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3496. static final private boolean jj_3_121() //parsemib.java
      |3497. if (jj_scan_token(DESCRIPTION)) return true;
```

[Table of Contents] [Description]

Path 237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3501. static final private boolean jj_3_136() //parsemib.java
      |3502. if (jj_scan_token(DISPLAY_HINT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3501. static final private boolean jj_3_136() //parsemib.java
      |...
      |3503. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3507. static final private boolean jj_3_135() //parsemib.java
      |...
      |3512. if (jj_3_122()) return true;
```

[Table of Contents] [Description]

Path 240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3507. static final private boolean jj_3_135() //parsemib.java
      |
```

```
...
3514. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3518. static final private boolean jj_3_134() //parsemib.java
      3519. if (jj_scan_token(STATUS)) return true;
```

[Table of Contents] [Description]

Path 242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3518. static final private boolean jj_3_134() //parsemib.java
...
3520. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3524. static final private boolean jj_3_139() //parsemib.java
...
3535. if (jj_3_138()) return true;
```

[Table of Contents] [Description]

Path 244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3543. static final private boolean jj_3_103() //parsemib.java
      3544. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3548. static final private boolean jj_3_145() //parsemib.java
      3549. if (jj_scan_token(TEXTUAL_CONVENTION_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3548. static final private boolean jj_3_145() //parsemib.java
      ...
3551. if (jj_3_139()) return true;
```

[Table of Contents] [Description]

Path 247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3559. static final private boolean jj_3_116() //parsemib.java
3560. if (jj_scan_token(OBJECT_TYPE)) return true;
```

[Table of Contents] [Description]

Path 248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3564. static final private boolean jj_3_110() //parsemib.java
      ...
3569. if (jj_3_112()) return true;
```

[Table of Contents] [Description]

Path 249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3574. static final private boolean jj_3_111() //parsemib.java
3575. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3579. static final private boolean jj_3_108() //parsemib.java
3580. if (jj_3R_45()) return true;
```

[Table of Contents] [Description]

Path 251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3584. static final private boolean jj_3_102() //parsemib.java
      |3585. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3584. static final private boolean jj_3_102() //parsemib.java
      |...
      |3586. if (jj_scan_token(IDENTIFIER_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3590. static final private boolean jj_3R_46() //parsemib.java
      |...
      |3595. if (jj_3_146()) return true;
```

[Table of Contents] [Description]

Path 254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3600. static final private boolean jj_3_119() //parsemib.java
      |3601. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3605. static final private boolean jj_3_114() //parsemib.java
      |3606. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3615. static final private boolean jj_3_117() //parsemib.java
      3616. if (jj_scan_token(OBJECT_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3615. static final private boolean jj_3_117() //parsemib.java
      ...
      3617. if (jj_scan_token(IDENTIFIER_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3621. static final private boolean jj_3_115() //parsemib.java
      3622. if (jj_scan_token(OBJECT_IDENTITY)) return true;
```

[Table of Contents] [Description]

Path 259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3626. static final private boolean jj_3_120() //parsemib.java
      3627. if (jj_3R_47()) return true;
```

[Table of Contents] [Description]

Path 260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3626. static final private boolean jj_3_120() //parsemib.java
      ...
      3638. if (jj_3_118()) return true;
```

[Table of Contents] [Description]

Path 261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3650. static final private boolean jj_3R_45() //parsemib.java
      3651. if (jj_scan_token(SEQUENCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3650. static final private boolean jj_3R_45() //parsemib.java
      ...
3652. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3650. static final private boolean jj_3R_45() //parsemib.java
      ...
3654. if (jj_3_120()) return true;
```

[Table of Contents] [Description]

Path 264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3662. static final private boolean jj_3_100() //parsemib.java
      3663. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3662. static final private boolean jj_3_100() //parsemib.java
      ...
3664. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3668. static final private boolean jj_3_101() //parsemib.java
      3669. if (jj_3R_39()) return true;
```

[Table of Contents] [Description]

Path 267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3673. static final private boolean jj_3R_32() //parsemib.java
      3674. if (jj_3R_48()) return true;
```

[Table of Contents] [Description]

Path 268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3673. static final private boolean jj_3R_32() //parsemib.java
      ...
      3675. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3673. static final private boolean jj_3R_32() //parsemib.java
      ...
      3680. if (jj_3_109()) return true;
```

[Table of Contents] [Description]

Path 270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3685. static final private boolean jj_3_104() //parsemib.java
      3686. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3685. static final private boolean jj_3_104() //parsemib.java
      ...
      3687. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3685. static final private boolean jj_3_104() //parsemib.java
      ...
3688. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3692. static final private boolean jj_3_105() //parsemib.java
3693. if (jj_scan_token(LPAREN)) return true;
```

[Table of Contents] [Description]

Path 274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3692. static final private boolean jj_3_105() //parsemib.java
      ...
3694. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3692. static final private boolean jj_3_105() //parsemib.java
      ...
3695. if (jj_scan_token(68)) return true;
```

[Table of Contents] [Description]

Path 276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3699. static final private boolean jj_3_106() //parsemib.java
3700. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3699. static final private boolean jj_3_106() //parsemib.java
      ...
3702. if (jj_3_104()) return true;
```

[Table of Contents] [Description]

Path 278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3710. static final private boolean jj_3_107() //parsemib.java
      ...
3715. if (jj_3_106()) return true;
```

[Table of Contents] [Description]

Path 279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3720. static final private boolean jj_3R_35() //parsemib.java
3721. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3720. static final private boolean jj_3R_35() //parsemib.java
      ...
3722. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3720. static final private boolean jj_3R_35() //parsemib.java
      ...
3727. if (jj_3_102()) return true;
```

[Table of Contents] [Description]

Path 282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3732. static final private boolean jj_3_43() //parsemib.java
```

```
3733. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3737. static final private boolean jj_3R_41() //parsemib.java
3738. if (jj_scan_token(INDEX_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3737. static final private boolean jj_3R_41() //parsemib.java
...
3739. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3737. static final private boolean jj_3R_41() //parsemib.java
...
3740. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3744. static final private boolean jj_3_94() //parsemib.java
3745. if (jj_scan_token(RBRACE)) return true;
```

[Table of Contents] [Description]

Path 287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3749. static final private boolean jj_3_40() //parsemib.java
3750. if (jj_scan_token(RBRACE)) return true;
```

[Table of Contents] [Description]

Path 288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3754. static final private boolean jj_3_91() //parsemib.java
      |3755. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3759. static final private boolean jj_3_48() //parsemib.java
      |3760. if (jj_3R_37()) return true;
```

[Table of Contents] [Description]

Path 290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3764. static final private boolean jj_3_92() //parsemib.java
      |3765. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3764. static final private boolean jj_3_92() //parsemib.java
      |...
      |3768. if (jj_3_91()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3772. static final private boolean jj_3_83() //parsemib.java
      |3773. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 293:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3777. static final private boolean jj_3_96() //parsemib.java
      3778. if (jj_scan_token(DESCRIPTION)) return true;
```

[Table of Contents] [Description]

Path 294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3777. static final private boolean jj_3_96() //parsemib.java
      ...
      3779. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3783. static final private boolean jj_3_38() //parsemib.java
      3784. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3788. static final private boolean jj_3_74() //parsemib.java
      3789. if (jj_3R_44()) return true;
```

[Table of Contents] [Description]

Path 297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3793. static final private boolean jj_3_81() //parsemib.java
      3794. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3798. static final private boolean jj_3_79() //parsemib.java
      3799. if (jj_scan_token(NOTIFICATION_GROUP_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3803. static final private boolean jj_3_50() //parsemib.java
      |3804. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3808. static final private boolean jj_3_84() //parsemib.java
      |3809. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3808. static final private boolean jj_3_84() //parsemib.java
      |...
      |3812. if (jj_3_83()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3816. static final private boolean jj_3_93() //parsemib.java
      |3817. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3816. static final private boolean jj_3_93() //parsemib.java
      |...
      |3818. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3822. static final private boolean jj_3_44() //parsemib.java
      3823. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3827. static final private boolean jj_3_31() //parsemib.java
      3828. if (jj_scan_token(OBJECT_GROUP_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3832. static final private boolean jj_3R_38() //parsemib.java
      3833. if (jj_scan_token(SEQUENCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3832. static final private boolean jj_3R_38() //parsemib.java
      ...
      3834. if (jj_scan_token(OF_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3832. static final private boolean jj_3R_38() //parsemib.java
      ...
      3835. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 309:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3839. static final private boolean jj_3_39() //parsemib.java
      ...
3844. if (jj_3_38()) return true;
```

[Table of Contents] [Description]

Path 310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3849. static final private boolean jj_3_37() //parsemib.java
3850. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3854. static final private boolean jj_3_35() //parsemib.java
3855. if (jj_scan_token(UNITS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3859. static final private boolean jj_3_82() //parsemib.java
3860. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3859. static final private boolean jj_3_82() //parsemib.java
      ...
3863. if (jj_3_81()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3867. static final private boolean jj_3_55() //parsemib.java
```

```
3868. if (jj_scan_token(OCTET_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3867. static final private boolean jj_3_55() //parsemib.java
      ...
3869. if (jj_scan_token(STRING_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3867. static final private boolean jj_3_55() //parsemib.java
      ...
3872. if (jj_3_48()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3876. static final private boolean jj_3_99() //parsemib.java
3877. if (jj_scan_token(ASSIGN)) return true;
```

[Table of Contents] [Description]

Path 318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3876. static final private boolean jj_3_99() //parsemib.java
      ...
3880. if (jj_3_93()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3876. static final private boolean jj_3_99() //parsemib.java
      ...
```

```
3881. if (jj_scan_token(INTEGER_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3876. static final private boolean jj_3_99() //parsemib.java
...
3883. if (jj_3_94()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3887. static final private boolean jj_3_73() //parsemib.java
3888. if (jj_3R_43()) return true;
```

[Table of Contents] [Description]

Path 322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3892. static final private boolean jj_3_86() //parsemib.java
3893. if (jj_scan_token(DESCRIPTION)) return true;
```

[Table of Contents] [Description]

Path 323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3892. static final private boolean jj_3_86() //parsemib.java
...
3894. if (jj_scan_token(DESCRIPTOR_LITERAL)) return true;
```

[Table of Contents] [Description]

Path 324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3898. static final private boolean jj_3_98() //parsemib.java
3899. if (jj_scan_token(VARIABLES_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3898. static final private boolean jj_3_98() //parsemib.java
      ...
3900. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3898. static final private boolean jj_3_98() //parsemib.java
      ...
3902. if (jj_3_92()) return true;
```

[Table of Contents] [Description]

Path 327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3910. static final private boolean jj_3_97() //parsemib.java
      3911. if (jj_scan_token(ENTERPRISE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3910. static final private boolean jj_3_97() //parsemib.java
      ...
3912. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3916. static final private boolean jj_3_90() //parsemib.java
      ...
3927. if (jj_3_99()) return true;
```

[Table of Contents] [Description]

Path 330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3935. static final private boolean jj_3_95() //parsemib.java
      3936. if (jj_scan_token(STATUS)) return true;
```

[Table of Contents] [Description]

Path 331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3935. static final private boolean jj_3_95() //parsemib.java
      ...
      3937. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3941. static final private boolean jj_3_45() //parsemib.java
      3942. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3941. static final private boolean jj_3_45() //parsemib.java
      ...
      3945. if (jj_3_44()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3949. static final private boolean jj_3_36() //parsemib.java
      3950. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 335:

Query Name - **Single_Line_If_Statement**

Severity

- **Information**

```
3954. static final private boolean jj_3_60() //parsemib.java
      | 3955. if (jj_scan_token(MIN_ACCESS_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3959. static final private boolean jj_3_46() //parsemib.java
      | 3960. if (jj_scan_token(COMMA)) return true;
```

[Table of Contents] [Description]

Path 337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3964. static final private boolean jj_3_78() //parsemib.java
      | 3965. if (jj_scan_token(NOTIFICATION_TYPE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3969. static final private boolean jj_3R_34() //parsemib.java
      | 3970. if (jj_scan_token(IDENTIFIER)) return true;
```

[Table of Contents] [Description]

Path 339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3969. static final private boolean jj_3R_34() //parsemib.java
      | ...
      | 3971. if (jj_scan_token(TRAP_TYPE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3980. static final private boolean jj_3_42() //parsemib.java
      | 3981. if (jj_scan_token(LBRACE)) return true;
```

[Table of Contents] [Description]

Path 341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3980. static final private boolean jj_3_42() //parsemib.java
      ...
3984. if (jj_3_36()) jj_scanpos = xsp;
```

[Table of Contents] [Description]

Path 342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3980. static final private boolean jj_3_42() //parsemib.java
      ...
3985. if (jj_3_39()) return true;
```

[Table of Contents] [Description]

Path 343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3980. static final private boolean jj_3_42() //parsemib.java
      ...
3990. if (jj_scan_token(RBRACE)) return true;
```

[Table of Contents] [Description]

Path 344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3994. static final private boolean jj_3_34() //parsemib.java
      3995. if (jj_scan_token(REFERENCE_TOKEN)) return true;
```

[Table of Contents] [Description]

Path 345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4113. static final private Token jj_consume_token(int kind) throws
ParseException //parsemib.java
      ...
4115. if ((oldToken = token).next != null) token = token.next;
```

[Table of Contents] [Description]

Path 346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4113. static final private Token jj_consume_token(int kind) throws
ParseException //parsemib.java
      ...
4125. if (c.gen < jj_gen) c.first = null;
```

[Table of Contents] [Description]

Path 347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4139. static final private boolean jj_scan_token(int kind) //parsemib.java
      ...
4153. if (tok != null) jj_add_error_token(kind, i);
```

[Table of Contents] [Description]

Path 348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4139. static final private boolean jj_scan_token(int kind) //parsemib.java
      ...
4153. if (tok != null) jj_add_error_token(kind, i);
```

[Table of Contents] [Description]

Path 349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4139. static final private boolean jj_scan_token(int kind) //parsemib.java
      ...
4155. if (jj_scanpos.kind != kind) return true;
```

[Table of Contents] [Description]

Path 350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4139. static final private boolean jj_scan_token(int kind) //parsemib.java
      ...
4156. if (jj_la == 0 && jj_scanpos == jj_lastpos) throw jj_ls;
```

[Table of Contents] [Description]

Path 351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4160. static final public Token getNextToken() //parsemib.java
      4161. if (token.next != null) token = token.next;
```

[Table of Contents] [Description]

Path 352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4168. static final public Token getToken(int index) //parsemib.java
      ...
      4171. if (t.next != null) t = t.next;
```

[Table of Contents] [Description]

Path 353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4177. static final private int jj_ntk() //parsemib.java
      4178. if ((jj_nt=token.next) == null)
```

[Table of Contents] [Description]

Path 354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4190. static private void jj_add_error_token(int kind, int pos) //parsemib.java
      4191. if (pos >= 100) return;
```

[Table of Contents] [Description]

Path 355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4190. static private void jj_add_error_token(int kind, int pos) //parsemib.java
      ...
      4210. if (exists) break;
```

[Table of Contents] [Description]

Path 356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4190. static private void jj_add_error_token(int kind, int pos) //parsemib.java
      ...
4213. if (!exists) jj_expentries.addElement(jj_expentry);
```

[Table of Contents] [Description]

Path 357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4190. static private void jj_add_error_token(int kind, int pos) //parsemib.java
      ...
4213. if (!exists) jj_expentries.addElement(jj_expentry);
```

[Table of Contents] [Description]

Path 358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4190. static private void jj_add_error_token(int kind, int pos) //parsemib.java
      ...
4214. if (pos != 0) jj_lasttokens[(jj_endpos = pos) - 1] = kind;
```

[Table of Contents] [Description]

Path 359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      ...
41. if ((active0 & 0x200000000000000L) != 0L)
```

[Table of Contents] [Description]

Path 360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      ...
51. if ((active0 & 0x20000000L) != 0L)
```

[Table of Contents] [Description]

Path 361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |
      ...
      61. if ((active0 & 0x4000L) != 0L)
```

[Table of Contents] [Description]

Path 362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |
      ...
      65. if ((active0 & 0x8000100L) != 0L)
```

[Table of Contents] [Description]

Path 363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |
      ...
      81. if ((active0 & 0x10440010080L) != 0L)
```

[Table of Contents] [Description]

Path 364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |
      ...
      94. if ((active0 & 0xf20980829200L) != 0L)
```

[Table of Contents] [Description]

Path 365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |
```

```
...
98. if ((active0 & 0x800000000040L) != 0L)
```

[Table of Contents] [Description]

Path 366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
...
114. if ((active0 & 0x14002000L) != 0L)
```

[Table of Contents] [Description]

Path 367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
...
124. if ((active0 & 0x80000200800L) != 0L)
```

[Table of Contents] [Description]

Path 368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
...
134. if ((active0 & 0x10400000c0000L) != 0L)
```

[Table of Contents] [Description]

Path 369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
...
144. if ((active0 & 0x100000000400L) != 0L)
```

[Table of Contents] [Description]

Path 370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |...
      |154. if ((active0 & 0x8803400000L) != 0L)
```

[Table of Contents] [Description]

Path 371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |...
      |164. if ((active0 & 0x2000000000L) != 0L)
```

[Table of Contents] [Description]

Path 372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |...
      |182. if ((active0 & 0x200000800000L) != 0L)
```

[Table of Contents] [Description]

Path 373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |...
      |192. if ((active0 & 0x200000000L) != 0L)
```

[Table of Contents] [Description]

Path 374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
31. private static final int jjStopStringLiteralDfa_0(int pos, long active0, long
active1) //parsemibtokenmanager.java
      |...
```

```
202. if ((active0 & 0x80100000L) != 0L)
```

[Table of Contents] [Description]

Path 375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
293. static private final int jjMoveStringLiteralDfa1_0(long active0, long
active1) //parsemibtokenmanager.java
...
315. if ((active0 & 0x20000000L) != 0L)
```

[Table of Contents] [Description]

Path 376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. static private final int jjMoveStringLiteralDfa2_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
...
341. if (((active0 &= old0) | (active1 &= old1)) == 0L)
```

[Table of Contents] [Description]

Path 377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. static private final int jjMoveStringLiteralDfa2_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
...
351. if ((active1 & 0x8L) != 0L)
```

[Table of Contents] [Description]

Path 378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. static private final int jjMoveStringLiteralDfa2_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
...
359. if ((active0 & 0x4000L) != 0L)
```

[Table of Contents] [Description]

Path 379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. static private final int jjMoveStringLiteralDfa3_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
     ...
399. if (((active0 &= old0) | (active1 &= old1)) == 0L)
```

[Table of Contents] [Description]

Path 380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. static private final int jjMoveStringLiteralDfa3_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
     ...
417. if ((active0 & 0x8000000L) != 0L)
```

[Table of Contents] [Description]

Path 381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. static private final int jjMoveStringLiteralDfa3_0(long old0, long active0,
long old1, long active1) //parsemibtokenmanager.java
     ...
423. if ((active0 & 0x100L) != 0L)
```

[Table of Contents] [Description]

Path 382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
     ...
445. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
     ...
467. if ((active0 & 0x80L) != 0L)
```

[Table of Contents] [Description]

Path 384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
471. if ((active0 & 0x400000000L) != 0L)
```

[Table of Contents] [Description]

Path 385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
477. if ((active0 & 0x10000L) != 0L)
```

[Table of Contents] [Description]

Path 386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
481. if ((active0 & 0x10000000000L) != 0L)
```

[Table of Contents] [Description]

Path 387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. static private final int jjMoveStringLiteralDfa4_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
487. if ((active0 & 0x40000000L) != 0L)
```

[Table of Contents] [Description]

Path 388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
497. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
520. if ((active0 & 0x20000000000L) != 0L)
```

[Table of Contents] [Description]

Path 390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
526. if ((active0 & 0x1000L) != 0L)
```

[Table of Contents] [Description]

Path 391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
534. if ((active0 & 0x200L) != 0L)
```

[Table of Contents] [Description]

Path 392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
536. else if ((active0 & 0x20000L) != 0L)
```

[Table of Contents] [Description]

Path 393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
495. static private final int jjMoveStringLiteralDfa5_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
549. if ((active0 & 0x8000L) != 0L)
```

[Table of Contents] [Description]

Path 394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. static private final int jjMoveStringLiteralDfa6_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
559. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. static private final int jjMoveStringLiteralDfa6_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
585. if ((active0 & 0x40L) != 0L)
```

[Table of Contents] [Description]

Path 396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. static private final int jjMoveStringLiteralDfa6_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
587. else if ((active0 & 0x800000000000L) != 0L)
```

[Table of Contents] [Description]

Path 397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
601. static private final int jjMoveStringLiteralDfa7_0(long old0, long
```

```
active0) //parsemibtokenmanager.java
    ...
603. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
601. static private final int jjMoveStringLiteralDfa7_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
621. if ((active0 & 0x10000000L) != 0L)
```

[Table of Contents] [Description]

Path 399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
601. static private final int jjMoveStringLiteralDfa7_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
629. if ((active0 & 0x4000000L) != 0L)
```

[Table of Contents] [Description]

Path 400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
601. static private final int jjMoveStringLiteralDfa7_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
637. if ((active0 & 0x2000L) != 0L)
```

[Table of Contents] [Description]

Path 401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
647. static private final int jjMoveStringLiteralDfa8_0(long old0, long
active0) //parsemibtokenmanager.java
    ...
649. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
647. static private final int jjMoveStringLiteralDfa8_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |665. if ((active0 & 0x800L) != 0L)
```

[Table of Contents] [Description]

Path 403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
647. static private final int jjMoveStringLiteralDfa8_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |667. else if ((active0 & 0x200000L) != 0L)
```

[Table of Contents] [Description]

Path 404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
647. static private final int jjMoveStringLiteralDfa8_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |679. if ((active0 & 0x80000000000L) != 0L)
```

[Table of Contents] [Description]

Path 405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
      |...
      |693. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
      |
```

```
...
705. if ((active0 & 0x40000000000L) != 0L)
```

[Table of Contents] [Description]

Path 407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
...
719. if ((active0 & 0x1000000000000L) != 0L)
```

[Table of Contents] [Description]

Path 408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
...
723. if ((active0 & 0x40000L) != 0L)
```

[Table of Contents] [Description]

Path 409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
691. static private final int jjMoveStringLiteralDfa9_0(long old0, long
active0) //parsemibtokenmanager.java
...
725. else if ((active0 & 0x80000L) != 0L)
```

[Table of Contents] [Description]

Path 410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. static private final int jjMoveStringLiteralDfa10_0(long old0, long
active0) //parsemibtokenmanager.java
...
737. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. static private final int jjMoveStringLiteralDfa10_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
747. if ((active0 & 0x100000000000L) != 0L)
```

[Table of Contents] [Description]

Path 412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. static private final int jjMoveStringLiteralDfa10_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
755. if ((active0 & 0x400L) != 0L)
```

[Table of Contents] [Description]

Path 413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
771. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
781. if ((active0 & 0x400000L) != 0L)
```

[Table of Contents] [Description]

Path 415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
```

```
787. if ((active0 & 0x1000000L) != 0L)
```

[Table of Contents] [Description]

Path 416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
...
791. if ((active0 & 0x2000000L) != 0L)
```

[Table of Contents] [Description]

Path 417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
...
795. if ((active0 & 0x800000000L) != 0L)
```

[Table of Contents] [Description]

Path 418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. static private final int jjMoveStringLiteralDfa11_0(long old0, long
active0) //parsemibtokenmanager.java
...
801. if ((active0 & 0x8000000000L) != 0L)
```

[Table of Contents] [Description]

Path 419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
811. static private final int jjMoveStringLiteralDfa12_0(long old0, long
active0) //parsemibtokenmanager.java
...
813. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
811. static private final int jjMoveStringLiteralDfa12_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      831. if ((active0 & 0x2000000000L) != 0L)
```

[Table of Contents] [Description]

Path 421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
839. static private final int jjMoveStringLiteralDfa13_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      841. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
865. static private final int jjMoveStringLiteralDfa14_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      867. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
865. static private final int jjMoveStringLiteralDfa14_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      885. if ((active0 & 0x800000L) != 0L)
```

[Table of Contents] [Description]

Path 424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
865. static private final int jjMoveStringLiteralDfa14_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
      887. else if ((active0 & 0x200000000000L) != 0L)
```

[Table of Contents] [Description]

Path 425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
895. static private final int jjMoveStringLiteralDfa15_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
897. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
895. static private final int jjMoveStringLiteralDfa15_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
915. if ((active0 & 0x200000000L) != 0L)
```

[Table of Contents] [Description]

Path 427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
923. static private final int jjMoveStringLiteralDfa16_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
925. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
923. static private final int jjMoveStringLiteralDfa16_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
935. if ((active0 & 0x100000L) != 0L)
```

[Table of Contents] [Description]

Path 429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
923. static private final int jjMoveStringLiteralDfa16_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
937. else if ((active0 & 0x80000000L) != 0L)
```

[Table of Contents] [Description]

Path 430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
949. static private final int jjMoveStringLiteralDfa17_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
951. if (((active0 &= old0)) == 0L)
```

[Table of Contents] [Description]

Path 431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
949. static private final int jjMoveStringLiteralDfa17_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
961. if ((active0 & 0x4000000000L) != 0L)
```

[Table of Contents] [Description]

Path 432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
949. static private final int jjMoveStringLiteralDfa17_0(long old0, long
active0) //parsemibtokenmanager.java
      ...
965. if ((active0 & 0x1000000000L) != 0L)
```

[Table of Contents] [Description]

Path 433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1016. if (++jjround == 0x7fffffff)
```

[Table of Contents] [Description]

Path 434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1016. if (++jjround == 0x7fffffff)
```

[Table of Contents] [Description]

Path 435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1026. if ((0xfffffffbffffffffL & 1) != 0L)
```

[Table of Contents] [Description]

Path 436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1026. if ((0xfffffffbffffffffL & 1) != 0L)
```

[Table of Contents] [Description]

Path 437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1030. if (kind > 62)
```

[Table of Contents] [Description]

Path 438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
```

```
curPos) //parsemibtokenmanager.java
      ...
1033. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1033. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1039. if (kind > 58)
```

[Table of Contents] [Description]

Path 441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1043. else if (curChar == 34)
```

[Table of Contents] [Description]

Path 442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1043. else if (curChar == 34)
```

[Table of Contents] [Description]

Path 443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1047. if (kind > 64)
```

[Table of Contents] [Description]

Path 444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1053. if (kind > 58)
```

[Table of Contents] [Description]

Path 445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1057. else if (curChar == 39)
```

[Table of Contents] [Description]

Path 446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1057. else if (curChar == 39)
```

[Table of Contents] [Description]

Path 447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |
```

```
...
1059. if (curChar == 45)
```

[Table of Contents] [Description]

Path 448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1061. if (curChar == 45)
```

[Table of Contents] [Description]

Path 449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1065. if (curChar == 45)
```

[Table of Contents] [Description]

Path 450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1065. if (curChar == 45)
```

[Table of Contents] [Description]

Path 451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1069. if ((0xffffffffffffdbffL & 1) != 0L)
```

[Table of Contents] [Description]

Path 452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1069. if ((0xfffffffffffdbffL & l) != 0L)
```

[Table of Contents] [Description]

Path 453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1073. if ((0x2400L & l) != 0L && kind > 5)
```

[Table of Contents] [Description]

Path 454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1077. if (curChar == 10 && kind > 5)
```

[Table of Contents] [Description]

Path 455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1081. if (curChar == 13)
```

[Table of Contents] [Description]

Path 456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
```

```
1085. if (curChar == 45)
```

[Table of Contents] [Description]

Path 457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1089. if ((0x3fe000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1091. if (kind > 58)
```

[Table of Contents] [Description]

Path 459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1089. if ((0x3fe000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1096. if ((0x3ff000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1098. if (kind > 58)
```

[Table of Contents] [Description]

Path 462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1096. if ((0x3ff000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1103. if (curChar == 39)
```

[Table of Contents] [Description]

Path 464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1103. if (curChar == 39)
```

[Table of Contents] [Description]

Path 465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1107. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1107. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1111. if (curChar == 39)
```

[Table of Contents] [Description]

Path 468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1115. if (curChar != 48)
```

[Table of Contents] [Description]

Path 469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1117. if (kind > 58)
```

[Table of Contents] [Description]

Path 470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1115. if (curChar != 48)
```

[Table of Contents] [Description]

Path 471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1122. if ((0xff000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1124. if (kind > 58)
```

[Table of Contents] [Description]

Path 473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1122. if ((0xff000000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      |1129. if (curChar != 45)
```

[Table of Contents] [Description]

Path 475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1131. if (kind > 64)
```

[Table of Contents] [Description]

Path 476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1129. if (curChar != 45)
```

[Table of Contents] [Description]

Path 477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1136. if ((0x3ff200000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1138. if (kind > 64)
```

[Table of Contents] [Description]

Path 479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
```

```
curPos) //parsemibtokenmanager.java
      ...
1136. if ((0x3ff200000000000L & l) == 0L)
```

[Table of Contents] [Description]

Path 480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1143. if (curChar == 34)
```

[Table of Contents] [Description]

Path 481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1143. if (curChar == 34)
```

[Table of Contents] [Description]

Path 482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1147. if ((0xfffffffbffffffffL & l) != 0L)
```

[Table of Contents] [Description]

Path 483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1147. if ((0xfffffffbffffffffL & l) != 0L)
```

[Table of Contents] [Description]

Path 484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1151. if (curChar == 34 && kind > 62)
```

[Table of Contents] [Description]

Path 485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1155. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1155. if ((0x3ff000000000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1159. if (curChar == 34 && kind > 63)
```

[Table of Contents] [Description]

Path 488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |
```

```
...
1179. if ((0x7fffffe87fffffeL & l) == 0L)
```

[Table of Contents] [Description]

Path 489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1181. if (kind > 64)
```

[Table of Contents] [Description]

Path 490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1179. if ((0x7fffffe87fffffeL & l) == 0L)
```

[Table of Contents] [Description]

Path 491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1189. if ((0x100000001000L & l) != 0L && kind > 58)
```

[Table of Contents] [Description]

Path 492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
...
1193. if ((0x7e0000007eL & l) != 0L)
```

[Table of Contents] [Description]

Path 493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1193. if ((0x7e0000007eL & l) != 0L)
```

[Table of Contents] [Description]

Path 494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1197. if (curChar != 104)
```

[Table of Contents] [Description]

Path 495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1199. if (kind > 58)
```

[Table of Contents] [Description]

Path 496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
      1204. if ((0x400000004000000L & l) != 0L)
```

[Table of Contents] [Description]

Path 497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      |...
```

```
1221. if ((jjbitVec0[i2] & l2) != 0L)
```

[Table of Contents] [Description]

Path 498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1221. if ((jjbitVec0[i2] & l2) != 0L)
```

[Table of Contents] [Description]

Path 499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1225. if ((jjbitVec0[i2] & l2) != 0L)
```

[Table of Contents] [Description]

Path 500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1225. if ((jjbitVec0[i2] & l2) != 0L)
```

[Table of Contents] [Description]

Path 501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1239. if ((i = jjnewStateCnt) == (startsAt = 24 - (jjnewStateCnt = startsAt)))
```

[Table of Contents] [Description]

Path 502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1281. public ParseMibTokenManager(SimpleCharStream
stream) //parsemibtokenmanager.java
      ...
1283. if (input_stream != null)
```

[Table of Contents] [Description]

Path 503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1311. static public void SwitchTo(int lexState) //parsemibtokenmanager.java
      ...
1313. if (lexState >= 1 || lexState < 0)
```

[Table of Contents] [Description]

Path 504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1339. public static Token getNextToken() //parsemibtokenmanager.java
      ...
1370. if (jjmatchedPos + 1 < curPos)
```

[Table of Contents] [Description]

Path 505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1339. public static Token getNextToken() //parsemibtokenmanager.java
      ...
1370. if (jjmatchedPos + 1 < curPos)
```

[Table of Contents] [Description]

Path 506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      ...
113. else if (tokenBegin < 0)
```

[Table of Contents] [Description]

Path 507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |
      ...
      113. else if (tokenBegin < 0)
```

[Table of Contents] [Description]

Path 508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |
      ...
      118. else if (available > tokenBegin)
```

[Table of Contents] [Description]

Path 509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |
      ...
      120. else if ((tokenBegin - available) < 2048)
```

[Table of Contents] [Description]

Path 510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |
      ...
      120. else if ((tokenBegin - available) < 2048)
```

[Table of Contents] [Description]

Path 511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. static protected void FillBuff() throws
java.io.IOException //simplecharstream.java
      |
```

```
...
141. if (tokenBegin == -1)
```

[Table of Contents] [Description]

Path 512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. static public char readChar() throws
java.io.IOException //simplecharstream.java
...
202. if (++bufpos == bufsize)
```

[Table of Contents] [Description]

Path 513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. static public char readChar() throws
java.io.IOException //simplecharstream.java
...
208. if (++bufpos >= maxNextCharInd)
```

[Table of Contents] [Description]

Path 514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. static public char readChar() throws
java.io.IOException //simplecharstream.java
...
208. if (++bufpos >= maxNextCharInd)
```

[Table of Contents] [Description]

Path 515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. static public void backup(int amount) //simplecharstream.java
...
254. if ((bufpos -= amount) < 0)
```

[Table of Contents] [Description]

Path 516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
258. public SimpleCharStream(java.io.Reader dstream, int
startline, //simplecharstream.java
      ...
261. if (inputStream != null)
```

[Table of Contents] [Description]

Path 517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static public String GetImage() //simplecharstream.java
      ...
348. if (bufpos >= tokenBegin)
```

[Table of Contents] [Description]

Path 518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
355. static public char[] GetSuffix(int len) //simplecharstream.java
      ...
359. if ((bufpos + 1) >= len)
```

[Table of Contents] [Description]

Path 519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
355. static public char[] GetSuffix(int len) //simplecharstream.java
      ...
359. if ((bufpos + 1) >= len)
```

[Table of Contents] [Description]

Path 520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
381. static public void adjustBeginLineColumn(int newLine, int
newCol) //simplecharstream.java
      ...
415. if (bufline[j = start % bufsize] != bufline[++start % bufsize])
```

[Table of Contents] [Description]

Path 521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
150. if (encoding.equals("") == true)
```

[Table of Contents] [Description]

Path 522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
174. if (chars[i+1] == '\n') continue;
```

[Table of Contents] [Description]

Path 523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
195. if (carry != -1) out.writeContent(String.valueOf((char)carry));
```

[Table of Contents] [Description]

Path 524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
     ...
195. if (carry != -1) out.writeContent(String.valueOf((char)carry));
```

[Table of Contents] [Description]

Path 525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
    ...
113. if (encoding.equals("") == true)
```

[Table of Contents] [Description]

Path 526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      ...
137. if (chars[i+1] == '\n') continue;
```

[Table of Contents] [Description]

Path 527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      ...
160. if (carry != -1) text += String.valueOf((char)carry);
```

[Table of Contents] [Description]

Path 528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. private void adjust() //runnableconsumerthreadpool.java
      ...
176. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. private void adjust() //runnableconsumerthreadpool.java
      ...
176. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. private void adjust() //runnableconsumerthreadpool.java
      ...
193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. private void adjust() //runnableconsumerthreadpool.java
       ...
193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public void add(Object element) throws FifoQueueException,
InterruptedException //runnableconsumerthreadpool.java
       255. if (m_isClosed)
```

[Table of Contents] [Description]

Path 533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public boolean add(Object element, long timeout) throws FifoQueueException,
InterruptedException //runnableconsumerthreadpool.java
       282. if (m_isClosed)
```

[Table of Contents] [Description]

Path 534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public Object remove() throws FifoQueueException,
InterruptedException //runnableconsumerthreadpool.java
       301. if (m_isClosed && size() == 0)
```

[Table of Contents] [Description]

Path 535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. public Object remove(long timeout) throws FifoQueueException,
InterruptedException //runnableconsumerthreadpool.java
       327. if (m_isClosed && size() == 0)
```

[Table of Contents] [Description]

Path 536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. public void run() //runnableconsumerthreadpool.java
      ...
400. if (obj == null)
```

[Table of Contents] [Description]

Path 537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. public void run() //runnableconsumerthreadpool.java
      ...
428. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. public void run() //runnableconsumerthreadpool.java
      ...
428. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private int livingFiberCount() //runnableconsumerthreadpool.java
      ...
537. if (m_fibers[i].getStatus() != Fiber.STOPPED)
```

[Table of Contents] [Description]

Path 540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. public RunnableConsumerThreadPool(String name, float loMark, float hiMark, int
max) //runnableconsumerthreadpool.java
576. if (loMark > hiMark)
```

[Table of Contents] [Description]

Path 541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. public RunnableConsumerThreadPool(String name, float loMark, float hiMark, int
max) //runnableconsumerthreadpool.java
      |...
      |579. if (max <= 0)
```

[Table of Contents] [Description]

Path 542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
625. public void stop() //runnableconsumerthreadpool.java
      |...
      |628. if (m_fibers[i] != null)
```

[Table of Contents] [Description]

Path 543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
625. public void stop() //runnableconsumerthreadpool.java
      |...
      |628. if (m_fibers[i] != null)
```

[Table of Contents] [Description]

Path 544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. public int getStatus() //runnableconsumerthreadpool.java
      |...
      |648. if (livingFiberCount() == 0)
```

[Table of Contents] [Description]

Path 545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. private static Method findMain(Class c) //vmtaskfiber.java
      |...
      |138. if (args[x].getName().equals(MAIN_PARAMETER_TYPES[x]) == false)
```

[Table of Contents] [Description]

Path 546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. private static Method findMain(Class c) //vmtaskfiber.java
      ...
146. if (isOK)
```

[Table of Contents] [Description]

Path 547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public VMTaskFiber(String taskName, String entryClassName, String[]
entryArguments, URL[] searchPaths) throws ClassNotFoundException,
NoSuchMethodException //vmtaskfiber.java
      ...
189. if (m_entryMethod == null)
```

[Table of Contents] [Description]

Path 548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. public synchronized void stop() //vmtaskfiber.java
244. if (m_fiberStatus != STOPPED)
```

[Table of Contents] [Description]

Path 549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public Object remove(long timeout) throws FifoQueueException,
InterruptedException //fifoqueueimpl.java
      ...
161. if (!m_delegate.isEmpty())
```

[Table of Contents] [Description]

Path 550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
      ...
263. if (System.getProperty((String) e.getKey()) == null)
```

[Table of Contents] [Description]

Path 551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
      ...
263. if (System.getProperty((String) e.getKey()) == null)
```

[Table of Contents] [Description]

Path 552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. public static byte[] decodeBase64(char[] data) //base64.java
      ...
152. if ((data.length % 4) != 0)
```

[Table of Contents] [Description]

Path 553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. public static byte[] decodeBase64(char[] data) //base64.java
      ...
162. if (data[data.length - i] == BASE64_PAD)
```

[Table of Contents] [Description]

Path 554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. public static byte[] decodeBase64(char[] data) //base64.java
      ...
193. if ((int) c >= BASE64_VALUES.length || BASE64_VALUES[(int) c] == -1)
```

[Table of Contents] [Description]

Path 555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
87. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
87. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
95. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
95. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
143. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
143. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
      ...
240. if (bytesRead > 0 && log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
      ...
240. if (bytesRead > 0 && log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
      ...
247. if (bytesRead > 0 && log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public static InputStream post(URL url, Reader dataReader, String username,
String password, int bufSize) throws IOException //httputils.java
      ...
247. if (bytesRead > 0 && log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
      ...
220. if (_mailHost != null)
```

[Table of Contents] [Description]

Path 570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
      ...
```

```
220. if (_mailHost != null)
```

[Table of Contents] [Description]

Path 571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
     ...
223. if (_authenticate)
```

[Table of Contents] [Description]

Path 572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
     ...
223. if (_authenticate)
```

[Table of Contents] [Description]

Path 573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
     ...
236. if (_from != null)
```

[Table of Contents] [Description]

Path 574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
     ...
236. if (_from != null)
```

[Table of Contents] [Description]

Path 575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
```

```
...
236. if (_from != null)
```

[Table of Contents] [Description]

Path 576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
...
286. if (_authenticate)
```

[Table of Contents] [Description]

Path 577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
...
286. if (_authenticate)
```

[Table of Contents] [Description]

Path 578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
...
286. if (_authenticate)
```

[Table of Contents] [Description]

Path 579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1824. public static String cleanText(List list) //installer.java
...
1832. if (l.indexOf(';') != -1) {
```

[Table of Contents] [Description]

Path 580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public SnmpCounter64(String value) //snmpcounter64.java
```

```
143. if (m_value == null)
```

[Table of Contents] [Description]

Path 581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public SnmpCounter64(String value) //snmpcounter64.java
     ...
147. if (m_value.signum() < 0)
```

[Table of Contents] [Description]

Path 582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
216. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpcounter64.java
     ...
219. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public SnmpInt32(String value) //snmpint32.java
     122. if (value == null)
```

[Table of Contents] [Description]

Path 584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpint32.java
     ...
206. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public SnmpIPAddress(byte[] data) //snmpipaddress.java
```

```
...
89. if (data.length < 4)
```

[Table of Contents] [Description]

Path 586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public SnmpIPAddress(byte[] data) //snmpipaddress.java
...
91. else if (data.length > 4)
```

[Table of Contents] [Description]

Path 587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public SnmpIPAddress(SnmpOctetString second) //snmpipaddress.java
...
118. if (getLength() < 4)
```

[Table of Contents] [Description]

Path 588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public SnmpIPAddress(SnmpOctetString second) //snmpipaddress.java
...
120. else if (getLength() > 4)
```

[Table of Contents] [Description]

Path 589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public void setString(byte[] data) //snmpipaddress.java
209. if (data == null || data.length < 4)
```

[Table of Contents] [Description]

Path 590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public void setString(byte[] data) //snmpipaddress.java
```

```
...
211. else if (data.length > 4)
```

[Table of Contents] [Description]

Path 591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public void setString(String data) //snmpipaddress.java
...
243. if (bdata == null || bdata.length < 4)
```

[Table of Contents] [Description]

Path 592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public void setString(String data) //snmpipaddress.java
...
245. else if (bdata.length > 4)
```

[Table of Contents] [Description]

Path 593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpipaddress.java
...
274. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpipaddress.java
...
278. if (data.length < 4)
```

[Table of Contents] [Description]

Path 595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpipaddress.java
      |
      ...
      280. else if (data.length > 4)
```

[Table of Contents] [Description]

Path 596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpnull.java
      |
      ...
      145. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. private static int[] convert(String idstr) //snmpobjectid.java
      |
      ...
      100. if (idArray[0] != '.')
```

[Table of Contents] [Description]

Path 598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. private static int[] convert(String idstr) //snmpobjectid.java
      |
      ...
      108. if (idArray[x++] == '.')
```

[Table of Contents] [Description]

Path 599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. private static int[] convert(String idstr) //snmpobjectid.java
      |
      ...
      132. if (idArray[0] == '.')
```

[Table of Contents] [Description]

Path 600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public int compare(int[] ids) //snmpobjectid.java
      |
      ...
      442. if (aNdx < m_data.length)
```

[Table of Contents] [Description]

Path 601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public int compare(int[] ids) //snmpobjectid.java
      |
      ...
      449. if (bNdx < ids.length)
```

[Table of Contents] [Description]

Path 602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public int compare(int[] ids) //snmpobjectid.java
      |
      ...
      456. if (rc != 0)
```

[Table of Contents] [Description]

Path 603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
479. public int compare(int[] ids, int dist) //snmpobjectid.java
      |
      ...
      503. if (aNdx < m_data.length)
```

[Table of Contents] [Description]

Path 604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
479. public int compare(int[] ids, int dist) //snmpobjectid.java
      |
      ...
      510. if (bNdx < ids.length)
```

[Table of Contents] [Description]

Path 605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
479. public int compare(int[] ids, int dist) //snmpobjectid.java
      ...
517. if (rc != 0)
```

[Table of Contents] [Description]

Path 606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
675. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpobjectid.java
      ...
678. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
177. public int getLength() //snmpoctetstring.java
      ...
179. if (m_data != null)
```

[Table of Contents] [Description]

Path 608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public int encodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnEncodingException //snmpoctetstring.java
      212. if (m_data == null)
```

[Table of Contents] [Description]

Path 609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpoctetstring.java
      ...
241. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public String toString() //snmpoctetstring.java
      |...
      |298. if (x < 16)
```

[Table of Contents] [Description]

Path 611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public String toString() //snmpoctetstring.java
      |...
      |298. if (x < 16)
```

[Table of Contents] [Description]

Path 612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public String toString() //snmpoctetstring.java
      |...
      |302. if (i < m_data.length - 1)
```

[Table of Contents] [Description]

Path 613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public String toString() //snmpoctetstring.java
      |...
      |302. if (i < m_data.length - 1)
```

[Table of Contents] [Description]

Path 614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String toString() //snmpopaque.java
      |...
      |152. if (x < 16)
```

[Table of Contents] [Description]

Path 615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String toString() //snmpopaque.java
      |
      ...
      |
152. if (x < 16)
```

[Table of Contents] [Description]

Path 616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String toString() //snmpopaque.java
      |
      ...
      |
156. if (i < data.length)
```

[Table of Contents] [Description]

Path 617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public String toString() //snmpopaque.java
      |
      ...
      |
156. if (i < data.length)
```

[Table of Contents] [Description]

Path 618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
670. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmppdupacket.java
      |
      ...
      |
681. if (cmd < 0)
```

[Table of Contents] [Description]

Path 619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
670. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmppdupacket.java
      |
      ...
      |
711. if (asnType != SnmpVarBind.ASNTYPE)
```

[Table of Contents] [Description]

Path 620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmppdutrap.java
      ...
518. if (cmd < 0)
```

[Table of Contents] [Description]

Path 621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmppdutrap.java
      ...
520. if (TRAP != cmd)
```

[Table of Contents] [Description]

Path 622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmppdutrap.java
      ...
560. if (asnType != SnmpVarBind.ASNTYPE)
```

[Table of Contents] [Description]

Path 623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. SnmpPortal(SnmpPacketHandler handler, AsnEncoder encoder, int port) throws
SocketException //snmpportal.java
      162. if (handler == null || encoder == null)
```

[Table of Contents] [Description]

Path 624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public void run() //snmpportal.java
      ...
284. if (!usedBuffers.isEmpty())
```

[Table of Contents] [Description]

Path 625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public void run() //snmpportal.java
      |
      ...
      |
288. if (buf == null || buf.length != bufSz)
```

[Table of Contents] [Description]

Path 626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public void run() //snmpportal.java
      |
      ...
      |
348. if (m_isClosing)
```

[Table of Contents] [Description]

Path 627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public void run() //snmpportal.java
      |
      ...
      |
473. if (usedBuffers.size() < 20)
```

[Table of Contents] [Description]

Path 628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public void run() //snmpportal.java
      |
      ...
      |
473. if (usedBuffers.size() < 20)
```

[Table of Contents] [Description]

Path 629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
506. void handlePkt(DatagramPacket pkt) throws SnmpPduEncodingException,
AsnDecodingException //snmpportal.java
      |
      ...
      |
529. if (asnType != (ASN1.SEQUENCE | ASN1.CONSTRUCTOR))
```

[Table of Contents] [Description]

Path 630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
673. void setPacketHandler(SnmpPacketHandler hdl) //snmpportal.java
     |674. if (hdl == null)
```

[Table of Contents] [Description]

Path 631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
696. void setAsnEncoder(AsnEncoder encoder) //snmpportal.java
     |697. if (encoder == null)
```

[Table of Contents] [Description]

Path 632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |151. if (m_expired)
```

[Table of Contents] [Description]

Path 633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     |162. if (((SnmpPduPacket) m_pdu).getCommand() != SnmpPduPacket.V2TRAP)
```

[Table of Contents] [Description]

Path 634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     |162. if (((SnmpPduPacket) m_pdu).getCommand() != SnmpPduPacket.V2TRAP)
```

[Table of Contents] [Description]

Path 635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmpsession.java
     |...
     205. if (req != null)
```

[Table of Contents] [Description]

Path 636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmpsession.java
     |...
     211. if (req != null && req.m_pdu instanceof SnmpPduPacket)
```

[Table of Contents] [Description]

Path 637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmpsession.java
     |...
     249. if (m_defHandler != null)
```

[Table of Contents] [Description]

Path 638:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmpsession.java
     |...
     249. if (m_defHandler != null)
```

[Table of Contents] [Description]

Path 639:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
282. public void run() //snmpsession.java
     |
```

```
...
288. if (req.m_expired)
```

[Table of Contents] [Description]

Path 640:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
282. public void run() //snmpsession.java
...
288. if (req.m_expired)
```

[Table of Contents] [Description]

Path 641:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
282. public void run() //snmpsession.java
...
297. if (!m_stopRun && !m_threadException)
```

[Table of Contents] [Description]

Path 642:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
282. public void run() //snmpsession.java
...
297. if (!m_stopRun && !m_threadException)
```

[Table of Contents] [Description]

Path 643:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
369. private ByteArrayInfo encode(SnmpPduPacket pdu) throws
SnmpPduEncodingException, AsnEncodingException //snmpsession.java
...
397. if (wrComm == null)
```

[Table of Contents] [Description]

Path 644:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
      ...
      596. if (req.m_pdu instanceof SnmpPduPacket)
```

[Table of Contents] [Description]

Path 645:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
      ...
      599. if (req.m_pdu instanceof SnmpPduTrap)
```

[Table of Contents] [Description]

Path 646:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
      ...
      621. if (pdu.getPeer() == null)
```

[Table of Contents] [Description]

Path 647:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
      ...
      621. if (pdu.getPeer() == null)
```

[Table of Contents] [Description]

Path 648:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
      ...
      621. if (pdu.getPeer() == null)
```

[Table of Contents] [Description]

Path 649:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. public int getOutstandingCount() //snmpsession.java
       ...
775. if (m_stopRun) // session has been closed!
```

[Table of Contents] [Description]

Path 650:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. public int getOutstandingCount() //snmpsession.java
       ...
788. if (req.m_expired)
```

[Table of Contents] [Description]

Path 651:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. public int getOutstandingCount() //snmpsession.java
       ...
788. if (req.m_expired)
```

[Table of Contents] [Description]

Path 652:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
808. public void cancel(int requestId) //snmpsession.java
       ...
813. if (m_stopRun) // session has been closed!
```

[Table of Contents] [Description]

Path 653:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
851. public int send(SnmpPduPacket pdu, SnmpHandler handler) //snmpsession.java
       852. if (handler == null)
```

[Table of Contents] [Description]

Path 654:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
851. public int send(SnmpPduPacket pdu, SnmpHandler handler) //snmpsession.java
     ...
859. if (m_stopRun) // session has been closed!
```

[Table of Contents] [Description]

Path 655:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
851. public int send(SnmpPduPacket pdu, SnmpHandler handler) //snmpsession.java
     ...
864. if (pdu.getCommand() != SnmpPduPacket.V2TRAP || pdu.getPeer() == null) //
traps
```

[Table of Contents] [Description]

Path 656:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
851. public int send(SnmpPduPacket pdu, SnmpHandler handler) //snmpsession.java
     ...
864. if (pdu.getCommand() != SnmpPduPacket.V2TRAP || pdu.getPeer() == null) //
traps
```

[Table of Contents] [Description]

Path 657:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
894. public int send(SnmpPduPacket pdu) //snmpsession.java
     895. if (m_defHandler == null)
```

[Table of Contents] [Description]

Path 658:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
916. public int send(SnmpPduTrap pdu, SnmpHandler handler) //snmpsession.java
     917. if (handler == null)
```

[Table of Contents] [Description]

Path 659:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
916. public int send(SnmpPduTrap pdu, SnmpHandler handler) //snmpsession.java
       ...
924. if (m_stopRun) // session has been closed!
```

[Table of Contents] [Description]

Path 660:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
946. public int send(SnmpPduTrap pdu) //snmpsession.java
947. if (m_defHandler == null)
```

[Table of Contents] [Description]

Path 661:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
973. public void close() //snmpsession.java
       ...
978. if (m_stopRun)
```

[Table of Contents] [Description]

Path 662:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1001. public void raise() throws Throwable //snmpsession.java
       ...
1003. if (m_threadException)
```

[Table of Contents] [Description]

Path 663:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void run() //snmptimer.java
       ...
126. if (m_exit)
```

[Table of Contents] [Description]

Path 664:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void run() //snmptimer.java
      |
      ...
      |
173. if (elem.m_when.getTime() < minTime)
```

[Table of Contents] [Description]

Path 665:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void run() //snmptimer.java
      |
      ...
      |
191. if (minTime > 0)
```

[Table of Contents] [Description]

Path 666:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void run() //snmptimer.java
      |
      ...
      |
191. if (minTime > 0)
```

[Table of Contents] [Description]

Path 667:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
264. void cancel() //snmptimer.java
      |
      ...
      |
276. if (m_thread.equals(Thread.currentThread()) == false)
```

[Table of Contents] [Description]

Path 668:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
264. void cancel() //snmptimer.java
      |
      ...
      |
276. if (m_thread.equals(Thread.currentThread()) == false)
```

[Table of Contents] [Description]

Path 669:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmptrapsession.java
      193. if (version.getValue() != SnmpSMI.SNMPV2 && pduType !=
      SnmpPduPacket.V2TRAP)
```

[Table of Contents] [Description]

Path 670:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public void close() //snmptrapsession.java
      365. if (m_portal.isClosed())
```

[Table of Contents] [Description]

Path 671:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
378. public void raise() throws Throwable //snmptrapsession.java
      379. if (m_threadException)
```

[Table of Contents] [Description]

Path 672:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. public void send(SnmpPeer peer, SnmpPduTrap trap) throws
SnmpPduEncodingException, AsnEncodingException,
java.io.IOException //snmptrapsession.java
      412. if (m_portal.isClosed())
```

[Table of Contents] [Description]

Path 673:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
503. public void send(SnmpPeer peer, SnmpPduPacket pdu) throws
SnmpPduEncodingException, AsnEncodingException,
java.io.IOException //snmptrapsession.java
      504. if (m_portal.isClosed())
```

[Table of Contents] [Description]

Path 674:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public SnmpUInt32(String value) //snmpuint32.java
     |147. if (value == null)
```

[Table of Contents] [Description]

Path 675:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public SnmpUInt32(String value) //snmpuint32.java
     |...
     |151. if (m_value < 0)
```

[Table of Contents] [Description]

Path 676:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpuint32.java
     |...
     |244. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 677:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. static boolean registerSyntax(SnmpSyntax obj) //snmputil.java
     |...
     |99. if (addIt == true)
```

[Table of Contents] [Description]

Path 678:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
     |260. if ((offset + length) > data.length)
```

[Table of Contents] [Description]

Path 679:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
      ...
      273. if ((cnt++ % 16) == 0 && cnt != 1)
```

[Table of Contents] [Description]

Path 680:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
      ...
      273. if ((cnt++ % 16) == 0 && cnt != 1)
```

[Table of Contents] [Description]

Path 681:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
133. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpv2error.java
      ...
      136. if (((Byte) rVals[1]).byteValue() != typeId())
```

[Table of Contents] [Description]

Path 682:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
133. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpv2error.java
      ...
      139. if (((Integer) rVals[2]).intValue() != 0)
```

[Table of Contents] [Description]

Path 683:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpvarbind.java
      ...
      301. if ((buf.length - offset) < asnLength)
```

[Table of Contents] [Description]

Path 684:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpvarbind.java
      ...
307. if (asnType != typeId())
```

[Table of Contents] [Description]

Path 685:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public int decodeASN(byte[] buf, int offset, AsnEncoder encoder) throws
AsnDecodingException //snmpvarbind.java
      ...
319. if (m_value == null)
```

[Table of Contents] [Description]

Path 686:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. protected static void copy(byte[] src, int srcOff, byte[] dest, int destOff,
int bytesToCopy) throws ArrayIndexOutOfBoundsException //berencoder.java
      147. if ((dest.length - destOff) < bytesToCopy || (src.length - srcOff) <
      bytesToCopy)
```

[Table of Contents] [Description]

Path 687:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. protected static void copy(int[] src, int srcOff, int[] dest, int destOff, int
intsToCopy) throws ArrayIndexOutOfBoundsException //berencoder.java
      177. if ((dest.length - destOff) < intsToCopy || (src.length - srcOff) <
      intsToCopy)
```

[Table of Contents] [Description]

Path 688:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. public int buildLength(byte[] buf, int startOffset, int asnLength) throws
AsnEncodingException //berencoder.java
      ...
264. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 689:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. public int buildLength(byte[] buf, int startOffset, int asnLength) throws
AsnEncodingException //berencoder.java
      ...
272. if ((buf.length - startOffset) < 2)
```

[Table of Contents] [Description]

Path 690:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. public int buildLength(byte[] buf, int startOffset, int asnLength) throws
AsnEncodingException //berencoder.java
      ...
282. if ((buf.length - startOffset) < 3)
```

[Table of Contents] [Description]

Path 691:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public Object[] parseLength(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
315. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 692:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public Object[] parseLength(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
341. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 693:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public Object[] parseLength(byte[] buf, int startOffset) throws
```

```
AsnDecodingException //berencoder.java
    ...
346. if ((buf.length - startOffset) < 2)
```

[Table of Contents] [Description]

Path 694:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
388. public int buildHeader(byte[] buf, int startOffset, byte asnType, int
asnLength) throws AsnEncodingException //berencoder.java
    389. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 695:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public Object[] parseHeader(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
    418. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 696:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public Object[] parseHeader(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
    ...
425. if (isExtensionId(asnType))
```

[Table of Contents] [Description]

Path 697:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
473. public int buildInteger32(byte[] buf, int startOffset, byte asnType, int
asnInt32) throws AsnEncodingException //berencoder.java
    ...
495. if ((buf.length - startOffset) < intSz)
```

[Table of Contents] [Description]

Path 698:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
533. public Object[] parseInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      546. if ((buf.length - startOffset) < asnLength)
```

[Table of Contents] [Description]

Path 699:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
533. public Object[] parseInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      553. if (asnLength > 4)
```

[Table of Contents] [Description]

Path 700:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
533. public Object[] parseInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      560. if ((buf[startOffset] & HIGH_BIT) == HIGH_BIT)
```

[Table of Contents] [Description]

Path 701:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
604. public int buildUInteger32(byte[] buf, int startOffset, byte asnType, long
asnUInt32) throws AsnEncodingException //berencoder.java
      |...
      656. if ((buf.length - startOffset) < intSz)
```

[Table of Contents] [Description]

Path 702:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
702. public Object[] parseUInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      715. if ((buf.length - startOffset) < asnLength)
```

[Table of Contents] [Description]

Path 703:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
702. public Object[] parseUInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      |722. if (asnLength > 5)
```

[Table of Contents] [Description]

Path 704:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
702. public Object[] parseUInteger32(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      |729. if ((buf[startOffset] & HIGH_BIT) == HIGH_BIT)
```

[Table of Contents] [Description]

Path 705:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
779. public int buildUInteger64(byte[] buf, int startOffset, byte asnType,
BigInteger asnUInt64) throws AsnEncodingException //berencoder.java
      |...
      |795. if ((buf.length - startOffset) < bytes.length)
```

[Table of Contents] [Description]

Path 706:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
826. public Object[] parseUInteger64(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
      |839. if ((buf.length - startOffset) < asnLength)
```

[Table of Contents] [Description]

Path 707:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
826. public Object[] parseUInteger64(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      |...
```

```
846. if (asnLength > 9)
```

[Table of Contents] [Description]

Path 708:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
910. public Object[] parseNull(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
916. if (((Integer) hdrVals[2]).intValue() != 0)
```

[Table of Contents] [Description]

Path 709:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
949. public int buildString(byte[] buf, int startOffset, byte asnType, byte[]
opaque) throws AsnEncodingException //berencoder.java
      ...
964. if ((buf.length - startOffset) < opaque.length)
```

[Table of Contents] [Description]

Path 710:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1002. public Object[] parseString(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
1015. if ((buf.length - startOffset) < asnLength)
```

[Table of Contents] [Description]

Path 711:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      1067. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 712:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1086. if (toEncode[0] < 0 || toEncode[0] > 2)
```

[Table of Contents] [Description]

Path 713:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1089. if (toEncode[1] < 0 || toEncode[1] > 40)
```

[Table of Contents] [Description]

Path 714:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1111. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 715:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1135. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 716:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1146. if (mask == 0x01e00000)
```

[Table of Contents] [Description]

Path 717:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      ...
1153. if ((buf.length - startOffset) < 1)
```

[Table of Contents] [Description]

Path 718:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1203. public Object[] parseObjectId(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
1213. if ((buf.length - startOffset) < asnLength)
```

[Table of Contents] [Description]

Path 719:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1203. public Object[] parseObjectId(byte[] buf, int startOffset) throws
AsnDecodingException //berencoder.java
      ...
1263. if ((b & HIGH_BIT) == 0)
```

[Table of Contents] [Description]

Path 720:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 721:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
```

```
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 722:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
162. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 723:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
162. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 724:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
166. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 725:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
```

```
...
166. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 726:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
170. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 727:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
170. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 728:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
175. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 729:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
```

```
175. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 730:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
180. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 731:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
180. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 732:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
197. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 733:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
...
197. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 734:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 735:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 736:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |207. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 737:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |207. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 738:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |215. if (created == null)
```

[Table of Contents] [Description]

Path 739:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |220. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 740:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      |...
      |220. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 741:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      |...
      |237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 742:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 743:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
240. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 744:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
240. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 745:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
249. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 746:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
```

```
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
249. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 747:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
262. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 748:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
262. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 749:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 750:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 751:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      |
      |...
      |275. if (m_nodes.size() <= 0)
```

[Table of Contents] [Description]

Path 752:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      |
      |...
      |301. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 753:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      |
      |...
      |301. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 754:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
374. private static synchronized int getDays(boolean isLeap, int
month) //availabilitydata.java
      |
      |...
      |392. if (isLeap)
```

[Table of Contents] [Description]

Path 755:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    ...
434. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 756:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    ...
434. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 757:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    ...
438. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 758:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    ...
438. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 759:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    ...
442. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 760:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
     ...
442. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 761:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
     ...
446. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 762:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
     ...
446. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 763:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
     ...
450. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 764:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
     ...
```

```
450. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 765:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
459. public void closeConnection() //availabilitydata.java
        ...
466. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 766:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
459. public void closeConnection() //availabilitydata.java
        ...
466. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 767:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public TreeMap getPercentNode() //availabilitydata.java
        ...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 768:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public TreeMap getPercentNode() //availabilitydata.java
        ...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 769:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public TreeMap getPercentNode() //availabilitydata.java
```

```
...
501. if (tmp == null)
```

[Table of Contents] [Description]

Path 770:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public TreeMap getPercentNode() //availabilitydata.java
...
508. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 771:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. public TreeMap getPercentNode() //availabilitydata.java
...
508. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 772:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
528. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 773:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
528. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 774:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
547. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 775:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
547. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 776:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
555. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 777:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
555. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 778:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
```

```
...
613. if (svcRS != null)
```

[Table of Contents] [Description]

Path 779:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
613. if (svcRS != null)
```

[Table of Contents] [Description]

Path 780:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
616. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 781:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
616. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 782:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
623. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 783:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
623. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 784:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
629. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 785:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
629. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 786:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
635. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 787:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
      |635. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 788:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
      |642. if (ipRS != null)
```

[Table of Contents] [Description]

Path 789:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
      |642. if (ipRS != null)
```

[Table of Contents] [Description]

Path 790:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
      |644. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 791:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
      |...
```

```
644. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 792:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
647. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 793:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
647. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 794:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
650. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 795:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
...
650. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 796:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
653. if (m_availConn != null)
```

[Table of Contents] [Description]

Path 797:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
653. if (m_availConn != null)
```

[Table of Contents] [Description]

Path 798:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
656. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 799:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
656. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 800:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
708. if (regained != null)
```

[Table of Contents] [Description]

Path 801:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
712. if (regainedtime <= m_12MonthsBack || losttime >= m_endTime)
```

[Table of Contents] [Description]

Path 802:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
715. if (losttime >= m_endTime)
```

[Table of Contents] [Description]

Path 803:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
722. if (rs != null)
```

[Table of Contents] [Description]

Path 804:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
      ...
```

```
722. if (rs != null)
```

[Table of Contents] [Description]

Path 805:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
...
726. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 806:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
667. private void getOutagesNodeIpSvc(int nodeid, String nodeName, String ipaddr,
int serviceid, String serviceName, OutageSvcTimesList outageSvcTimesList,
PreparedStatement outagesGetStmt) throws SQLException //availabilitydata.java
...
726. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 807:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
...
737. if (m_nodes == null)
```

[Table of Contents] [Description]

Path 808:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
...
745. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 809:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     |...
     |745. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 810:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     |...
     |745. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 811:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     |...
     |769. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 812:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     |...
     |769. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 813:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     |...
     |769. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 814:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     ...
780. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 815:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     ...
780. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 816:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
735. public void addNode(String nodeName, int nodeid, String ipaddr, String
serviceid, long losttime, long regainedtime) //availabilitydata.java
     ...
780. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 817:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public AvailabilityReport(String author, String startMonth, String startDate,
String startYear) //availabilityreport.java
     ...
110. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 818:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public AvailabilityReport(String author, String startMonth, String startDate,
String startYear) //availabilityreport.java
```

```
...
110. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 819:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
...
171. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 820:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
...
171. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 821:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
...
227. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 822:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
...
227. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 823:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     |...
     231. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 824:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     |...
     231. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 825:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     |...
     234. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 826:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     |...
     234. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 827:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
     |...
     237. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 828:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void marshalReport() throws ValidationException, MarshalException,
IOException, Exception //availabilityreport.java
      ...
237. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 829:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
      ...
248. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 830:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
      ...
248. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 831:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
      ...
252. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 832:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
```

```
...
252. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 833:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
...
263. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 834:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
...
263. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 835:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
...
267. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 836:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
...
267. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 837:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public static void main(String args[]) //availabilityreport.java
       |...
       279. if (categoryName == null || categoryName.equals(""))
```

[Table of Contents] [Description]

Path 838:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public static void main(String args[]) //availabilityreport.java
       |...
       282. if (format == null || format.equals(""))
```

[Table of Contents] [Description]

Path 839:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. public static void main(String args[]) //availabilityreport.java
       |...
       285. if (monthFormat == null || format.equals(""))
```

[Table of Contents] [Description]

Path 840:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
       |...
       312. if (format.equals("SVG"))
```

[Table of Contents] [Description]

Path 841:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
       |...
       314. else if (format.equals("PDF"))
```

[Table of Contents] [Description]

Path 842:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
     ...
326. if (format.equals("SVG"))
```

[Table of Contents] [Description]

Path 843:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
     ...
321. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 844:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
     ...
321. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 845:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
     ...
326. if (format.equals("SVG"))
```

[Table of Contents] [Description]

Path 846:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
    ...
328. else if (format.equals("PDF"))
```

[Table of Contents] [Description]

Path 847:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
    ...
330. else if (format.equals("HTML"))
```

[Table of Contents] [Description]

Path 848:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
    ...
338. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 849:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
    ...
338. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 850:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
    ...
```

```
341. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 851:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
...
341. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 852:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private int numRowsWithValue(Section section, String title, String
data) //availabilityreporttest.java
...
168. if (value[k].getType().equals("title") &&
```

[Table of Contents] [Description]

Path 853:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private int numRowsWithValue(Section section, String title, String
data) //availabilityreporttest.java
...
171. if (value[k].getType().equals("data") &&
```

[Table of Contents] [Description]

Path 854:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private int numRowsWithValue(Section section, String title, String
data) //availabilityreporttest.java
...
174. if (datamatch && titlematch)
```

[Table of Contents] [Description]

Path 855:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 856:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 857:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 858:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 859:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
212. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 860:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
214. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 861:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 862:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 863:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 864:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 865:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
228. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 866:

Query Name - **Single_Line_If_Segment**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
230. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 867:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
```

```
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 868:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 869:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
243. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 870:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
243. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 871:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
```

```
sectionIndex) //availcalculations.java
    ...
247. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 872:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
249. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 873:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 874:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 875:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
```

```
...
266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 876:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 877:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
270. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 878:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
272. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 879:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
```

```
275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 880:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 881:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 882:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 883:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
286. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 884:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
288. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 885:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 886:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 887:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
299. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 888:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |299. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 889:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |303. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 890:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |305. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 891:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |308. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 892:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
308. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 893:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
316. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 894:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
316. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 895:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
320. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 896:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
322. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 897:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
325. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 898:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
325. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 899:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 900:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 901:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
337. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 902:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
339. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 903:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 904:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
```

```
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 905:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
365. if (svcOutages == null)
```

[Table of Contents] [Description]

Path 906:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 907:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 908:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
```

```
sectionIndex) //availcalculations.java
    ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 909:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 910:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
387. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 911:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
    ...
389. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 912:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
```

```
...
392. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 913:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
392. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 914:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 915:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 916:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
```

```
404. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 917:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
406. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 918:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
409. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 919:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
409. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 920:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
...
417. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 921:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
417. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 922:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
421. if (label == null || label.length() == 0)
```

[Table of Contents] [Description]

Path 923:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
423. if (descr == null || descr.length() == 0)
```

[Table of Contents] [Description]

Path 924:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
     ...
426. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 925:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |426. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 926:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |437. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 927:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public AvailCalculations(List nodes, long endTime, long lastMonthEndTime, List
monitoredServices, Report report, TreeMap offenders, double warning, double normal,
String comments, String name, String format, String monthFormat, int catIndex, int
sectionIndex) //availcalculations.java
      |...
      |437. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 928:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
455. private void lastMonTopNServiceOutages(CatSections catSections, String label,
String descr) //availcalculations.java
      |...
      |485. if (tmpList == null)
```

[Table of Contents] [Description]

Path 929:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. private void lastMoTopNOffenders(TreeMap offenders, CatSections catSections,
String label, String descr) //availcalculations.java
      ...
690. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 930:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. private void lastMoTopNOffenders(TreeMap offenders, CatSections catSections,
String label, String descr) //availcalculations.java
      ...
690. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 931:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
752. private void lastNDaysCalDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
      ...
755. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 932:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
752. private void lastNDaysCalDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
      ...
755. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 933:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
752. private void lastNDaysCalDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
      ...
```

```
764. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 934:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
752. private void lastNDaysCalDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
...
764. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 935:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
752. private void lastNDaysCalDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
...
778. if (serviceCount > 0)
```

[Table of Contents] [Description]

Path 936:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
832. private void lastNDaysDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
...
835. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 937:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
832. private void lastNDaysDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
...
835. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 938:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
832. private void lastNDaysDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
     |...
     |844. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 939:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
832. private void lastNDaysDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
     |...
     |844. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 940:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
832. private void lastNDaysDailyAvailability(int days, long endTime, CatSections
catSections, String label, String descr, String
sectionName) //availcalculations.java
     |...
     |858. if (serviceCount > 0)
```

[Table of Contents] [Description]

Path 941:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
923. private void lastNDaysTotalAvailability(int days, long endTime, CatSections
catSections, String label, String descr) //availcalculations.java
     |...
     |948. if (serviceCount > 0)
```

[Table of Contents] [Description]

Path 942:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
997. private void lastNMonthsAvailability(int nMonths, long endTime, CatSections
```

```
catSections, String label, String descr) //availcalculations.java
    ...
1034. if (serviceCount > 0)
```

[Table of Contents] [Description]

Path 943:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1104. private static synchronized int getDays(boolean isLeap, int
month) //availcalculations.java
    ...
1122. if (isLeap)
```

[Table of Contents] [Description]

Path 944:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1157. private void lastNDaysDailyServiceAvailability(int days, long endTime,
CatSections catSections, String label, String descr) //availcalculations.java
    ...
1178. if (m_services != null)
```

[Table of Contents] [Description]

Path 945:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1157. private void lastNDaysDailyServiceAvailability(int days, long endTime,
CatSections catSections, String label, String descr) //availcalculations.java
    ...
1254. if (den > 0)
```

[Table of Contents] [Description]

Path 946:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
71. public boolean equals(Object o) //interface.java
    ...
74. if (o == this)
```

[Table of Contents] [Description]

Path 947:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
71. public boolean equals(Object o) //interface.java
      ...
76. else if (o instanceof Service)
```

[Table of Contents] [Description]

Path 948:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
71. public boolean equals(Object o) //interface.java
      ...
78. else if (o instanceof String)
```

[Table of Contents] [Description]

Path 949:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public Interface(String name, ArrayList services) //interface.java
      ...
102. if (services == null)
```

[Table of Contents] [Description]

Path 950:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public Interface(String name, String service, long losttime) //interface.java
      ...
147. if (losttime > 0)
```

[Table of Contents] [Description]

Path 951:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public Interface(String name, String service, long losttime) //interface.java
      ...
147. if (losttime > 0)
```

[Table of Contents] [Description]

Path 952:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public Interface(String name, String service, long losttime, long
regainedtime) //interface.java
      |...
      |175. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 953:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public Interface(String name, String service, long losttime, long
regainedtime) //interface.java
      |...
      |175. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 954:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public Interface(String name, String service, long losttime, long
regainedtime) //interface.java
      |...
      |175. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 955:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
201. public Interface(ArrayList services) //interface.java
      |202. if (services == null)
```

[Table of Contents] [Description]

Path 956:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
239. public void addService(Service service) //interface.java
      |240. if (service != null)
```

[Table of Contents] [Description]

Path 957:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
239. public void addService(Service service) //interface.java
     240. if (service != null)
```

[Table of Contents] [Description]

Path 958:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. public void addService(String service, long losttime) //interface.java
     ...
     275. if (losttime > 0)
```

[Table of Contents] [Description]

Path 959:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. public void addService(String service, long losttime) //interface.java
     ...
     275. if (losttime > 0)
```

[Table of Contents] [Description]

Path 960:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. public void addService(String service, long losttime) //interface.java
     ...
     283. if (losttime > 0)
```

[Table of Contents] [Description]

Path 961:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. public void addService(String service, long losttime) //interface.java
     ...
     283. if (losttime > 0)
```

[Table of Contents] [Description]

Path 962:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |301. if (service == null)
```

[Table of Contents] [Description]

Path 963:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |309. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 964:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |309. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 965:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |309. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 966:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |320. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 967:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |320. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 968:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
300. public void addService(String service, long losttime, long
regainedtime) //interface.java
     |...
     |320. if (regainedtime > 0)
```

[Table of Contents] [Description]

Path 969:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. public long getDownTime(long currentTime, long rollingWindow) //interface.java
     |...
     |344. if (down > 0)
```

[Table of Contents] [Description]

Path 970:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
354. public int getServiceCount() //interface.java
     |355. if (m_services != null && m_services.size() > 0)
```

[Table of Contents] [Description]

Path 971:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. public int getServiceAffectCount() //interface.java
     |...
     |369. if (service.getOutages().size() > 0)
```

[Table of Contents] [Description]

Path 972:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
376. public boolean equals(Object obj) //interface.java
       ...
378. if (obj instanceof String)
```

[Table of Contents] [Description]

Path 973:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
376. public boolean equals(Object obj) //interface.java
       ...
380. else if (obj instanceof Interface)
```

[Table of Contents] [Description]

Path 974:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public boolean equals(Object o) //node.java
      ...
73. if (o == this)
```

[Table of Contents] [Description]

Path 975:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public boolean equals(Object o) //node.java
      ...
75. else if (o instanceof Interface)
```

[Table of Contents] [Description]

Path 976:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public boolean equals(Object o) //node.java
      ...
77. else if (o instanceof String)
```

[Table of Contents] [Description]

Path 977:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
248. public int getInterfaceCount() //node.java
     |249. if (m_interfaces != null)
```

[Table of Contents] [Description]

Path 978:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
314. public void addInterface(String intfname) //node.java
     |315. if (intfname == null)
```

[Table of Contents] [Description]

Path 979:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void addInterface(String intfname, String service) //node.java
     |336. if (intfname == null)
```

[Table of Contents] [Description]

Path 980:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void addInterface(String intfname, String service) //node.java
     |...
     |341. if (service != null)
```

[Table of Contents] [Description]

Path 981:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void addInterface(String intfname, String service) //node.java
     |...
     |341. if (service != null)
```

[Table of Contents] [Description]

Path 982:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void addInterface(String intfname, String service) //node.java
     |...
     |346. if (service != null)
```

[Table of Contents] [Description]

Path 983:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void addInterface(String intfname, String service) //node.java
     |...
     |346. if (service != null)
```

[Table of Contents] [Description]

Path 984:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
388. public Interface getInterface(String intfname) //node.java
     |389. if (intfname == null)
```

[Table of Contents] [Description]

Path 985:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
415. public void addInterface(String intfname, String service, long lost, long
regained) //node.java
     |416. if (intfname == null)
```

[Table of Contents] [Description]

Path 986:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
443. public double getPercentAvail(long endTime, long rollingWindow) //node.java
     |...
     |454. if (down > 0)
```

[Table of Contents] [Description]

Path 987:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
478. public long getOutage(long endTime, long rollingWindow) //node.java
```

```
...
489. if (down > 0)
```

[Table of Contents] [Description]

Path 988:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. public void addSvcTime(long losttime, long
regainedtime) //outagesvctimeslist.java
     109. if (regainedtime < losttime)
```

[Table of Contents] [Description]

Path 989:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public long getDownTime(long curTime, long
rollingWindow) //outagesvctimeslist.java
     ...
     147. if (outtime > 0)
```

[Table of Contents] [Description]

Path 990:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
     165. if (nodeName == null)
```

[Table of Contents] [Description]

Path 991:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
     ...
     181. if (svcTime.getRegainedTime() <= startTime)
```

[Table of Contents] [Description]

Path 992:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
      ...
184. if (svcTime.getLostTime() > curTime)
```

[Table of Contents] [Description]

Path 993:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
      ...
194. if (outtime > 0)
```

[Table of Contents] [Description]

Path 994:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
      ...
196. if (outageSince != null)
```

[Table of Contents] [Description]

Path 995:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public List getServiceOutages(String nodeName, long curTime, long
rollingWindow) //outagesvctimeslist.java
      ...
196. if (outageSince != null)
```

[Table of Contents] [Description]

Path 996:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      ...
101. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 997:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
      101. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 998:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
      116. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 999:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
      116. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1000:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
      121. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1001:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      |...
```

```
121. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1002:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
...
137. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1003:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
...
137. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1004:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
...
155. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1005:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
...
155. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1006:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      |...
      189. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1007:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      |...
      189. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1008:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      |...
      194. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1009:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      |...
      194. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1010:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void initDriver() throws Exception //pdfwriter.java
       ...
220. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1011:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void initDriver() throws Exception //pdfwriter.java
       ...
220. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1012:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. private XMLReader createParser() throws Exception //pdfwriter.java
       ...
241. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1013:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. private XMLReader createParser() throws Exception //pdfwriter.java
       ...
241. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 1014:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public Service(String name, OutageSvcTimesList outages) //service.java
       ...
109. if (outages != null)
```

[Table of Contents] [Description]

Path 1015:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public Service(OutageSvcTimesList outages) //service.java
     |122. if (outages != null)
```

[Table of Contents] [Description]

Path 1016:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void addOutage(long lost, long regained) //service.java
     |183. if (m_outageList == null)
```

[Table of Contents] [Description]

Path 1017:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void addOutage(long lost) //service.java
     |192. if (m_outageList == null)
```

[Table of Contents] [Description]

Path 1018:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
200. public void addOutage(Outage outage) //service.java
     |201. if (m_outageList == null)
```

[Table of Contents] [Description]

Path 1019:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
209. public long getDownTime(long currentTime, long rollingWindow) //service.java
     |210. if (m_outageList != null)
```

[Table of Contents] [Description]

Path 1020:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public boolean equals(Object obj) //service.java
     |...
     |244. if (obj instanceof String)
```

[Table of Contents] [Description]

Path 1021:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public boolean equals(Object obj) //service.java
       ...
246. else if (obj instanceof Service)
```

[Table of Contents] [Description]

Path 1022:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
654. public static final String getHome() //configfileconstants.java
       ...
662. if (home.endsWith("/") || home.endsWith(File.separator))
```

[Table of Contents] [Description]

Path 1023:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
131. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1024:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
131. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1025:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public void run() //executor.java
       ...
232. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1026:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public void run() //executor.java
       ...
232. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 1027:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
382. public void run() //executor.java
       ...
420. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1028:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
382. public void run() //executor.java
       ...
420. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1029:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
382. public void run() //executor.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1030:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
382. public void run() //executor.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1031:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public synchronized int getStatus() //executor.java
       ...
605. if (m_reaper.isAlive())
```

[Table of Contents] [Description]

Path 1032:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public synchronized int getStatus() //executor.java
       ...
605. if (m_reaper.isAlive())
```

[Table of Contents] [Description]

Path 1033:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
       ...
353. if (m_logCat.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1034:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
       ...
353. if (m_logCat.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1035:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
       ...
367. if (ret)
```

[Table of Contents] [Description]

Path 1036:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
       ...
379. if (m_logCat.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1037:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
       ...
379. if (m_logCat.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1038:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. protected AbstractTcpPlugin(String protocol, int defaultPort, int
defaultTimeout, int defaultRetry) //abstracttcpplugin.java
       85. if (protocol == null)
```

[Table of Contents] [Description]

Path 1039:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. final protected boolean checkConnection(ConnectionConfig
config) //abstracttcpplugin.java
       ...
159. if (socket != null)
```

[Table of Contents] [Description]

Path 1040:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. final protected boolean checkConnection(ConnectionConfig
config) //abstracttcpplugin.java
       ...
159. if (socket != null)
```

[Table of Contents] [Description]

Path 1041:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
171. protected void closeSocket(Socket socket, ConnectionConfig
config) //abstracttcpplugin.java
      |...
      |173. if (socket != null)
```

[Table of Contents] [Description]

Path 1042:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
171. protected void closeSocket(Socket socket, ConnectionConfig
config) //abstracttcpplugin.java
      |...
      |173. if (socket != null)
```

[Table of Contents] [Description]

Path 1043:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
186. protected List getConnectionConfigList(Map qualifiers, InetAddress
address) //abstracttcpplugin.java
      |187. if (m_defaultPort == -1)
```

[Table of Contents] [Description]

Path 1044:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. final protected int getKeyedInteger(Map qualifiers, String key, int
defaultVal) //abstracttcpplugin.java
      |195. if (qualifiers == null)
```

[Table of Contents] [Description]

Path 1045:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
201. final protected int[] getKeyedIntegerArray(Map qualifiers, String key, int[]
defaultVal) //abstracttcpplugin.java
      |
```

```
202. if (qualifiers == null)
```

[Table of Contents] [Description]

Path 1046:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. final public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //abstracttcpplugin.java
...
265. if (qualifiers != null)
```

[Table of Contents] [Description]

Path 1047:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. final public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //abstracttcpplugin.java
...
265. if (qualifiers != null)
```

[Table of Contents] [Description]

Path 1048:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
289. final protected void saveKeyedInteger(Map qualifiers, String key, int
value) //abstracttcpplugin.java
290. if (qualifiers != null && !qualifiers.containsKey(key))
```

[Table of Contents] [Description]

Path 1049:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
289. final protected void saveKeyedInteger(Map qualifiers, String key, int
value) //abstracttcpplugin.java
290. if (qualifiers != null && !qualifiers.containsKey(key))
```

[Table of Contents] [Description]

Path 1050:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
```

```
ipAddr) throws SQLException //broadcasteventprocessor.java
    ...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1051:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
ipAddr) throws SQLException //broadcasteventprocessor.java
    ...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1052:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
ipAddr) throws SQLException //broadcasteventprocessor.java
    ...
253. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1053:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
ipAddr) throws SQLException //broadcasteventprocessor.java
    ...
253. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1054:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
ipAddr) throws SQLException //broadcasteventprocessor.java
    ...
255. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1055:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. private int countOtherInterfacesOnNode(Connection dbConn, long nodeId, String
ipAddr) throws SQLException //broadcasteventprocessor.java
     ...
255. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1056:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
293. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1057:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
293. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1058:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
299. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1059:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
299. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1060:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
301. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1061:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. private int countOtherServicesOnInterface(Connection dbConn, long nodeId,
String ipAddr, String service) throws SQLException //broadcasteventprocessor.java
     ...
301. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1062:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
     ...
336. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1063:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
     ...
336. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1064:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
```

```
      ...
342. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1065:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
342. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1066:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
344. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1067:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
318. private int countServicesOnOtherInterfaces(Connection dbConn, long nodeId,
String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
344. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1068:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1069:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
      383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1070:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
      409. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1071:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
      409. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1072:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
      411. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1073:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. private List createInterfaceOnNode(Connection dbConn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
```

```
FailedOperationException //broadcasteventprocessor.java
      ...
411. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1074:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
493. private List createNodeWithInterface(Connection conn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
496. if (nodeLabel == null)
```

[Table of Contents] [Description]

Path 1075:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
493. private List createNodeWithInterface(Connection conn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1076:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
493. private List createNodeWithInterface(Connection conn, String nodeLabel, String
ipaddr, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1077:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
614. private List doAddServiceMapping(Connection dbConn, String ipaddr, String
serviceName, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
      ...
630. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1078:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
614. private List doAddServiceMapping(Connection dbConn, String ipaddr, String
serviceName, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
630. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1079:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. private List doAddServiceToInterface(Connection dbConn, String ipaddr, String
serviceName, int serviceId, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
681. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1080:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. private List doAddServiceToInterface(Connection dbConn, String ipaddr, String
serviceName, int serviceId, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
681. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1081:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. private List doAddServiceToInterface(Connection dbConn, String ipaddr, String
serviceName, int serviceId, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
683. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1082:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. private List doAddServiceToInterface(Connection dbConn, String ipaddr, String
serviceName, int serviceId, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
     ...
683. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1083:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
732. private List doCreateInterfaceMappings(Connection dbConn, String nodeLabel,
String ipaddr, String hostName, long txNo) throws
SQLException //broadcasteventprocessor.java
     ...
751. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1084:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
732. private List doCreateInterfaceMappings(Connection dbConn, String nodeLabel,
String ipaddr, String hostName, long txNo) throws
SQLException //broadcasteventprocessor.java
     ...
751. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1085:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
801. private List doDeleteInterfaceMappings(Connection dbConn, String nodeLabel,
String ipaddr, String hostName, long txNo) throws
SQLException //broadcasteventprocessor.java
     ...
836. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1086:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
801. private List doDeleteInterfaceMappings(Connection dbConn, String nodeLabel,
String ipaddr, String hostName, long txNo) throws
SQLException //broadcasteventprocessor.java
```

```
...
836. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1087:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
934. private List doDeleteServiceMapping(Connection dbConn, String ipaddr, String
serviceName, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
...
950. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1088:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
934. private List doDeleteServiceMapping(Connection dbConn, String ipaddr, String
serviceName, long txNo) throws SQLException,
FailedOperationException //broadcasteventprocessor.java
...
950. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1089:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. private boolean existsInServerMap(Connection dbConn, String hostName, String
ipaddr) throws SQLException //broadcasteventprocessor.java
...
1043. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1090:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. private boolean existsInServerMap(Connection dbConn, String hostName, String
ipaddr) throws SQLException //broadcasteventprocessor.java
...
1043. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1091:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1056. private int[] findNodeIdForServiceAndInterface(Connection dbConn, String
ipaddr, String serviceName) throws SQLException //broadcasteventprocessor.java
      ...
1087. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1092:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1056. private int[] findNodeIdForServiceAndInterface(Connection dbConn, String
ipaddr, String serviceName) throws SQLException //broadcasteventprocessor.java
      ...
1087. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1093:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1056. private int[] findNodeIdForServiceAndInterface(Connection dbConn, String
ipaddr, String serviceName) throws SQLException //broadcasteventprocessor.java
      ...
1089. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1094:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1056. private int[] findNodeIdForServiceAndInterface(Connection dbConn, String
ipaddr, String serviceName) throws SQLException //broadcasteventprocessor.java
      ...
1089. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1095:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1103. private long[] findNodeIdsForInterfaceAndLabel(Connection dbConn, String
nodeLabel, String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
1126. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1096:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1103. private long[] findNodeIdsForInterfaceAndLabel(Connection dbConn, String
nodeLabel, String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
1126. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1097:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1103. private long[] findNodeIdsForInterfaceAndLabel(Connection dbConn, String
nodeLabel, String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
1128. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1098:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1103. private long[] findNodeIdsForInterfaceAndLabel(Connection dbConn, String
nodeLabel, String ipAddr) throws SQLException //broadcasteventprocessor.java
      ...
1128. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1099:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1160. private void handleAddInterface(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1165. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1160. private void handleAddInterface(Event event) throws
```

```
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1165. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1160. private void handleAddInterface(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1160. private void handleAddInterface(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1222. private void handleAddNode(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1234. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1222. private void handleAddNode(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1234. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1283. private void handleChangeService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1296. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1283. private void handleChangeService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1296. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1346. private void handleDeleteInterface(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1351. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1346. private void handleDeleteInterface(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1351. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1346. private void handleDeleteInterface(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1346. private void handleDeleteInterface(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1411. private void handleDeleteNode(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1415. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1411. private void handleDeleteNode(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
      1415. if (isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1411. private void handleDeleteNode(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
```

```
...
1422. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1411. private void handleDeleteNode(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
...
1422. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1478. private void handleDeleteService(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
...
1489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1478. private void handleDeleteService(Event e) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
...
1489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
...
1555. if (event.hasNodeid())
```

[Table of Contents] [Description]

Path 1118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      |...
      |1584. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      |...
      |1584. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      |...
      |1589. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      |...
      |1589. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      |...
```

```
1594. if (dbc != null) dbc.close();
```

[Table of Contents] [Description]

Path 1123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1548. private void handleForceRescan(Event event) throws
InsufficientInformationException //broadcasteventprocessor.java
      ...
1594. if (dbc != null) dbc.close();
```

[Table of Contents] [Description]

Path 1124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1670. private void handleUpdateServer(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1687. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1670. private void handleUpdateServer(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1687. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1742. private void handleUpdateService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      ...
1756. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1742. private void handleUpdateService(Event event) throws
InsufficientInformationException,
FailedOperationException //broadcasteventprocessor.java
      |...
      1756. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1811. private boolean interfaceExists(Connection dbConn, String nodeLabel, String
ipaddr) throws SQLException //broadcasteventprocessor.java
      |...
      1824. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1811. private boolean interfaceExists(Connection dbConn, String nodeLabel, String
ipaddr) throws SQLException //broadcasteventprocessor.java
      |...
      1824. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1811. private boolean interfaceExists(Connection dbConn, String nodeLabel, String
ipaddr) throws SQLException //broadcasteventprocessor.java
      |...
      1826. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1811. private boolean interfaceExists(Connection dbConn, String nodeLabel, String
ipaddr) throws SQLException //broadcasteventprocessor.java
      |...
      1826. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
      ...
1885. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
      ...
1885. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
      ...
1891. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
      ...
1891. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
    ...
1893. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1847. private List markAllServicesForInterfaceDeleted(Connection dbConn, String
source, long nodeId, String ipAddr, long txNo) throws
SQLException //broadcasteventprocessor.java
    ...
1893. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1916. private List markInterfaceDeleted(Connection dbConn, String source, long
nodeId, String ipAddr, long txNo) throws SQLException //broadcasteventprocessor.java
    ...
1927. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1916. private List markInterfaceDeleted(Connection dbConn, String source, long
nodeId, String ipAddr, long txNo) throws SQLException //broadcasteventprocessor.java
    ...
1927. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1916. private List markInterfaceDeleted(Connection dbConn, String source, long
nodeId, String ipAddr, long txNo) throws SQLException //broadcasteventprocessor.java
    ...
1930. if (count > 0)
```

[Table of Contents] [Description]

Path 1141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1916. private List markInterfaceDeleted(Connection dbConn, String source, long
nodeId, String ipAddr, long txNo) throws SQLException //broadcasteventprocessor.java
      ...
1935. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1916. private List markInterfaceDeleted(Connection dbConn, String source, long
nodeId, String ipAddr, long txNo) throws SQLException //broadcasteventprocessor.java
      ...
1935. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1958. private List markInterfacesAndServicesDeleted(Connection dbConn, String
source, long nodeId, long txNo) throws SQLException //broadcasteventprocessor.java
      ...
1988. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1958. private List markInterfacesAndServicesDeleted(Connection dbConn, String
source, long nodeId, long txNo) throws SQLException //broadcasteventprocessor.java
      ...
1988. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1958. private List markInterfacesAndServicesDeleted(Connection dbConn, String
```

```
source, long nodeId, long txNo) throws SQLException //broadcasteventprocessor.java
    ...
1990. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1958. private List markInterfacesAndServicesDeleted(Connection dbConn, String
source, long nodeId, long txNo) throws SQLException //broadcasteventprocessor.java
    ...
1990. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2010. private List markNodeDeleted(Connection dbConn, String source, long nodeId,
long txNo) throws SQLException //broadcasteventprocessor.java
    ...
2020. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2010. private List markNodeDeleted(Connection dbConn, String source, long nodeId,
long txNo) throws SQLException //broadcasteventprocessor.java
    ...
2020. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2010. private List markNodeDeleted(Connection dbConn, String source, long nodeId,
long txNo) throws SQLException //broadcasteventprocessor.java
    ...
2023. if (count > 0)
```

[Table of Contents] [Description]

Path 1150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2010. private List markNodeDeleted(Connection dbConn, String source, long nodeId,
long txNo) throws SQLException //broadcasteventprocessor.java
      |...
      |2028. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2010. private List markNodeDeleted(Connection dbConn, String source, long nodeId,
long txNo) throws SQLException //broadcasteventprocessor.java
      |...
      |2028. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2052. private List markServiceDeleted(Connection dbConn, String source, long nodeId,
String ipAddr, String service, long txNo) throws
SQLException //broadcasteventprocessor.java
      |...
      |2069. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2052. private List markServiceDeleted(Connection dbConn, String source, long nodeId,
String ipAddr, String service, long txNo) throws
SQLException //broadcasteventprocessor.java
      |...
      |2069. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2052. private List markServiceDeleted(Connection dbConn, String source, long nodeId,
String ipAddr, String service, long txNo) throws
SQLException //broadcasteventprocessor.java
      |
```

```
...
2072. if (count > 0)
```

[Table of Contents] [Description]

Path 1155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2052. private List markServiceDeleted(Connection dbConn, String source, long nodeId,
String ipAddr, String service, long txNo) throws
SQLException //broadcasteventprocessor.java
...
2077. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2052. private List markServiceDeleted(Connection dbConn, String source, long nodeId,
String ipAddr, String service, long txNo) throws
SQLException //broadcasteventprocessor.java
...
2077. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2092. private boolean nodeExists(Connection dbConn, String nodeLabel) throws
SQLException //broadcasteventprocessor.java
...
2104. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2092. private boolean nodeExists(Connection dbConn, String nodeLabel) throws
SQLException //broadcasteventprocessor.java
...
2104. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2092. private boolean nodeExists(Connection dbConn, String nodeLabel) throws
SQLException //broadcasteventprocessor.java
      ...
2106. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2092. private boolean nodeExists(Connection dbConn, String nodeLabel) throws
SQLException //broadcasteventprocessor.java
      ...
2106. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2119. private void notifyEventError(Event event, String msg, Exception
ex) //broadcasteventprocessor.java
      2120. if (!isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2135. private void notifyEventReceived(Event event) //broadcasteventprocessor.java
      2136. if (!isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2156. private void notifyEventSuccess(Event event) //broadcasteventprocessor.java
      2157. if (!isXmlRpcEnabled())
```

[Table of Contents] [Description]

Path 1164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2244. private boolean serviceMappingExists(Connection dbConn, String ipaddr, String
serviceName) throws SQLException //broadcasteventprocessor.java
      ...
2260. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2244. private boolean serviceMappingExists(Connection dbConn, String ipaddr, String
serviceName) throws SQLException //broadcasteventprocessor.java
      ...
2260. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
      ...
2287. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
      ...
2287. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
      ...
2293. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
    ...
2293. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
    ...
2301. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
    ...
2301. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 1172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
    ...
2303. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2273. private int verifyServiceExists(Connection dbConn, String serviceName) throws
SQLException, FailedOperationException //broadcasteventprocessor.java
```

```
...
2303. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
376. public void pause() //capsd.java
     377. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 1175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
80. public int getKeyedInteger(String key, int defaultVal) //connectionconfig.java
     81. if (m_qualifiers == null)
```

[Table of Contents] [Description]

Path 1176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public String getKeyedString(String key, String
defaultVal) //connectionconfig.java
     88. if (m_qualifiers == null)
```

[Table of Contents] [Description]

Path 1177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
94. public void saveKeyedInteger(String key, int value) //connectionconfig.java
     95. if (m_qualifiers != null && !m_qualifiers.containsKey(key))
```

[Table of Contents] [Description]

Path 1178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
94. public void saveKeyedInteger(String key, int value) //connectionconfig.java
     95. if (m_qualifiers != null && !m_qualifiers.containsKey(key))
```

[Table of Contents] [Description]

Path 1179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
     199. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
     ...
     246. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
     ...
     246. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
     ...
     260. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
     ...
     260. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
       ...
280. if ((m_changed & CHANGED_QUALIFIER) == CHANGED_QUALIFIER)
```

[Table of Contents] [Description]

Path 1185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. private void insert(Connection c) throws SQLException //dbifserviceentry.java
       ...
280. if ((m_changed & CHANGED_QUALIFIER) == CHANGED_QUALIFIER)
```

[Table of Contents] [Description]

Path 1186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
306. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
       ...
361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
       ...
361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     ...
361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     ...
389. if (m_source == SOURCE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     ...
389. if (m_source == SOURCE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     ...
389. if (m_source == SOURCE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     ...
396. if (m_notify == NOTIFY_UNKNOWN)
```

[Table of Contents] [Description]

Path 1194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     |...
     396. if (m_notify == NOTIFY_UNKNOWN)
```

[Table of Contents] [Description]

Path 1195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. private void update(Connection c) throws SQLException //dbifserviceentry.java
     |...
     396. if (m_notify == NOTIFY_UNKNOWN)
```

[Table of Contents] [Description]

Path 1196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |430. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
     457. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     |...
     471. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     ...
477. if (str != null && !rset.wasNull())
```

[Table of Contents] [Description]

Path 1200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     ...
485. if (str != null && !rset.wasNull())
```

[Table of Contents] [Description]

Path 1201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
429. private boolean load(Connection c) throws SQLException //dbifserviceentry.java
     ...
493. if (str != null && !rset.wasNull())
```

[Table of Contents] [Description]

Path 1202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. boolean hasIfIndexChanged() //dbifserviceentry.java
     714. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
747. boolean hasStatusChanged() //dbifserviceentry.java
     748. if ((m_changed & CHANGED_STATUS) == CHANGED_STATUS)
```

[Table of Contents] [Description]

Path 1204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
777. boolean hasSourceChanged() //dbifserviceentry.java
     |778. if ((m_changed & CHANGED_SOURCE) == CHANGED_SOURCE)
```

[Table of Contents] [Description]

Path 1205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
807. boolean hasNotifyChanged() //dbifserviceentry.java
     |808. if ((m_changed & CHANGED_NOTIFY) == CHANGED_NOTIFY)
```

[Table of Contents] [Description]

Path 1206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
831. boolean hasQualifierChanged() //dbifserviceentry.java
     |832. if ((m_changed & CHANGED_QUALIFIER) == CHANGED_QUALIFIER)
```

[Table of Contents] [Description]

Path 1207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
838. boolean updateQualifier(String newQualifier) //dbifserviceentry.java
     |...
     |841. if (!newQualifier.equals(m_qualifier))
```

[Table of Contents] [Description]

Path 1208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
862. void store() throws SQLException //dbifserviceentry.java
     |...
     |868. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
862. void store() throws SQLException //dbifserviceentry.java
     |...
```

```
868. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
862. void store() throws SQLException //dbifserviceentry.java
     ...
872. if (db != null)
```

[Table of Contents] [Description]

Path 1211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
862. void store() throws SQLException //dbifserviceentry.java
     ...
872. if (db != null)
```

[Table of Contents] [Description]

Path 1212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
205. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
261. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
```

```
261. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
264. if ((m_changed & CHANGED_HOSTNAME) == CHANGED_HOSTNAME)
```

[Table of Contents] [Description]

Path 1216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
264. if ((m_changed & CHANGED_HOSTNAME) == CHANGED_HOSTNAME)
```

[Table of Contents] [Description]

Path 1217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
270. if ((m_changed & CHANGED_STATUS) == CHANGED_STATUS)
```

[Table of Contents] [Description]

Path 1218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
     ...
270. if ((m_changed & CHANGED_STATUS) == CHANGED_STATUS)
```

[Table of Contents] [Description]

Path 1219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
```

```
...
285. if (stmt !=null) stmt.close();
```

[Table of Contents] [Description]

Path 1220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private void insert(Connection c) throws SQLException //dbipinterfaceentry.java
...
285. if (stmt !=null) stmt.close();
```

[Table of Contents] [Description]

Path 1221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
305. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
365. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
365. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
```

```
...
365. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
372. if (m_hostname != null)
```

[Table of Contents] [Description]

Path 1226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
372. if (m_hostname != null)
```

[Table of Contents] [Description]

Path 1227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
372. if (m_hostname != null)
```

[Table of Contents] [Description]

Path 1228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
379. if (m_managedState == STATE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
```

```
...
379. if (m_managedState == STATE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
379. if (m_managedState == STATE_UNKNOWN)
```

[Table of Contents] [Description]

Path 1231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
386. if (m_status == -1)
```

[Table of Contents] [Description]

Path 1232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
386. if (m_status == -1)
```

[Table of Contents] [Description]

Path 1233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
386. if (m_status == -1)
```

[Table of Contents] [Description]

Path 1234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
```

```
...
400. if (m_primaryState == SNMP_UNKNOWN)
```

[Table of Contents] [Description]

Path 1235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
400. if (m_primaryState == SNMP_UNKNOWN)
```

[Table of Contents] [Description]

Path 1236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
400. if (m_primaryState == SNMP_UNKNOWN)
```

[Table of Contents] [Description]

Path 1237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
410. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
...
410. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
```

```
      ...
410. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
      ...
421. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
304. private void update(Connection c) throws SQLException //dbipinterfaceentry.java
      ...
421. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      442. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      ...
476. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      |...
      |482. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      |...
      |488. if (str != null && rset.wasNull() == false)
```

[Table of Contents] [Description]

Path 1246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      |...
      |496. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. private boolean load(Connection c) throws
SQLException //dbipinterfaceentry.java
      |...
      |506. if (str != null && rset.wasNull() == false)
```

[Table of Contents] [Description]

Path 1248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
683. boolean hasIfIndexChanged() //dbipinterfaceentry.java
      |684. if ((m_changed & CHANGED_IFINDEX) == CHANGED_IFINDEX)
```

[Table of Contents] [Description]

Path 1249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
716. boolean hasHostnameChanged() //dbipinterfaceentry.java
     |717. if ((m_changed & CHANGED_HOSTNAME) == CHANGED_HOSTNAME)
```

[Table of Contents] [Description]

Path 1250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
723. boolean updateHostname(String newHostname) //dbipinterfaceentry.java
     |...
     |726. if (!newHostname.equals(m_hostname))
```

[Table of Contents] [Description]

Path 1251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
759. boolean hasManagedStateChanged() //dbipinterfaceentry.java
     |760. if ((m_changed & CHANGED_MANAGED) == CHANGED_MANAGED)
```

[Table of Contents] [Description]

Path 1252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
793. boolean hasStatusChanged() //dbipinterfaceentry.java
     |794. if ((m_changed & CHANGED_STATUS) == CHANGED_STATUS)
```

[Table of Contents] [Description]

Path 1253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
826. boolean hasPrimaryStateChanged() //dbipinterfaceentry.java
     |827. if ((m_changed & CHANGED_PRIMARY) == CHANGED_PRIMARY)
```

[Table of Contents] [Description]

Path 1254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. void store() throws SQLException //dbipinterfaceentry.java
     |
```

```
...
853. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. void store() throws SQLException //dbipinterfaceentry.java
...
853. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. void store() throws SQLException //dbipinterfaceentry.java
...
857. if (db != null)
```

[Table of Contents] [Description]

Path 1257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. void store() throws SQLException //dbipinterfaceentry.java
...
857. if (db != null)
```

[Table of Contents] [Description]

Path 1258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
876. void store(Connection db) throws SQLException //dbipinterfaceentry.java
...
878. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
876. void store(Connection db) throws SQLException //dbipinterfaceentry.java
```

```
...
878. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
876. void store(Connection db) throws SQLException //dbipinterfaceentry.java
...
878. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. DbIfServiceEntry[] getServices() throws SQLException //dbipinterfaceentry.java
...
894. if (db != null)
```

[Table of Contents] [Description]

Path 1262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. DbIfServiceEntry[] getServices() throws SQLException //dbipinterfaceentry.java
...
894. if (db != null)
```

[Table of Contents] [Description]

Path 1263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
904. DbIfServiceEntry[] getServices(Connection db) throws
SQLException //dbipinterfaceentry.java
...
915. if (entry != null)
```

[Table of Contents] [Description]

Path 1264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
904. DbIfServiceEntry[] getServices(Connection db) throws
SQLException //dbipinterfaceentry.java
      ...
      915. if (entry != null)
```

[Table of Contents] [Description]

Path 1265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
991. static DbIpInterfaceEntry get(int nid, InetAddress addr) throws
SQLException //dbipinterfaceentry.java
      ...
      998. if (db != null)
```

[Table of Contents] [Description]

Path 1266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
991. static DbIpInterfaceEntry get(int nid, InetAddress addr) throws
SQLException //dbipinterfaceentry.java
      ...
      998. if (db != null)
```

[Table of Contents] [Description]

Path 1267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1021. static DbIpInterfaceEntry get(int nid, InetAddress addr, int ifIndex) throws
SQLException //dbipinterfaceentry.java
      ...
      1028. if (db != null)
```

[Table of Contents] [Description]

Path 1268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1021. static DbIpInterfaceEntry get(int nid, InetAddress addr, int ifIndex) throws
SQLException //dbipinterfaceentry.java
      ...
      1028. if (db != null)
```

[Table of Contents] [Description]

Path 1269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1051. static DbIpInterfaceEntry get(Connection db, int nid, InetAddress addr) throws
SQLException //dbipinterfaceentry.java
      ...
1053. if (!entry.load(db))
```

[Table of Contents] [Description]

Path 1270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1075. static DbIpInterfaceEntry get(Connection db, int nid, InetAddress addr, int
ifIndex) throws SQLException //dbipinterfaceentry.java
      ...
1077. if (!entry.load(db))
```

[Table of Contents] [Description]

Path 1271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
      298. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
      ...
414. if ((m_changed & CHANGED_DPNAME) == CHANGED_DPNAME)
```

[Table of Contents] [Description]

Path 1273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
      ...
414. if ((m_changed & CHANGED_DPNAME) == CHANGED_DPNAME)
```

[Table of Contents] [Description]

Path 1274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
417. if ((m_changed & CHANGED_PARENT_ID) == CHANGED_PARENT_ID)
```

[Table of Contents] [Description]

Path 1275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
430. if ((m_changed & CHANGED_SYSOID) == CHANGED_SYSOID)
```

[Table of Contents] [Description]

Path 1276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
430. if ((m_changed & CHANGED_SYSOID) == CHANGED_SYSOID)
```

[Table of Contents] [Description]

Path 1277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
433. if ((m_changed & CHANGED_SYSNAME) == CHANGED_SYSNAME)
```

[Table of Contents] [Description]

Path 1278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
433. if ((m_changed & CHANGED_SYSNAME) == CHANGED_SYSNAME)
```

[Table of Contents] [Description]

Path 1279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
436. if ((m_changed & CHANGED_SYSDESCR) == CHANGED_SYSDESCR)
```

[Table of Contents] [Description]

Path 1280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
436. if ((m_changed & CHANGED_SYSDESCR) == CHANGED_SYSDESCR)
```

[Table of Contents] [Description]

Path 1281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
439. if ((m_changed & CHANGED_SYSLOC) == CHANGED_SYSLOC)
```

[Table of Contents] [Description]

Path 1282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
439. if ((m_changed & CHANGED_SYSLOC) == CHANGED_SYSLOC)
```

[Table of Contents] [Description]

Path 1283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
442. if ((m_changed & CHANGED_SYSCONTACT) == CHANGED_SYSCONTACT)
```

[Table of Contents] [Description]

Path 1284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
442. if ((m_changed & CHANGED_SYSCONTACT) == CHANGED_SYSCONTACT)
```

[Table of Contents] [Description]

Path 1285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
445. if ((m_changed & CHANGED_LABEL) == CHANGED_LABEL)
```

[Table of Contents] [Description]

Path 1286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
445. if ((m_changed & CHANGED_LABEL) == CHANGED_LABEL)
```

[Table of Contents] [Description]

Path 1287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
451. if ((m_changed & CHANGED_NETBIOS_NAME) == CHANGED_NETBIOS_NAME)
```

[Table of Contents] [Description]

Path 1288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
451. if ((m_changed & CHANGED_NETBIOS_NAME) == CHANGED_NETBIOS_NAME)
```

[Table of Contents] [Description]

Path 1289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
454. if ((m_changed & CHANGED_DOMAIN_NAME) == CHANGED_DOMAIN_NAME)
```

[Table of Contents] [Description]

Path 1290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
454. if ((m_changed & CHANGED_DOMAIN_NAME) == CHANGED_DOMAIN_NAME)
```

[Table of Contents] [Description]

Path 1291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
457. if ((m_changed & CHANGED_OS) == CHANGED_OS)
```

[Table of Contents] [Description]

Path 1292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
457. if ((m_changed & CHANGED_OS) == CHANGED_OS)
```

[Table of Contents] [Description]

Path 1293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
484. if (rset != null) rset.close();
```

[Table of Contents] [Description]

Path 1294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
484. if (rset != null) rset.close();
```

[Table of Contents] [Description]

Path 1295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
486. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. private void insert(Connection c) throws SQLException //dbnodeentry.java
     ...
486. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     501. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
594. if ((m_changed & CHANGED_PARENT_ID) == CHANGED_PARENT_ID)
```

[Table of Contents] [Description]

Path 1299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
612. if (m_sysoid == null)
```

[Table of Contents] [Description]

Path 1300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
612. if (m_sysoid == null)
```

[Table of Contents] [Description]

Path 1301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
612. if (m_sysoid == null)
```

[Table of Contents] [Description]

Path 1302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
619. if (m_sysname == null)
```

[Table of Contents] [Description]

Path 1303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
619. if (m_sysname == null)
```

[Table of Contents] [Description]

Path 1304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
619. if (m_sysname == null)
```

[Table of Contents] [Description]

Path 1305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
626. if (m_sysdescr == null)
```

[Table of Contents] [Description]

Path 1306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
626. if (m_sysdescr == null)
```

[Table of Contents] [Description]

Path 1307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
626. if (m_sysdescr == null)
```

[Table of Contents] [Description]

Path 1308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
633. if (m_syslocation == null)
```

[Table of Contents] [Description]

Path 1309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
633. if (m_syslocation == null)
```

[Table of Contents] [Description]

Path 1310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
633. if (m_syslocation == null)
```

[Table of Contents] [Description]

Path 1311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
640. if (m_syscontact == null)
```

[Table of Contents] [Description]

Path 1312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
640. if (m_syscontact == null)
```

[Table of Contents] [Description]

Path 1313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
640. if (m_syscontact == null)
```

[Table of Contents] [Description]

Path 1314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
       ...
647. if (m_label == null)
```

[Table of Contents] [Description]

Path 1315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
       ...
647. if (m_label == null)
```

[Table of Contents] [Description]

Path 1316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
       ...
647. if (m_label == null)
```

[Table of Contents] [Description]

Path 1317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
       ...
658. if (m_nbName == null)
```

[Table of Contents] [Description]

Path 1318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
       ...
658. if (m_nbName == null)
```

[Table of Contents] [Description]

Path 1319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
658. if (m_nbName == null)
```

[Table of Contents] [Description]

Path 1320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
665. if (m_nbDomainName == null)
```

[Table of Contents] [Description]

Path 1321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
665. if (m_nbDomainName == null)
```

[Table of Contents] [Description]

Path 1322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
665. if (m_nbDomainName == null)
```

[Table of Contents] [Description]

Path 1323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
     ...
672. if (m_os == null)
```

[Table of Contents] [Description]

Path 1324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
      ...
672. if (m_os == null)
```

[Table of Contents] [Description]

Path 1325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
      ...
672. if (m_os == null)
```

[Table of Contents] [Description]

Path 1326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
      ...
698. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. private void update(Connection c) throws SQLException //dbnodeentry.java
      ...
698. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      714. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
744. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
750. if (str != null && !rset.wasNull())
```

[Table of Contents] [Description]

Path 1331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
758. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
764. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
770. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
776. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
782. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
788. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
794. if (rset.wasNull() || str == null)
```

[Table of Contents] [Description]

Path 1338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
     ...
802. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
808. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
713. private boolean load(Connection c) throws SQLException //dbnodeentry.java
      ...
814. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1007. boolean hasParentIdChanged() //dbnodeentry.java
      1008. if ((m_changed & CHANGED_PARENT_ID) == CHANGED_PARENT_ID)
```

[Table of Contents] [Description]

Path 1342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1041. boolean hasNodeTypeChanged() //dbnodeentry.java
      1042. if ((m_changed & CHANGED_TYPE) == CHANGED_TYPE)
```

[Table of Contents] [Description]

Path 1343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1071. boolean hasSystemOIDChanged() //dbnodeentry.java
      1072. if ((m_changed & CHANGED_SYSOID) == CHANGED_SYSOID)
```

[Table of Contents] [Description]

Path 1344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1082. boolean updateSystemOID(String newsysoid) //dbnodeentry.java
```

```
1083. if (newsysoid == null || newsysoid.equals(m_sysoid))
```

[Table of Contents] [Description]

Path 1345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1109. boolean hasSystemNameChanged() //dbnodeentry.java
1110. if ((m_changed & CHANGED_SYSNAME) == CHANGED_SYSNAME)
```

[Table of Contents] [Description]

Path 1346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1116. boolean updateSystemName(String newsysname) //dbnodeentry.java
1117. if (newsysname == null || newsysname.equals(m_sysname))
```

[Table of Contents] [Description]

Path 1347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1143. boolean hasSystemDescriptionChanged() //dbnodeentry.java
1144. if ((m_changed & CHANGED_SYSDESCR) == CHANGED_SYSDESCR)
```

[Table of Contents] [Description]

Path 1348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1150. boolean updateSystemDescription(String newsysdescr) //dbnodeentry.java
1151. if (newsysdescr == null || newsysdescr.equals(m_sysdescr))
```

[Table of Contents] [Description]

Path 1349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1177. boolean hasSystemLocationChanged() //dbnodeentry.java
1178. if ((m_changed & CHANGED_SYSLOC) == CHANGED_SYSLOC)
```

[Table of Contents] [Description]

Path 1350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1184. boolean updateSystemLocation(String newsyslocation) //dbnodeentry.java
      1185. if (newsyslocation == null || newsyslocation.equals(m_syslocation))
```

[Table of Contents] [Description]

Path 1351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1211. boolean hasSystemContactChanged() //dbnodeentry.java
      1212. if ((m_changed & CHANGED_SYSCONTACT) == CHANGED_SYSCONTACT)
```

[Table of Contents] [Description]

Path 1352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1218. boolean updateSystemContact(String newsyscontact) //dbnodeentry.java
      1219. if (newsyscontact == null || newsyscontact.equals(m_syscontact))
```

[Table of Contents] [Description]

Path 1353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1245. boolean hasLabelChanged() //dbnodeentry.java
      1246. if ((m_changed & CHANGED_LABEL) == CHANGED_LABEL)
```

[Table of Contents] [Description]

Path 1354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1252. boolean updateLabel(String newlabel) //dbnodeentry.java
      ...
      1255. if (!newlabel.equals(m_label))
```

[Table of Contents] [Description]

Path 1355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1288. boolean hasLabelSourceChanged() //dbnodeentry.java
      1289. if ((m_changed & CHANGED_LABEL_SOURCE) == CHANGED_LABEL_SOURCE)
```

[Table of Contents] [Description]

Path 1356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1325. boolean hasNetBIOSNameChanged() //dbnodeentry.java
      1326. if ((m_changed & CHANGED_NETBIOS_NAME) == CHANGED_NETBIOS_NAME)
```

[Table of Contents] [Description]

Path 1357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1332. boolean updateNetBIOSName(String newnbName) //dbnodeentry.java
      ...
      1335. if (!newnbName.toUpperCase().equals(m_nbName))
```

[Table of Contents] [Description]

Path 1358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1368. boolean hasDomainNameChanged() //dbnodeentry.java
      1369. if ((m_changed & CHANGED_DOMAIN_NAME) == CHANGED_DOMAIN_NAME)
```

[Table of Contents] [Description]

Path 1359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1375. boolean updateDomainName(String domain) //dbnodeentry.java
      ...
      1378. if (!domain.equals(m_nbDomainName))
```

[Table of Contents] [Description]

Path 1360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1411. boolean hasOSChanged() //dbnodeentry.java
      1412. if ((m_changed & CHANGED_NETBIOS_NAME) == CHANGED_NETBIOS_NAME)
```

[Table of Contents] [Description]

Path 1361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1418. boolean updateOS(String newos) //dbnodeentry.java
      ...
1421. if (!newos.equals(m_os))
```

[Table of Contents] [Description]

Path 1362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1502. void store() throws SQLException //dbnodeentry.java
      ...
1508. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1502. void store() throws SQLException //dbnodeentry.java
      ...
1508. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1502. void store() throws SQLException //dbnodeentry.java
      ...
1512. if (db != null)
```

[Table of Contents] [Description]

Path 1365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1502. void store() throws SQLException //dbnodeentry.java
      ...
1512. if (db != null)
```

[Table of Contents] [Description]

Path 1366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1531. void store(Connection db) throws SQLException //dbnodeentry.java
      ...
1535. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1531. void store(Connection db) throws SQLException //dbnodeentry.java
      ...
1535. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1531. void store(Connection db) throws SQLException //dbnodeentry.java
      ...
1535. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1542. DbIpInterfaceEntry[] getInterfaces() throws SQLException //dbnodeentry.java
      ...
1551. if (db != null)
```

[Table of Contents] [Description]

Path 1370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1542. DbIpInterfaceEntry[] getInterfaces() throws SQLException //dbnodeentry.java
      ...
1551. if (db != null)
```

[Table of Contents] [Description]

Path 1371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1571. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1583. if (ifIndex == -1)
```

[Table of Contents] [Description]

Path 1373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1588. if (entry != null)
```

[Table of Contents] [Description]

Path 1374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1561. DbIpInterfaceEntry[] getInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1588. if (entry != null)
```

[Table of Contents] [Description]

Path 1375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1599. DbIpInterfaceEntry[] getManagedInterfaces() throws
```

```
SQLException //dbnodeentry.java
      ...
1608. if (db != null)
```

[Table of Contents] [Description]

Path 1376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1599. DbIpInterfaceEntry[] getManagedInterfaces() throws
SQLException //dbnodeentry.java
      ...
1608. if (db != null)
```

[Table of Contents] [Description]

Path 1377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1628. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1640. if (ifIndex == -1)
```

[Table of Contents] [Description]

Path 1379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1645. if (entry != null)
```

[Table of Contents] [Description]

Path 1380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1618. DbIpInterfaceEntry[] getManagedInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1645. if (entry != null)
```

[Table of Contents] [Description]

Path 1381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1656. static DbIpInterfaceEntry getPrimarySnmpInterface(DbIpInterfaceEntry[]
ipInterfaces) //dbnodeentry.java
      1657. if (ipInterfaces == null)
```

[Table of Contents] [Description]

Path 1382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1667. DbSnmpInterfaceEntry[] getSnmpInterfaces() throws
SQLException //dbnodeentry.java
      ...
1676. if (db != null)
```

[Table of Contents] [Description]

Path 1383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1667. DbSnmpInterfaceEntry[] getSnmpInterfaces() throws
SQLException //dbnodeentry.java
      ...
1676. if (db != null)
```

[Table of Contents] [Description]

Path 1384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1696. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1711. if (entry != null)
```

[Table of Contents] [Description]

Path 1386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1686. DbSnmpInterfaceEntry[] getSnmpInterfaces(Connection db) throws
SQLException //dbnodeentry.java
      ...
1711. if (entry != null)
```

[Table of Contents] [Description]

Path 1387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1742. static DbNodeEntry create(String poller) //dbnodeentry.java
      1743. if (poller == null)
```

[Table of Contents] [Description]

Path 1388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1805. static DbNodeEntry get(int nid, String dpName) throws
SQLException //dbnodeentry.java
      ...
1812. if (db != null)
```

[Table of Contents] [Description]

Path 1389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1805. static DbNodeEntry get(int nid, String dpName) throws
SQLException //dbnodeentry.java
      ...
1812. if (db != null)
```

[Table of Contents] [Description]

Path 1390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1852. static DbNodeEntry get(Connection db, int nid, String dpName) throws
SQLException //dbnodeentry.java
      ...
1854. if (!entry.load(db))
```

[Table of Contents] [Description]

Path 1391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      155. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
230. if ((m_changed & CHANGED_IFADDRESS) == CHANGED_IFADDRESS)
```

[Table of Contents] [Description]

Path 1393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
230. if ((m_changed & CHANGED_IFADDRESS) == CHANGED_IFADDRESS)
```

[Table of Contents] [Description]

Path 1394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
```

```
...
230. if ((m_changed & CHANGED_IFADDRESS) == CHANGED_IFADDRESS)
```

[Table of Contents] [Description]

Path 1395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
234. if ((m_changed & CHANGED_NETMASK) == CHANGED_NETMASK)
```

[Table of Contents] [Description]

Path 1396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
234. if ((m_changed & CHANGED_NETMASK) == CHANGED_NETMASK)
```

[Table of Contents] [Description]

Path 1397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
237. if ((m_changed & CHANGED_PHYSADDR) == CHANGED_PHYSADDR)
```

[Table of Contents] [Description]

Path 1398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
237. if ((m_changed & CHANGED_PHYSADDR) == CHANGED_PHYSADDR)
```

[Table of Contents] [Description]

Path 1399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
240. if ((m_changed & CHANGED_DESCRIPTION) == CHANGED_DESCRIPTION)
```

[Table of Contents] [Description]

Path 1400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
240. if ((m_changed & CHANGED_DESCRIPTION) == CHANGED_DESCRIPTION)
```

[Table of Contents] [Description]

Path 1401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
243. if ((m_changed & CHANGED_IFTYPE) == CHANGED_IFTYPE)
```

[Table of Contents] [Description]

Path 1402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
243. if ((m_changed & CHANGED_IFTYPE) == CHANGED_IFTYPE)
```

[Table of Contents] [Description]

Path 1403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
246. if ((m_changed & CHANGED_IFNAME) == CHANGED_IFNAME)
```

[Table of Contents] [Description]

Path 1404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      246. if ((m_changed & CHANGED_IFNAME) == CHANGED_IFNAME)
```

[Table of Contents] [Description]

Path 1405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      249. if ((m_changed & CHANGED_IFSPEED) == CHANGED_IFSPEED)
```

[Table of Contents] [Description]

Path 1406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      249. if ((m_changed & CHANGED_IFSPEED) == CHANGED_IFSPEED)
```

[Table of Contents] [Description]

Path 1407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      252. if ((m_changed & CHANGED_IFADMINSTATUS) == CHANGED_IFADMINSTATUS)
```

[Table of Contents] [Description]

Path 1408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
```

```
...
252. if ((m_changed & CHANGED_IFADMINSTATUS) == CHANGED_IFADMINSTATUS)
```

[Table of Contents] [Description]

Path 1409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
255. if ((m_changed & CHANGED_IFOPERSTATUS) == CHANGED_IFOPERSTATUS)
```

[Table of Contents] [Description]

Path 1410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
255. if ((m_changed & CHANGED_IFOPERSTATUS) == CHANGED_IFOPERSTATUS)
```

[Table of Contents] [Description]

Path 1411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
258. if ((m_changed & CHANGED_IFALIAS) == CHANGED_IFALIAS)
```

[Table of Contents] [Description]

Path 1412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. private void insert(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
258. if ((m_changed & CHANGED_IFALIAS) == CHANGED_IFALIAS)
```

[Table of Contents] [Description]

Path 1413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     284. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
     361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
     361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
     361. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
     368. if (m_netmask == null)
```

[Table of Contents] [Description]

Path 1418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      368. if (m_netmask == null)
```

[Table of Contents] [Description]

Path 1419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      368. if (m_netmask == null)
```

[Table of Contents] [Description]

Path 1420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      375. if (m_physAddr == null)
```

[Table of Contents] [Description]

Path 1421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      375. if (m_physAddr == null)
```

[Table of Contents] [Description]

Path 1422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
```

```
375. if (m_physAddr == null)
```

[Table of Contents] [Description]

Path 1423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
382. if (m_ifDescription == null)
```

[Table of Contents] [Description]

Path 1424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
382. if (m_ifDescription == null)
```

[Table of Contents] [Description]

Path 1425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
382. if (m_ifDescription == null)
```

[Table of Contents] [Description]

Path 1426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
389. if (m_ifType == -1)
```

[Table of Contents] [Description]

Path 1427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      389. if (m_ifType == -1)
```

[Table of Contents] [Description]

Path 1428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      389. if (m_ifType == -1)
```

[Table of Contents] [Description]

Path 1429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      396. if (m_ifName == null)
```

[Table of Contents] [Description]

Path 1430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      396. if (m_ifName == null)
```

[Table of Contents] [Description]

Path 1431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      396. if (m_ifName == null)
```

[Table of Contents] [Description]

Path 1432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |443. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |443. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |443. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |410. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     |...
```

```
410. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
410. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
417. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
417. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
...
417. if (m_ifSpeed == -1)
```

[Table of Contents] [Description]

Path 1441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      424. if (m_ifAlias == null)
```

[Table of Contents] [Description]

Path 1442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      424. if (m_ifAlias == null)
```

[Table of Contents] [Description]

Path 1443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      424. if (m_ifAlias == null)
```

[Table of Contents] [Description]

Path 1444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      432. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      |
      ...
      432. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
      432. if (m_ifIndex == -1)
```

[Table of Contents] [Description]

Path 1447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
      443. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. private void update(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
      443. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 1449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      464. if (!m_fromDb)
```

[Table of Contents] [Description]

Path 1450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
      ...
      517. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
523. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
529. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
535. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
     ...
541. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
```

```
SQLException //dbsnmpinterfaceentry.java
    ...
547. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
553. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
559. if (rset.wasNull())
```

[Table of Contents] [Description]

Path 1458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
564. if (rset != null ) rset.close();
```

[Table of Contents] [Description]

Path 1459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. private boolean load(Connection c) throws
SQLException //dbsnmpinterfaceentry.java
    ...
564. if (rset != null ) rset.close();
```

[Table of Contents] [Description]

Path 1460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
662. boolean hasIfAddressChanged() //dbsnmpinterfaceentry.java
     663. if ((m_changed & CHANGED_IFADDRESS) == CHANGED_IFADDRESS)
```

[Table of Contents] [Description]

Path 1461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. boolean updateIfAddress(InetAddress addr) //dbsnmpinterfaceentry.java
     670. if (addr == null || addr.equals(m_ipAddr))
```

[Table of Contents] [Description]

Path 1462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
701. boolean hasNetmaskChanged() //dbsnmpinterfaceentry.java
     702. if ((m_changed & CHANGED_NETMASK) == CHANGED_NETMASK)
```

[Table of Contents] [Description]

Path 1463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
708. boolean updateNetmask(InetAddress newNetmask) //dbsnmpinterfaceentry.java
     709. if (newNetmask == null || newNetmask.equals(m_netmask))
```

[Table of Contents] [Description]

Path 1464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
726. boolean hasPhysicalAddressChanged() //dbsnmpinterfaceentry.java
     727. if ((m_changed & CHANGED_PHYSADDR) == CHANGED_PHYSADDR)
```

[Table of Contents] [Description]

Path 1465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
733. boolean updatePhysicalAddress(String newPhysAddr) //dbsnmpinterfaceentry.java
     734. if (newPhysAddr == null || newPhysAddr.equals(m_physAddr))
```

[Table of Contents] [Description]

Path 1466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
751. boolean hasDescriptionChanged() //dbsnmpinterfaceentry.java
     |752. if ((m_changed & CHANGED_DESCRIPTION) == CHANGED_DESCRIPTION)
```

[Table of Contents] [Description]

Path 1467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
758. boolean updateDescription(String newIfDescription) //dbsnmpinterfaceentry.java
     |759. if (newIfDescription == null || newIfDescription.equals(m_ifDescription))
```

[Table of Contents] [Description]

Path 1468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
776. boolean hasNameChanged() //dbsnmpinterfaceentry.java
     |777. if ((m_changed & CHANGED_IFNAME) == CHANGED_IFNAME)
```

[Table of Contents] [Description]

Path 1469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
783. boolean updateName(String newIfName) //dbsnmpinterfaceentry.java
     |784. if (newIfName == null || newIfName.equals(m_ifName))
```

[Table of Contents] [Description]

Path 1470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
801. boolean hasTypeChanged() //dbsnmpinterfaceentry.java
     |802. if ((m_changed & CHANGED_IFTYPE) == CHANGED_IFTYPE)
```

[Table of Contents] [Description]

Path 1471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
808. boolean updateType(int newIfType) //dbsnmpinterfaceentry.java
     809. if (newIfType == -1 || newIfType == m_ifType)
```

[Table of Contents] [Description]

Path 1472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
826. boolean hasSpeedChanged() //dbsnmpinterfaceentry.java
     827. if ((m_changed & CHANGED_IFSPEED) == CHANGED_IFSPEED)
```

[Table of Contents] [Description]

Path 1473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
833. boolean updateSpeed(int newIfSpeed) //dbsnmpinterfaceentry.java
     834. if (newIfSpeed == -1 || newIfSpeed == m_ifSpeed)
```

[Table of Contents] [Description]

Path 1474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
851. boolean hasAdminStatusChanged() //dbsnmpinterfaceentry.java
     852. if ((m_changed & CHANGED_IFADMINSTATUS) == CHANGED_IFADMINSTATUS)
```

[Table of Contents] [Description]

Path 1475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
858. boolean updateAdminStatus(int newIfAdminStatus) //dbsnmpinterfaceentry.java
     859. if (newIfAdminStatus == -1 || newIfAdminStatus == m_ifAdminStatus)
```

[Table of Contents] [Description]

Path 1476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
876. boolean hasOperationalStatusChanged() //dbsnmpinterfaceentry.java
     877. if ((m_changed & CHANGED_IFOPERSTATUS) == CHANGED_IFOPERSTATUS)
```

[Table of Contents] [Description]

Path 1477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
883. boolean updateOperationalStatus(int
newIfOperStatus) //dbsnmpinterfaceentry.java
     |884. if (newIfOperStatus == -1 || newIfOperStatus == m_ifOperStatus)
```

[Table of Contents] [Description]

Path 1478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
901. boolean hasAliasChanged() //dbsnmpinterfaceentry.java
     |902. if ((m_changed & CHANGED_IFALIAS) == CHANGED_IFALIAS)
```

[Table of Contents] [Description]

Path 1479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
908. boolean updateAlias(String newIfAlias) //dbsnmpinterfaceentry.java
     |909. if (newIfAlias == null || newIfAlias.equals(m_ifAlias))
```

[Table of Contents] [Description]

Path 1480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
     |...
     |930. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
     |...
     |930. if (db.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 1482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
       ...
934. if (db != null)
```

[Table of Contents] [Description]

Path 1483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
       ...
934. if (db != null)
```

[Table of Contents] [Description]

Path 1484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. void store(Connection db) throws SQLException //dbsnmpinterfaceentry.java
       ...
955. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. void store(Connection db) throws SQLException //dbsnmpinterfaceentry.java
       ...
955. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. void store(Connection db) throws SQLException //dbsnmpinterfaceentry.java
       ...
955. if (m_fromDb)
```

[Table of Contents] [Description]

Path 1487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. static DbSnmpInterfaceEntry get(int nid, int ifIndex) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |997. if (db != null)
```

[Table of Contents] [Description]

Path 1488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. static DbSnmpInterfaceEntry get(int nid, int ifIndex) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |997. if (db != null)
```

[Table of Contents] [Description]

Path 1489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1020. static DbSnmpInterfaceEntry get(Connection db, int nid, int ifIndex) throws
SQLException //dbsnmpinterfaceentry.java
     |...
     |1022. if (!entry.load(db))
```

[Table of Contents] [Description]

Path 1490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. private boolean isServer(InetAddress host, int retries, int
timeout) //dhcpplugin.java
     |...
     |130. if (responseTime >= 0)
```

[Table of Contents] [Description]

Path 1491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public boolean isProtocolSupported(InetAddress host, Map
qualifiers) //dhcpplugin.java
     |...
```

```
190. if (isAServer && qualifiers != null)
```

[Table of Contents] [Description]

Path 1492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public boolean isProtocolSupported(InetAddress host, Map
qualifiers) //dhcpplugin.java
...
190. if (isAServer && qualifiers != null)
```

[Table of Contents] [Description]

Path 1493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
...
156. if (socket != null)
```

[Table of Contents] [Description]

Path 1494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
...
156. if (socket != null)
```

[Table of Contents] [Description]

Path 1495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
209. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //dnsplugin.java
...
222. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
209. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //dnsplugin.java
      |...
      222. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. protected boolean preconnectCheck(ConnectionConfig
tcpConfig) //dominoiiopplugin.java
      |...
      186. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. protected boolean preconnectCheck(ConnectionConfig
tcpConfig) //dominoiiopplugin.java
      |...
      186. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
200. private String retrieveIORText(String host, int port) throws
IOException //dominoiiopplugin.java
      |...
      221. if (!IOR.startsWith("IOR:"))
```

[Table of Contents] [Description]

Path 1500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. static public void checkEventId(Event e) throws
InsufficientInformationException //eventutils.java
      |89. if (e == null)
```

[Table of Contents] [Description]

Path 1501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. static public void checkEventId(Event e) throws
InsufficientInformationException //eventutils.java
      ...
      92. if (!e.hasDbid())
```

[Table of Contents] [Description]

Path 1502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. static public void checkInterface(Event e) throws
InsufficientInformationException //eventutils.java
      105. if (e == null)
```

[Table of Contents] [Description]

Path 1503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. static public void checkInterface(Event e) throws
InsufficientInformationException //eventutils.java
      ...
      108. if (e.getInterface() == null || e.getInterface().length() == 0)
```

[Table of Contents] [Description]

Path 1504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. static public void checkHost(Event e) throws
InsufficientInformationException //eventutils.java
      121. if (e == null)
```

[Table of Contents] [Description]

Path 1505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. static public void checkHost(Event e) throws
InsufficientInformationException //eventutils.java
      ...
      124. if (e.getHost() == null || e.getHost().length() == 0)
```

[Table of Contents] [Description]

Path 1506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. static public void checkNodeId(Event e) throws
InsufficientInformationException //eventutils.java
      |137. if (e == null)
```

[Table of Contents] [Description]

Path 1507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. static public void checkNodeId(Event e) throws
InsufficientInformationException //eventutils.java
      |...
      |140. if (!e.hasNodeid())
```

[Table of Contents] [Description]

Path 1508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public static void checkService(Event e) throws
InsufficientInformationException //eventutils.java
      |153. if (e == null)
```

[Table of Contents] [Description]

Path 1509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public static void checkService(Event e) throws
InsufficientInformationException //eventutils.java
      |...
      |156. if (e.getService() == null || e.getService().length() == 0)
```

[Table of Contents] [Description]

Path 1510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. public static long getEventID(Event e) //eventutils.java
      |...
      |402. if (e.hasDbid())
```

[Table of Contents] [Description]

Path 1511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
420. public static long getLongParm(Event e, String parmName, long
defaultValue) //eventutils.java
      ...
423. if (longVal == null)
```

[Table of Contents] [Description]

Path 1512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
441. public static long getNodeId(Event e) //eventutils.java
      ...
444. if (e.hasNodeid())
```

[Table of Contents] [Description]

Path 1513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
475. public static String getParm(Event e, String parmName, String
defaultValue) //eventutils.java
      ...
477. if (parms == null)
```

[Table of Contents] [Description]

Path 1514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public static void requireParm(Event e, String parmName) throws
InsufficientInformationException //eventutils.java
      ...
509. if (parms == null)
```

[Table of Contents] [Description]

Path 1515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
538. public static void sendEvent(Event newEvent, String callerUei, long txNo,
boolean isXmlRpcEnabled) //eventutils.java
      ...
```

```
544. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
538. public static void sendEvent(Event newEvent, String callerUei, long txNo,
boolean isXmlRpcEnabled) //eventutils.java
    ...
544. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
566. public static Event createNodeAddedEvent(DbNodeEntry nodeEntry, long
txNo) //eventutils.java
    ...
568. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
566. public static Event createNodeAddedEvent(DbNodeEntry nodeEntry, long
txNo) //eventutils.java
    ...
568. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
617. public static Event createNodeGainedInterfaceEvent(DbNodeEntry nodeEntry,
InetAddress ifaddr, long txNo) //eventutils.java
    ...
619. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
617. public static Event createNodeGainedInterfaceEvent(DbNodeEntry nodeEntry,
InetAddress ifaddr, long txNo) //eventutils.java
      |
      ...
      619. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
664. public static Event createNodeDeletedEvent(String source, int nodeId, String
hostName, String nodeLabel, long txNo) //eventutils.java
      |
      ...
      666. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
664. public static Event createNodeDeletedEvent(String source, int nodeId, String
hostName, String nodeLabel, long txNo) //eventutils.java
      |
      ...
      666. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
664. public static Event createNodeDeletedEvent(String source, int nodeId, String
hostName, String nodeLabel, long txNo) //eventutils.java
      |
      ...
      696. if ((nodeLabel != null) && (((new Long(txNo)).toString()) != null))
```

[Table of Contents] [Description]

Path 1524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
664. public static Event createNodeDeletedEvent(String source, int nodeId, String
hostName, String nodeLabel, long txNo) //eventutils.java
      |
      ...
      696. if ((nodeLabel != null) && (((new Long(txNo)).toString()) != null))
```

[Table of Contents] [Description]

Path 1525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
715. public static Event createAndSendDeleteNodeEvent(String source, String
nodeLabel, String hostName, long txNo) //eventutils.java
     ...
717. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
715. public static Event createAndSendDeleteNodeEvent(String source, String
nodeLabel, String hostName, long txNo) //eventutils.java
     ...
717. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
760. public static Event createForceRescanEvent(String hostName, long
nodeId) //eventutils.java
     ...
762. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
760. public static Event createForceRescanEvent(String hostName, long
nodeId) //eventutils.java
     ...
762. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
790. public static Event createAndSendInterfaceDeletedEvent(String source, int
nodeId, String ipaddr, String hostName, long txNo) //eventutils.java
     ...
```

```
792. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
790. public static Event createAndSendInterfaceDeletedEvent(String source, int
nodeId, String ipaddr, String hostName, long txNo) //eventutils.java
...
792. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
835. public static Event createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
InetAddress ifaddr, String service, long txNo) //eventutils.java
...
837. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
835. public static Event createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
InetAddress ifaddr, String service, long txNo) //eventutils.java
...
837. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
922. public static Event createAndSendDeleteServiceEvent(String source, DbNodeEntry
nodeEntry, InetAddress ifaddr, String service, String hostName, long
txNo) //eventutils.java
...
924. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
922. public static Event createAndSendDeleteServiceEvent(String source, DbNodeEntry
nodeEntry, InetAddress ifaddr, String service, String hostName, long
txNo) //eventutils.java
     ...
924. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
987. public static Event createAddInterfaceEvent(String source, String nodeLabel,
String ipaddr, String hostName, long txNo) //eventutils.java
     ...
989. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
987. public static Event createAddInterfaceEvent(String source, String nodeLabel,
String ipaddr, String hostName, long txNo) //eventutils.java
     ...
989. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1039. public static Event createAndSendDeleteInterfaceEvent(String source, String
nodeLabel, String ipaddr, String hostName, long txNo) //eventutils.java
     ...
1041. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1039. public static Event createAndSendDeleteInterfaceEvent(String source, String
nodeLabel, String ipaddr, String hostName, long txNo) //eventutils.java
     ...
1041. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1093. public static Event createChangeServiceEvent(String source, String ipaddr,
String service, String action, String hostName, long txNo) //eventutils.java
      ...
1095. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1093. public static Event createChangeServiceEvent(String source, String ipaddr,
String service, String action, String hostName, long txNo) //eventutils.java
      ...
1095. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
130. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //ftpplugin.java
      ...
196. if (rc > 99 && rc < 600)
```

[Table of Contents] [Description]

Path 1542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      ...
130. if (args == null)
```

[Table of Contents] [Description]

Path 1543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
```

```
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
    ...
147. if (response.equals(""))
```

[Table of Contents] [Description]

Path 1544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
    ...
147. if (response.equals(""))
```

[Table of Contents] [Description]

Path 1545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
    ...
149. if (!error.equals(""))
```

[Table of Contents] [Description]

Path 1546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
    ...
149. if (!error.equals(""))
```

[Table of Contents] [Description]

Path 1547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
    ...
152. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      ...
152. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      ...
155. if (bannerResult != null)
```

[Table of Contents] [Description]

Path 1550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      ...
155. if (bannerResult != null)
```

[Table of Contents] [Description]

Path 1551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      ...
159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
     ...
159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //gpplugin.java
     ...
256. if (0 < timeout && timeout < 1000)
```

[Table of Contents] [Description]

Path 1554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //gpplugin.java
     ...
276. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 1555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //gpplugin.java
     ...
276. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 1556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //httpplugin.java
     ...
242. if (rVal >= 99 && rVal <= maxRetCode )
```

[Table of Contents] [Description]

Path 1557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public void run() //icmpplugin.java
       ...
193. if (ping != null && ping.isTarget(pong.getAddress()))
```

[Table of Contents] [Description]

Path 1558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. public void run() //icmpplugin.java
       ...
193. if (ping != null && ping.isTarget(pong.getAddress()))
```

[Table of Contents] [Description]

Path 1559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
232. private boolean isPingable(InetAddress ipv4Addr, int retries, long
timeout) //icmpplugin.java
       ...
276. if (reply.isSignaled())
```

[Table of Contents] [Description]

Path 1560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. private static void probe(InetAddress target, List supports) //ifcollector.java
       ...
204. if (r)
```

[Table of Contents] [Description]

Path 1561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. private static void probe(InetAddress target, List supports) //ifcollector.java
       ...
204. if (r)
```

[Table of Contents] [Description]

Path 1562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
387. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
387. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |
      ...
      |
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |
      ...
      |
405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |
      ...
      |
405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |
      ...
      |
430. if (ifIndex == null)
```

[Table of Contents] [Description]

Path 1571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
      |
      ...
      |
460. if (subtarget == null || subtarget.equals(m_target) ||
m_previouslyProbed.contains(subtarget))
```

[Table of Contents] [Description]

Path 1572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
469. if (ifType == null)
```

[Table of Contents] [Description]

Path 1573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
525. if (subtarget == null || subtarget.equals(m_target))
```

[Table of Contents] [Description]

Path 1576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
567. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //ifcollector.java
       ...
567. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. private String toMacString(byte[] mac) //ifsmbcollector.java
       ...
132. if (i != 5)
```

[Table of Contents] [Description]

Path 1583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. private String toMacString(byte[] mac) //ifsmbcollector.java
       ...
132. if (i != 5)
```

[Table of Contents] [Description]

Path 1584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. String getNbtName() //ifsmbcollector.java
224. if (m_addr != null)
```

[Table of Contents] [Description]

Path 1585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
276. boolean containsCtrlChars(String nbName) //ifsmbcollector.java
       ...
279. if (bytes[i] < 20)
```

[Table of Contents] [Description]

Path 1586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
301. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
301. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
306. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
306. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
317. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
317. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
329. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
329. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
340. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
340. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
343. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
343. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
350. if (m_domain != null)
```

[Table of Contents] [Description]

Path 1599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
353. if (authentication == null)
```

[Table of Contents] [Description]

Path 1600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
356. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
356. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
362. if (authentication == null)
```

[Table of Contents] [Description]

Path 1603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
380. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
380. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
      ...
402. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
402. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
408. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
408. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
411. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. public void run() //ifsmbcollector.java
       ...
411. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
286. int getAdminStatus(int ifIndex) //ifsnmpcollector.java
       ...
300. if (ifStatus != null)
```

[Table of Contents] [Description]

Path 1620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
307. int getIfType(int ifIndex) //ifsnmpcollector.java
       ...
321. if (ifType != null)
```

[Table of Contents] [Description]

Path 1621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. int getIfIndex(InetAddress address) //ifsnmpcollector.java
       ...
337. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. int getIfIndex(InetAddress address) //ifsnmpcollector.java
     ...
337. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. int getIfIndex(InetAddress address) //ifsnmpcollector.java
     ...
358. if (ndx != null)
```

[Table of Contents] [Description]

Path 1624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. SnmpOctetString getIfName(int ifIndex) //ifsnmpcollector.java
     ...
383. if (snmpIfIndex != null)
```

[Table of Contents] [Description]

Path 1625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. SnmpOctetString getIfName(int ifIndex) //ifsnmpcollector.java
     ...
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. SnmpOctetString getIfName(int ifIndex) //ifsnmpcollector.java
     ...
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. SnmpOctetString getIfName(int ifIndex) //ifsnmpcollector.java
       ...
401. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. SnmpOctetString getIfName(int ifIndex) //ifsnmpcollector.java
       ...
401. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. SnmpOctetString getIfAlias(int ifIndex) //ifsnmpcollector.java
       ...
424. if (snmpIfIndex != null)
```

[Table of Contents] [Description]

Path 1630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. SnmpOctetString getIfAlias(int ifIndex) //ifsnmpcollector.java
       ...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. SnmpOctetString getIfAlias(int ifIndex) //ifsnmpcollector.java
       ...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. SnmpOctetString getIfAlias(int ifIndex) //ifsnmpcollector.java
       ...
442. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. SnmpOctetString getIfAlias(int ifIndex) //ifsnmpcollector.java
       ...
442. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
492. if (!this.hasSystemGroup())
```

[Table of Contents] [Description]

Path 1635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
492. if (!this.hasSystemGroup())
```

[Table of Contents] [Description]

Path 1636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
494. if (!this.hasIfTable())
```

[Table of Contents] [Description]

Path 1637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
494. if (!this.hasIfTable())
```

[Table of Contents] [Description]

Path 1638:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
496. if (!this.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1639:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
496. if (!this.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1640:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
519. if (session != null)
```

[Table of Contents] [Description]

Path 1641:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public void run() //ifsnmpcollector.java
       ...
519. if (session != null)
```

[Table of Contents] [Description]

Path 1642:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. private boolean isServer( String hostname, Map qualifiers ) //jdbcplugin.java
     ...
148. if ( status == true )
```

[Table of Contents] [Description]

Path 1643:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. private boolean isServer( String hostname, Map qualifiers ) //jdbcplugin.java
     ...
148. if ( status == true )
```

[Table of Contents] [Description]

Path 1644:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcplugin.java
     ...
210. if ( result != null )
```

[Table of Contents] [Description]

Path 1645:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. void update(Connection db) throws SQLException //knownipmgr.java
     ...
213. if (stmt != null)
```

[Table of Contents] [Description]

Path 1646:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. void update(Connection db) throws SQLException //knownipmgr.java
     ...
213. if (stmt != null)
```

[Table of Contents] [Description]

Path 1647:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
294. static synchronized void dataSourceSync() throws SQLException //knownipmgr.java
     ...
341. if (c != null)
```

[Table of Contents] [Description]

Path 1648:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
294. static synchronized void dataSourceSync() throws SQLException //knownipmgr.java
     ...
341. if (c != null)
```

[Table of Contents] [Description]

Path 1649:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
     ...
173. if (lc != null)
```

[Table of Contents] [Description]

Path 1650:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
     ...
173. if (lc != null)
```

[Table of Contents] [Description]

Path 1651:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
     ...
196. if (socket != null)
```

[Table of Contents] [Description]

Path 1652:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
      |...
      |196. if (socket != null)
```

[Table of Contents] [Description]

Path 1653:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. public boolean isProtocolSupported(InetAddress address) //ldapplugin.java
      |...
      |226. if (isServer(address, DEFAULT_PORTS[i], DEFAULT_RETRY, DEFAULT_TIMEOUT))
```

[Table of Contents] [Description]

Path 1654:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //msexchangeplugin.java
      |...
      |151. if (rc > -1)
```

[Table of Contents] [Description]

Path 1655:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private boolean[] isServer(InetAddress host, int[] ports, int retries, int
timeout) //msexchangeplugin.java
      |...
      |211. if (ports[POP3_INDEX] > 0)
```

[Table of Contents] [Description]

Path 1656:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. private boolean[] isServer(InetAddress host, int[] ports, int retries, int
timeout) //msexchangeplugin.java
```

```
...
214. if (ports[IMAP_INDEX] > 0)
```

[Table of Contents] [Description]

Path 1657:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
...
288. if (result[0] && !qualifiers.containsKey("pop3 port"))
```

[Table of Contents] [Description]

Path 1658:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
...
288. if (result[0] && !qualifiers.containsKey("pop3 port"))
```

[Table of Contents] [Description]

Path 1659:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
...
291. if (result[1] && !qualifiers.containsKey("imap port"))
```

[Table of Contents] [Description]

Path 1660:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
...
291. if (result[1] && !qualifiers.containsKey("imap port"))
```

[Table of Contents] [Description]

Path 1661:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
      |
      ...
      294. if (result[2] && !qualifiers.containsKey("mapi port"))
```

[Table of Contents] [Description]

Path 1662:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //msexchangeplugin.java
      |
      ...
      294. if (result[2] && !qualifiers.containsKey("mapi port"))
```

[Table of Contents] [Description]

Path 1663:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, String command, int
padding, int retries, int timeout, RE regex, StringBuffer
bannerResult) //nrpeplugin.java
      |
      ...
      204. if (socket != null)
```

[Table of Contents] [Description]

Path 1664:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, String command, int
padding, int retries, int timeout, RE regex, StringBuffer
bannerResult) //nrpeplugin.java
      |
      ...
      204. if (socket != null)
```

[Table of Contents] [Description]

Path 1665:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
258. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //nrpeplugin.java
      |
      ...
      293. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 1666:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
258. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //nrpeplugin.java
      |...
      |293. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 1667:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout) //ntpplugin.java
      |...
      |153. if (socket != null)
```

[Table of Contents] [Description]

Path 1668:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
97. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout) //ntpplugin.java
      |...
      |153. if (socket != null)
```

[Table of Contents] [Description]

Path 1669:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //ntpplugin.java
      |...
      |217. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1670:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public boolean isProtocolSupported(InetAddress address, Map
```

```
qualifiers) //ntpplugin.java
    ...
217. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1671:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
    ...
153. if (t.nextToken().equals("+OK"))
```

[Table of Contents] [Description]

Path 1672:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
    ...
180. if (socket != null)
```

[Table of Contents] [Description]

Path 1673:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
    ...
180. if (socket != null)
```

[Table of Contents] [Description]

Path 1674:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
227. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //pop3plugin.java
    ...
239. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1675:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
227. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //pop3plugin.java
     |...
     |239. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 1676:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. LightWeightNodeEntry(int nodeID, String netbiosName) //reparentviasmb.java
     |...
     |236. if (netbiosName != null)
```

[Table of Contents] [Description]

Path 1677:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. boolean hasHeavyWeightNodeEntry() //reparentviasmb.java
     |302. if (m_hwNodeEntry == null)
```

[Table of Contents] [Description]

Path 1678:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
316. public boolean equals(Object o) //reparentviasmb.java
     |...
     |319. if (m_netbiosName == null || node.getNetbiosName() == null)
```

[Table of Contents] [Description]

Path 1679:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
316. public boolean equals(Object o) //reparentviasmb.java
     |...
     |321. else if (node.getNetbiosName().equals(m_netbiosName))
```

[Table of Contents] [Description]

Path 1680:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
410. if (outerNetbiosName == null || m_netbiosNamesToSkip.contains
(outerNetbiosName))
```

[Table of Contents] [Description]

Path 1681:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
414. if (outerEntry.isDuplicate())
```

[Table of Contents] [Description]

Path 1682:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
427. if (innerEntry.getNodeId() <= outerEntry.getNodeId())
```

[Table of Contents] [Description]

Path 1683:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
431. if (innerNetbiosName == null || m_netbiosNamesToSkip.contains
(innerNetbiosName))
```

[Table of Contents] [Description]

Path 1684:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
435. if (innerEntry.isDuplicate())
```

[Table of Contents] [Description]

Path 1685:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
441. if (duplicateNodeList == null)
```

[Table of Contents] [Description]

Path 1686:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1687:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1688:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
455. if (m_reparentNodeMap == null)
```

[Table of Contents] [Description]

Path 1689:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
     ...
458. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1690:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. private void buildNodeLists() throws SQLException //reparentviasmb.java
       ...
458. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1691:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public void sync() throws SQLException //reparentviasmb.java
       ...
481. if (m_reparentedIfMap != null && !m_reparentedIfMap.isEmpty())
```

[Table of Contents] [Description]

Path 1692:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public void sync() throws SQLException //reparentviasmb.java
       ...
481. if (m_reparentedIfMap != null && !m_reparentedIfMap.isEmpty())
```

[Table of Contents] [Description]

Path 1693:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
       ...
533. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1694:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
       ...
533. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1695:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
     ...
668. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1696:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
     ...
668. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1697:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
     ...
562. if (reparentedIfList == null)
```

[Table of Contents] [Description]

Path 1698:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
     ...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1699:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
     ...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1700:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      ...
615. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1701:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      ...
615. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1702:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void reparentInterfaces() throws SQLException //reparentviasmb.java
      ...
635. if (m_reparentedIfMap == null)
```

[Table of Contents] [Description]

Path 1703:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
      ...
654. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1704:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
      ...
654. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1705:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
       ...
668. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1706:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
       ...
668. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1707:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
       ...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1708:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
       ...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1709:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
       ...
688. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1710:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
648. private void generateEvents() //reparentviasmb.java
      ...
688. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1711:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
708. private synchronized void sendInterfaceReparentedEvent(String ipAddr, String
ipHostName, int newNodeId, int oldNodeId, DbNodeEntry
reparentNodeEntry) //reparentviasmb.java
      ...
710. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1712:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
708. private synchronized void sendInterfaceReparentedEvent(String ipAddr, String
ipHostName, int newNodeId, int oldNodeId, DbNodeEntry
reparentNodeEntry) //reparentviasmb.java
      ...
710. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1713:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
708. private synchronized void sendInterfaceReparentedEvent(String ipAddr, String
ipHostName, int newNodeId, int oldNodeId, DbNodeEntry
reparentNodeEntry) //reparentviasmb.java
      ...
714. if (ipHostName == null)
```

[Table of Contents] [Description]

Path 1714:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. RescanProcessor(Scheduler.NodeInfo nodeInfo, boolean
forceRescan) //rescanprocessor.java
```

```
...
236. if (nodeInfo == null)
```

[Table of Contents] [Description]

Path 1715:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1716:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1717:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
...
308. if (primaryIfc.getSnmpCollector() != null)
```

[Table of Contents] [Description]

Path 1718:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
...
330. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1719:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
     ...
330. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1720:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
     ...
335. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1721:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
     ...
335. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1722:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java
     ...
340. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1723:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java

     ...
340. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1724:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java

     ...
345. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1725:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. private DbNodeEntry updateNode(Connection dbc, Date now, DbNodeEntry
dbNodeEntry, InetAddress currPrimarySnmpIf, DbIpInterfaceEntry[] dbIpInterfaces, Map
collectorMap) throws SQLException //rescanprocessor.java

     ...
345. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1726:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
427. private void updateInterfaces(Connection dbc, Date now, DbNodeEntry node, Map
collectorMap, boolean doesSnmp) throws SQLException //rescanprocessor.java

     ...
438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1727:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
427. private void updateInterfaces(Connection dbc, Date now, DbNodeEntry node, Map
collectorMap, boolean doesSnmp) throws SQLException //rescanprocessor.java

     ...
438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1728:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
561. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1729:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
561. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1730:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
566. if (snmpc == null || snmpc.failed())
```

[Table of Contents] [Description]

Path 1731:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
588. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1732:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
```

```
...
588. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1733:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
599. if (status != -1)
```

[Table of Contents] [Description]

Path 1734:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
599. if (status != -1)
```

[Table of Contents] [Description]

Path 1735:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
608. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1736:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
608. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1737:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
618. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1738:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
618. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1739:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
637. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1740:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
637. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1741:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
664. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1742:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
664. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1743:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
666. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1744:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
666. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1745:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
     ...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1746:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
```

```
...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1747:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
717. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 1748:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
717. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 1749:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
725. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 1750:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
558. private void updateNonIpInterface(Connection dbc, Date now, DbNodeEntry node,
int ifIndex, IfSnmpCollector snmpc) throws SQLException //rescanprocessor.java
...
725. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 1751:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
814. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1752:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
814. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1753:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
880. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1754:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
880. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1755:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
```

```
...
889. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1756:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
...
889. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1757:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
...
921. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1758:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
...
921. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1759:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
...
1083. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1760:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1086. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1761:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1086. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1762:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1113. if (cFactory.isAddressUnmanaged(ifaddr))
```

[Table of Contents] [Description]

Path 1763:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1113. if (cFactory.isAddressUnmanaged(ifaddr))
```

[Table of Contents] [Description]

Path 1764:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1118. if (ipPkg != null)
```

[Table of Contents] [Description]

Path 1765:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1121. if (ipToBePolled)
```

[Table of Contents] [Description]

Path 1766:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1121. if (ipToBePolled)
```

[Table of Contents] [Description]

Path 1767:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1121. if (ipToBePolled)
```

[Table of Contents] [Description]

Path 1768:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
```

```
1126. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1769:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1126. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1770:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1174. if (dbIpIfEntry.hasIfIndexChanged())
```

[Table of Contents] [Description]

Path 1771:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1177. if (dbIpIfEntry.hasHostnameChanged())
```

[Table of Contents] [Description]

Path 1772:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
    ...
1192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1773:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1774:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1201. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1775:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1201. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1776:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1213. if (ipPkg != null)
```

[Table of Contents] [Description]

Path 1777:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1284. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1778:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1284. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1779:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1296. if (dbSupportedServices[i].getServiceId() == sid.intValue())
```

[Table of Contents] [Description]

Path 1780:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1305. if (cFactory.isAddressUnmanaged(ifaddr))
```

[Table of Contents] [Description]

Path 1781:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
```

```
1305. if (cFactory.isAddressUnmanaged(ifaddr))
```

[Table of Contents] [Description]

Path 1782:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1313. if (!svcToBePolled)
```

[Table of Contents] [Description]

Path 1783:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
2117. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1784:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
2117. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1785:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1786:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1787:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1788:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1789:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1366. if (m_forceRescan)
```

[Table of Contents] [Description]

Path 1790:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
    ...
1366. if (m_forceRescan)
```

[Table of Contents] [Description]

Path 1791:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
    ...
1398. if (ipPkg != null)
```

[Table of Contents] [Description]

Path 1792:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
    ...
1401. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1793:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
    ...
1401. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1794:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
    ...
```

```
1426. if (ctest != null)
```

[Table of Contents] [Description]

Path 1795:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
...
1426. if (ctest != null)
```

[Table of Contents] [Description]

Path 1796:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
...
1438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1797:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
...
1438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1798:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
...
1449. if (ipPkg == null)
```

[Table of Contents] [Description]

Path 1799:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1462. if (!svcToBePolled)
```

[Table of Contents] [Description]

Path 1800:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1468. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1801:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1468. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1802:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1472. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1803:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1472. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1804:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1479. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1805:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1479. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1806:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1485. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1807:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
```

```
1485. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1808:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1495. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1809:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1495. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1810:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1811:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1812:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1511. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1813:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1511. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1814:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1517. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1815:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1517. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1816:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1817:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1818:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1819:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
1534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1820:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
      ...
```

```
1539. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1821:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1384. private void updateServicesOnForcedRescan(DbNodeEntry node, DbIpInterfaceEntry
dbIpIfEntry, DbIfServiceEntry[] dbSupportedServices) throws
SQLException //rescanprocessor.java
    ...
1539. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1822:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
    ...
1582. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1823:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
    ...
1582. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1824:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
    ...
1592. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1825:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1592. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1826:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1624. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1827:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1624. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1828:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1657. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1829:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1657. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1830:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1669. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1831:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1669. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1832:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1671. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1833:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
```

```
1671. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 1834:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1687. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1835:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1687. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1836:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1711. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 1837:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1711. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 1838:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1719. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 1839:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1719. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 1840:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1753. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1841:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1753. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1842:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1763. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1843:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1763. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1844:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1790. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1845:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
1790. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1846:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
      ...
```

```
1800. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1847:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1570. private void updateSnmpInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc) throws
SQLException //rescanprocessor.java
...
1800. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1848:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
...
1840. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1849:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
...
1840. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1850:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
...
1861. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1851:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
      ...
1861. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1852:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
      ...
1865. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1853:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1836. private boolean isDuplicateNode(Connection dbc, DbNodeEntry suspectNode,
IfSnmpCollector snmpc) //rescanprocessor.java
      ...
1865. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1854:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
      ...
1918. if (addrUnmanaged)
```

[Table of Contents] [Description]

Path 1855:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
```

```
...
1918. if (addrUnmanaged)
```

[Table of Contents] [Description]

Path 1856:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
    ...
1924. if (!svcToBePolled)
```

[Table of Contents] [Description]

Path 1857:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
    ...
1928. if (svcToBePolled)
```

[Table of Contents] [Description]

Path 1858:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
    ...
1928. if (svcToBePolled)
```

[Table of Contents] [Description]

Path 1859:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
    ...
```

```
1951. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 1860:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
      ...
1951. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 1861:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
1993. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1862:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
1993. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1863:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2004. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1864:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2004. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1865:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2018. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1866:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2018. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1867:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2033. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1868:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2033. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1869:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2052. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1870:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2052. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1871:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2067. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1872:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2067. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1873:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
```

```
...
2089. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1874:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
2089. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1875:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
2103. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1876:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
2103. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1877:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
...
2117. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1878:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2117. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1879:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2161. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1880:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2161. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1881:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2166. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1882:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2166. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1883:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      |...
      2187. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1884:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      |...
      2187. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1885:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      |...
      2205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1886:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2154. private static List buildLBSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      |...
      2205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1887:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      |
```

```
...
2237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1888:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1889:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2258. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1890:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2258. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1891:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2276. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1892:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2230. private static List buildSnmpAddressList(Map collectorMap, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2276. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1893:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2299. private InetAddress determinePrimaryIpInterface(Map
collectorMap) //rescanprocessor.java
      ...
2333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1894:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2423. if (smbc.getDomainName() != null)
```

[Table of Contents] [Description]

Path 1895:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2423. if (smbc.getDomainName() != null)
```

[Table of Contents] [Description]

Path 1896:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2426. if (smbc.getOS() != null)
```

[Table of Contents] [Description]

Path 1897:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2426. if (smbc.getOS() != null)
```

[Table of Contents] [Description]

Path 1898:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2458. if (savedSmbcRef.getDomainName() != null)
```

[Table of Contents] [Description]

Path 1899:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2458. if (savedSmbcRef.getDomainName() != null)
```

[Table of Contents] [Description]

Path 1900:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
      ...
2461. if (savedSmbcRef.getOS() != null)
```

[Table of Contents] [Description]

Path 1901:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
```

```
...
2461. if (savedSmbcRef.getOS() != null)
```

[Table of Contents] [Description]

Path 1902:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2357. private void setNodeLabelAndSmbInfo(Map collectorMap, DbNodeEntry dbNodeEntry,
DbNodeEntry currNodeEntry, InetAddress currPrimarySnmpIf) //rescanprocessor.java
...
2483. if (ifc.getSnmpCollector() != null)
```

[Table of Contents] [Description]

Path 1903:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2530. private boolean isNodeDeleted(Connection dbc, int nodeId) throws
SQLException //rescanprocessor.java
...
2544. if (nodeType == DbNodeEntry.NODE_TYPE_DELETED)
```

[Table of Contents] [Description]

Path 1904:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2576. if (ipAddr == null || snmpc == null)
```

[Table of Contents] [Description]

Path 1905:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
...
2580. if (!snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1906:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2587. if (snmpc.hasIfTable())
```

[Table of Contents] [Description]

Path 1907:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2590. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1908:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2593. if (ifTable == null || ipAddrTable == null)
```

[Table of Contents] [Description]

Path 1909:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2605. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1910:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2605. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1911:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2617. if (snmpIfIndex != null)
```

[Table of Contents] [Description]

Path 1912:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2572. private boolean isInterfaceAlias(InetAddress ipAddr, IfSnmpCollector
snmpc) //rescanprocessor.java
      ...
2636. if (isAlias == false)
```

[Table of Contents] [Description]

Path 1913:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2657. private boolean areDbInterfacesInSnmpCollection(DbIpInterfaceEntry[]
dbInterfaces, IfSnmpCollector snmpc) //rescanprocessor.java
      ...
2675. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 1914:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2657. private boolean areDbInterfacesInSnmpCollection(DbIpInterfaceEntry[]
dbInterfaces, IfSnmpCollector snmpc) //rescanprocessor.java
      ...
2706. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1915:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2657. private boolean areDbInterfacesInSnmpCollection(DbIpInterfaceEntry[]
dbInterfaces, IfSnmpCollector snmpc) //rescanprocessor.java
```

```
...
2706. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1916:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2657. private boolean areDbInterfacesInSnmpCollection(DbIpInterfaceEntry[]
dbInterfaces, IfSnmpCollector snmpc) //rescanprocessor.java
...
2712. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1917:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2657. private boolean areDbInterfacesInSnmpCollection(DbIpInterfaceEntry[]
dbInterfaces, IfSnmpCollector snmpc) //rescanprocessor.java
...
2712. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1918:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
...
2731. if (dbNodeEntry == null) return;
```

[Table of Contents] [Description]

Path 1919:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
...
2733. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1920:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      |2733. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1921:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      |2738. if (dbInterfaces == null) return;
```

[Table of Contents] [Description]

Path 1922:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      |2771. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1923:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      |2771. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1924:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      |...
      |2778. if (snmpc != null)
```

[Table of Contents] [Description]

Path 1925:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2819. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1926:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2819. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1927:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2823. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1928:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2823. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1929:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2836. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1930:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2836. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1931:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2840. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1932:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2840. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1933:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2847. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1934:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2847. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1935:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2946. if (dbc != null)
```

[Table of Contents] [Description]

Path 1936:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2946. if (dbc != null)
```

[Table of Contents] [Description]

Path 1937:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2974. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1938:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2724. public void run() //rescanprocessor.java
      ...
2974. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1939:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2984. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1940:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2984. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1941:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2990. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1942:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2990. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1943:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2995. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1944:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2995. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1945:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2998. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1946:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
2998. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1947:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
3002. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1948:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2978. private boolean scanPrimarySnmpInterface(Category log, DbIpInterfaceEntry
oldPrimarySnmpInterface, Map collectorMap, Set probedAddrs) //rescanprocessor.java
      ...
3002. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1949:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3009. private InetAddress updatePrimarySnmpInterface(Connection dbc, DbNodeEntry
dbNodeEntry, Map collectorMap, InetAddress oldPriIf) throws
SQLException //rescanprocessor.java
```

```
...
3087. if(log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1950:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3009. private InetAddress updatePrimarySnmpInterface(Connection dbc, DbNodeEntry
dbNodeEntry, Map collectorMap, InetAddress oldPriIf) throws
SQLException //rescanprocessor.java
...
3087. if(log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1951:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3113. private DbIpInterfaceEntry[] getInterfaces(DbNodeEntry
dbNodeEntry) //rescanprocessor.java
...
3125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1952:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3113. private DbIpInterfaceEntry[] getInterfaces(DbNodeEntry
dbNodeEntry) //rescanprocessor.java
...
3125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1953:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPriIf, InetAddress primarySnmpIf) //rescanprocessor.java
...
3179. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1954:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPriIf, InetAddress primarySnmpIf) //rescanprocessor.java
      ...
3179. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1955:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPriIf, InetAddress primarySnmpIf) //rescanprocessor.java
      ...
3205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1956:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3168. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPriIf, InetAddress primarySnmpIf) //rescanprocessor.java
      ...
3205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1957:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3220. private void createNodeLabelChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1958:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3220. private void createNodeLabelChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1959:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3220. private void createNodeLabelChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3285. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1960:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3220. private void createNodeLabelChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3285. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1961:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3298. private void createNodeInfoChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3301. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1962:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3298. private void createNodeInfoChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
      ...
3301. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1963:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3298. private void createNodeInfoChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
```

```
...
3413. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1964:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3298. private void createNodeInfoChangedEvent(DbNodeEntry updatedEntry, DbNodeEntry
originalEntry) //rescanprocessor.java
...
3413. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1965:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
...
3432. if (oldPrimaryIf != null)
```

[Table of Contents] [Description]

Path 1966:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
...
3436. if (newPrimaryIf != null)
```

[Table of Contents] [Description]

Path 1967:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
...
3439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1968:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
      ...
3439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1969:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
      ...
3489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1970:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3428. private void createPrimarySnmpInterfaceChangedEvent(int nodeId, InetAddress
newPrimaryIf, InetAddress oldPrimaryIf) //rescanprocessor.java
      ...
3489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1971:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3502. private void createInterfaceIndexChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3505. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1972:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3502. private void createInterfaceIndexChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3505. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1973:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3502. private void createInterfaceIndexChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1974:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3502. private void createInterfaceIndexChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1975:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3562. private void createIpHostNameChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3565. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1976:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3562. private void createIpHostNameChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
      ...
3565. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1977:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3562. private void createIpHostNameChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
```

```
...
3613. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1978:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3562. private void createIpHostNameChangedEvent(DbIpInterfaceEntry updatedEntry,
DbIpInterfaceEntry originalEntry) //rescanprocessor.java
...
3613. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1979:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3628. private void createInterfaceReparentedEvent(DbNodeEntry newNode, int
oldNodeId, InetAddress reparentedIf) //rescanprocessor.java
...
3631. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1980:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3628. private void createInterfaceReparentedEvent(DbNodeEntry newNode, int
oldNodeId, InetAddress reparentedIf) //rescanprocessor.java
...
3631. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1981:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3628. private void createInterfaceReparentedEvent(DbNodeEntry newNode, int
oldNodeId, InetAddress reparentedIf) //rescanprocessor.java
...
3722. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1982:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3628. private void createInterfaceReparentedEvent(DbNodeEntry newNode, int
oldNodeId, InetAddress reparentedIf) //rescanprocessor.java
      ...
3722. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1983:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3733. private void createDuplicateNodeDeletedEvent(DbNodeEntry
deletedNode) //rescanprocessor.java
      ...
3736. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1984:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3733. private void createDuplicateNodeDeletedEvent(DbNodeEntry
deletedNode) //rescanprocessor.java
      ...
3736. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1985:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3733. private void createDuplicateNodeDeletedEvent(DbNodeEntry
deletedNode) //rescanprocessor.java
      ...
3754. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1986:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3733. private void createDuplicateNodeDeletedEvent(DbNodeEntry
deletedNode) //rescanprocessor.java
      ...
3754. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1987:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3765. private void createNodeGainedInterfaceEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
      ...
3768. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1988:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3765. private void createNodeGainedInterfaceEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
      ...
3768. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1989:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3765. private void createNodeGainedInterfaceEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
      ...
3792. if (ifEntry.getHostname() == null)
```

[Table of Contents] [Description]

Path 1990:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3765. private void createNodeGainedInterfaceEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
      ...
3818. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1991:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3765. private void createNodeGainedInterfaceEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
```

```
...
3818. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1992:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3829. private void createDuplicateIpAddressEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
3832. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1993:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3829. private void createDuplicateIpAddressEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
3832. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1994:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3829. private void createDuplicateIpAddressEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
3856. if (ifEntry.getHostname() == null)
```

[Table of Contents] [Description]

Path 1995:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3829. private void createDuplicateIpAddressEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
3882. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1996:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3829. private void createDuplicateIpAddressEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
      ...
      3882. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1997:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3897. private void createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
      3900. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1998:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3897. private void createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
      3900. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 1999:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3897. private void createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
      3926. if (ifEntry.getHostname() == null)
```

[Table of Contents] [Description]

Path 2000:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3897. private void createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
      3964. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2001:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3897. private void createNodeGainedServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
3964. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2002:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3979. private void createSuspendPollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
4005. if (ifEntry.getHostname() == null)
```

[Table of Contents] [Description]

Path 2003:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3979. private void createSuspendPollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
4043. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2004:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
3979. private void createSuspendPollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
      ...
4043. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2005:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4058. private void createResumePollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
```

```
...
4084. if (ifEntry.getHostname() == null)
```

[Table of Contents] [Description]

Path 2006:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4058. private void createResumePollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
...
4122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2007:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4058. private void createResumePollingServiceEvent(DbNodeEntry nodeEntry,
DbIpInterfaceEntry ifEntry, String svcName) //rescanprocessor.java
...
4122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2008:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4206. private void createInterfaceSupportsSNMPEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
4209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2009:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4206. private void createInterfaceSupportsSNMPEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
...
4209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2010:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4206. private void createInterfaceSupportsSNMPEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
    ...
4229. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2011:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4206. private void createInterfaceSupportsSNMPEvent(DbIpInterfaceEntry
ifEntry) //rescanprocessor.java
    ...
4229. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2012:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4242. private void createReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //rescanprocessor.java
    ...
4245. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2013:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4242. private void createReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //rescanprocessor.java
    ...
4245. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2014:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4242. private void createReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //rescanprocessor.java
    ...
4265. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2015:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
4242. private void createReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //rescanprocessor.java
      ...
      4265. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2016:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. boolean timeForRescan() //scheduler.java
      184. if (System.currentTimeMillis() >= (m_lastScanned.getTime() + m_interval))
```

[Table of Contents] [Description]

Path 2017:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
      ...
      208. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2018:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
      ...
      208. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2019:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
      ...
      214. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2020:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
     ...
214. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2021:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
     ...
224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2022:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
     ...
224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2023:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
     ...
232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2024:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
     ...
232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2025:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2026:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2027:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
274. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2028:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
274. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2029:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2030:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2031:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
286. if (nodeStmt != null)
```

[Table of Contents] [Description]

Path 2032:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
286. if (nodeStmt != null)
```

[Table of Contents] [Description]

Path 2033:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
292. if (ifStmt != null)
```

[Table of Contents] [Description]

Path 2034:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
     ...
292. if (ifStmt != null)
```

[Table of Contents] [Description]

Path 2035:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
       ...
298. if (db != null)
```

[Table of Contents] [Description]

Path 2036:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
       ...
298. if (db != null)
```

[Table of Contents] [Description]

Path 2037:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
316. void scheduleNode(int nodeId) throws SQLException //scheduler.java
       ...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2038:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
316. void scheduleNode(int nodeId) throws SQLException //scheduler.java
       ...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2039:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
390. public synchronized void start() //scheduler.java
       391. if (m_worker != null)
```

[Table of Contents] [Description]

Path 2040:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
390. public synchronized void start() //scheduler.java
      ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2041:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
390. public synchronized void start() //scheduler.java
      ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2042:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
411. public synchronized void stop() //scheduler.java
      412. if (m_worker == null)
```

[Table of Contents] [Description]

Path 2043:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public synchronized void pause() //scheduler.java
      432. if (m_worker == null)
```

[Table of Contents] [Description]

Path 2044:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public synchronized void pause() //scheduler.java
      ...
435. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 2045:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public synchronized void pause() //scheduler.java
      |...
      |438. if (m_status == PAUSED)
```

[Table of Contents] [Description]

Path 2046:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public synchronized void resume() //scheduler.java
      |454. if (m_worker == null)
```

[Table of Contents] [Description]

Path 2047:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public synchronized void resume() //scheduler.java
      |...
      |457. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 2048:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public synchronized void resume() //scheduler.java
      |...
      |460. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 2049:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public synchronized int getStatus() //scheduler.java
      |473. if (m_worker != null && m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 2050:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
      |
```

```
...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2051:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2052:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
511. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2053:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
511. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2054:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
525. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2055:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
```

```
...
525. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2056:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
529. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2057:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
529. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2058:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
543. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2059:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
543. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2060:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
```

```
...
550. if (node.isScheduled())
```

[Table of Contents] [Description]

Path 2061:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
554. if (!node.timeForRescan())
```

[Table of Contents] [Description]

Path 2062:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2063:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2064:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
598. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2065:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
```

```
...
598. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2066:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
605. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2067:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
492. public void run() //scheduler.java
...
605. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2068:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. private boolean isSmb(InetAddress host) //smbplugin.java
...
85. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2069:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. private boolean isSmb(InetAddress host) //smbplugin.java
...
85. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2070:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. private boolean isSmb(InetAddress host) //smbplugin.java
```

```
...
91. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2071:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. private boolean isSmb(InetAddress host) //smbplugin.java
...
91. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2072:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. static String getOsLabel(String nativeOS, NbtAddress[] addresses, boolean
isSamba, boolean hasExchange) //smbutils.java
...
234. if (hasExchange || isNTServer(addresses))
```

[Table of Contents] [Description]

Path 2073:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. static String getOsLabel(String nativeOS, NbtAddress[] addresses, boolean
isSamba, boolean hasExchange) //smbutils.java
...
241. if (hasExchange || isNTServer(addresses))
```

[Table of Contents] [Description]

Path 2074:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static String getSmbURL(SmbAuth smbAuth, String cname) //smbutils.java
...
366. if (smbAuth.getType().equalsIgnoreCase("domain"))
```

[Table of Contents] [Description]

Path 2075:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static String getSmbURL(SmbAuth smbAuth, String cname) //smbutils.java
       ...
372. if (domainParm != null)
```

[Table of Contents] [Description]

Path 2076:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static String getSmbURL(SmbAuth smbAuth, String cname) //smbutils.java
       ...
375. if (useridParm != null)
```

[Table of Contents] [Description]

Path 2077:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static String getSmbURL(SmbAuth smbAuth, String cname) //smbutils.java
       ...
378. if (passwordParm != null)
```

[Table of Contents] [Description]

Path 2078:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
346. static String getSmbURL(SmbAuth smbAuth, String cname) //smbutils.java
       ...
381. if (useridParm != null)
```

[Table of Contents] [Description]

Path 2079:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
       ...
257. if (rc == 221)
```

[Table of Contents] [Description]

Path 2080:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      |
      ...
      286. if (socket != null)
```

[Table of Contents] [Description]

Path 2081:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      |
      ...
      286. if (socket != null)
```

[Table of Contents] [Description]

Path 2082:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //smtpplugin.java
      |
      ...
      349. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 2083:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //smtpplugin.java
      |
      ...
      349. if (result && qualifiers != null && !qualifiers.containsKey("port"))
```

[Table of Contents] [Description]

Path 2084:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public boolean isProtocolSupported(InetAddress address) //snmpplugin.java
      |
      ...
      101. if (peer == null)
```

[Table of Contents] [Description]

Path 2085:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public boolean isProtocolSupported(InetAddress address) //snmpplugin.java
      |...
      123. if (session != null)
```

[Table of Contents] [Description]

Path 2086:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public boolean isProtocolSupported(InetAddress address) //snmpplugin.java
      |...
      123. if (session != null)
```

[Table of Contents] [Description]

Path 2087:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
      |...
      149. if (peer == null)
```

[Table of Contents] [Description]

Path 2088:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
      |...
      179. if (version.equalsIgnoreCase("snmpv1"))
```

[Table of Contents] [Description]

Path 2089:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
      |...
      179. if (version.equalsIgnoreCase("snmpv1"))
```

[Table of Contents] [Description]

Path 2090:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |...
     181. else if (version.equalsIgnoreCase("snmpv2"))
```

[Table of Contents] [Description]

Path 2091:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |...
     181. else if (version.equalsIgnoreCase("snmpv2"))
```

[Table of Contents] [Description]

Path 2092:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |...
     225. if (retrievedValue.equals(expectedValue))
```

[Table of Contents] [Description]

Path 2093:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |...
     232. if (peer.getParameters().getVersion() == SnmpSMI.SNMPV1)
```

[Table of Contents] [Description]

Path 2094:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
     |
```

```
...
232. if (peer.getParameters().getVersion() == SnmpSMI.SNMPV1)
```

[Table of Contents] [Description]

Path 2095:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
...
234. else if (peer.getParameters().getVersion() == SnmpSMI.SNMPV2)
```

[Table of Contents] [Description]

Path 2096:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
...
234. else if (peer.getParameters().getVersion() == SnmpSMI.SNMPV2)
```

[Table of Contents] [Description]

Path 2097:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
...
241. if (session != null)
```

[Table of Contents] [Description]

Path 2098:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //snmpplugin.java
...
241. if (session != null)
```

[Table of Contents] [Description]

Path 2099:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     ...
150. if (bannerResult != null)
```

[Table of Contents] [Description]

Path 2102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     ...
150. if (bannerResult != null)
```

[Table of Contents] [Description]

Path 2103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
     ...
168. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      |
      ...
      168. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      |
      ...
      198. if (socket != null)
```

[Table of Contents] [Description]

Path 2106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      |
      ...
      198. if (socket != null)
```

[Table of Contents] [Description]

Path 2107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //sshplugin.java
      |
      ...
      270. if (port == -1)
```

[Table of Contents] [Description]

Path 2108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //sshplugin.java
      |
```

```
...
288. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 2109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //sshplugin.java
...
288. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 2110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. SuspectEventProcessor(String ifAddress) //suspecteventprocessor.java
...
134. if (ifAddress == null)
```

[Table of Contents] [Description]

Path 2111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
166. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
166. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
171. if (!collector.hasSnmpCollection() || collector.getSnmpCollector().failed
())
```

[Table of Contents] [Description]

Path 2114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
179. if (snmpc.hasIfTable())
```

[Table of Contents] [Description]

Path 2115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
182. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 2116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
185. if (ifTable == null || ipAddrTable == null)
```

[Table of Contents] [Description]

Path 2117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
211. if (snmpIfIndex != null)
```

[Table of Contents] [Description]

Path 2118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
225. if (snmpIfType != null)
```

[Table of Contents] [Description]

Path 2121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
237. if (ipAddress == null || ipAddress.getHostAddress().equals("0.0.0.0") ||
ipAddress.getHostAddress().startsWith("127."))
```

[Table of Contents] [Description]

Path 2122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
```

```
collector) throws SQLException //suspecteventprocessor.java
    ...
252. if (firstAddress)
```

[Table of Contents] [Description]

Path 2123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
    ...
271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java

     ...
280. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java

     ...
280. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java

     ...
285. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java

     ...
285. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java

     ...
289. if (nodeID == -1)
```

[Table of Contents] [Description]

Path 2132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
299. if (!ipaddr.equals("0.0.0.0"))
```

[Table of Contents] [Description]

Path 2133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
299. if (!ipaddr.equals("0.0.0.0"))
```

[Table of Contents] [Description]

Path 2134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
314. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
314. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. private String getDuplicateIpaddress(List ipListA, List
ipListB) //suspecteventprocessor.java
```

```
...
339. if (ipListA == null || ipListB == null)
```

[Table of Contents] [Description]

Path 2137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. private String getDuplicateIpaddress(List ipListA, List
ipListB) //suspecteventprocessor.java
...
347. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. private String getDuplicateIpaddress(List ipListA, List
ipListB) //suspecteventprocessor.java
...
347. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
397. if (entryNode.getLabel().equals(primaryIf.getHostAddress()))
```

[Table of Contents] [Description]

Path 2140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
397. if (entryNode.getLabel().equals(primaryIf.getHostAddress()))
```

[Table of Contents] [Description]

Path 2141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     |...
     397. if (entryNode.getLabel().equals(primaryIf.getHostAddress()))
```

[Table of Contents] [Description]

Path 2142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     |...
     408. if (sysObjectId != null)
```

[Table of Contents] [Description]

Path 2143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     |...
     408. if (sysObjectId != null)
```

[Table of Contents] [Description]

Path 2144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     |...
     408. if (sysObjectId != null)
```

[Table of Contents] [Description]

Path 2145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     |...
     415. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
415. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
434. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
```

```
...
434. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
441. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
...
441. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
446. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
446. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
448. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. private DbNodeEntry createNode(Connection dbc, InetAddress ifaddr, IfCollector
collector) throws SQLException //suspecteventprocessor.java
     ...
448. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
552. if (ipPkg != null)
```

[Table of Contents] [Description]

Path 2160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
578. if (status != -1)
```

[Table of Contents] [Description]

Path 2161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
578. if (status != -1)
```

[Table of Contents] [Description]

Path 2162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
586. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
586. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
589. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
603. if (useExistingNode == true)
```

[Table of Contents] [Description]

Path 2166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
618. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
618. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
```

```
...
639. if (xaddrUnmanaged)
```

[Table of Contents] [Description]

Path 2169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
639. if (xaddrUnmanaged)
```

[Table of Contents] [Description]

Path 2170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
639. if (xaddrUnmanaged)
```

[Table of Contents] [Description]

Path 2171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
654. if (status != -1)
```

[Table of Contents] [Description]

Path 2172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
654. if (status != -1)
```

[Table of Contents] [Description]

Path 2173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
678. if (xipPkg != null)
```

[Table of Contents] [Description]

Path 2174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
710. if (status != -1)
```

[Table of Contents] [Description]

Path 2175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
710. if (status != -1)
```

[Table of Contents] [Description]

Path 2176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
732. if (sint != null)
```

[Table of Contents] [Description]

Path 2177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
745. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 2178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
766. if (aaddrs[0].getHostAddress().startsWith("127."))
```

[Table of Contents] [Description]

Path 2179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
773. if(aaddrs[0].equals(ifaddr))
```

[Table of Contents] [Description]

Path 2180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
777. if (aaddrs[1] != null)
```

[Table of Contents] [Description]

Path 2181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
```

```
777. if (aaddrs[1] != null)
```

[Table of Contents] [Description]

Path 2182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
782. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
782. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
784. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
...
784. if (str != null && str.length() > 0)
```

[Table of Contents] [Description]

Path 2186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
800. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
800. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
839. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 2189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
839. if (ifName != null && ifName.length() > 0)
```

[Table of Contents] [Description]

Path 2190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
847. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 2191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private void addInterfaces(Connection dbc, DbNodeEntry node, boolean
useExistingNode, InetAddress ifaddr, IfCollector collector) throws
SQLException //suspecteventprocessor.java
     ...
847. if (ifAlias != null && ifAlias.length() > 0)
```

[Table of Contents] [Description]

Path 2192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
     ...
908. if (addrUnmanaged)
```

[Table of Contents] [Description]

Path 2193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
     ...
908. if (addrUnmanaged)
```

[Table of Contents] [Description]

Path 2194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
```

```
SQLException //suspecteventprocessor.java
      ...
      914. if (!svcToBePolled)
```

[Table of Contents] [Description]

Path 2195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
      ...
      918. if (svcToBePolled)
```

[Table of Contents] [Description]

Path 2196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
      ...
      918. if (svcToBePolled)
```

[Table of Contents] [Description]

Path 2197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
      ...
      918. if (svcToBePolled)
```

[Table of Contents] [Description]

Path 2198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
      |
```

```
...
938. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
...
938. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. static boolean supportsSnmp(List
supportedProtocols) //suspecteventprocessor.java
...
957. if (p.getProtocolName().equals("SNMP"))
```

[Table of Contents] [Description]

Path 2201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. static boolean supportsSnmp(List
supportedProtocols) //suspecteventprocessor.java
...
959. else if (p.getProtocolName().equals("SNMPv1"))
```

[Table of Contents] [Description]

Path 2202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
953. static boolean supportsSnmp(List
supportedProtocols) //suspecteventprocessor.java
...
961. else if (p.getProtocolName().equals("SNMPv2"))
```

[Table of Contents] [Description]

Path 2203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
979. static boolean hasIfIndex(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      |...
      |981. if (snmpc.hasIpAddrTable())
```

[Table of Contents] [Description]

Path 2204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
979. static boolean hasIfIndex(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      |...
      |985. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
979. static boolean hasIfIndex(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      |...
      |985. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
979. static boolean hasIfIndex(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      |...
      |987. if (ifIndex == -1)
```

[Table of Contents] [Description]

Path 2207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1005. static int getIfType(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      |...
      |1010. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1005. static int getIfType(InetAddress ipaddr, IfSnmpCollector
snmpc) //suspecteventprocessor.java
      ...
1010. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1035. static InetAddress compareAndSelectPrimary(InetAddress currentIf, InetAddress
oldPrimary, String method) //suspecteventprocessor.java
      ...
1038. if (!CapsdConfigFactory.getInstance().isAddressUnmanaged(currentIf))
```

[Table of Contents] [Description]

Path 2210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1035. static InetAddress compareAndSelectPrimary(InetAddress currentIf, InetAddress
oldPrimary, String method) //suspecteventprocessor.java
      ...
1052. if (!CapsdConfigFactory.getInstance().isAddressUnmanaged(currentIf))
```

[Table of Contents] [Description]

Path 2211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1035. static InetAddress compareAndSelectPrimary(InetAddress currentIf, InetAddress
oldPrimary, String method) //suspecteventprocessor.java
      ...
1060. if (!CapsdConfigFactory.getInstance().isAddressUnmanaged(currentIf))
```

[Table of Contents] [Description]

Path 2212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1035. static InetAddress compareAndSelectPrimary(InetAddress currentIf, InetAddress
oldPrimary, String method) //suspecteventprocessor.java
```

```
...
1065. if (newPrimary != null)
```

[Table of Contents] [Description]

Path 2213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
...
1091. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
...
1091. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
...
1114. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
...
1114. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      ...
1130. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1084. private static List buildLBSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      ...
1130. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      ...
1156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      ...
1156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      ...
1179. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      |...
      1179. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      |...
      1195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1149. private static List buildSnmpAddressList(IfCollector
collector) //suspecteventprocessor.java
      |...
      1195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1215. private InetAddress determinePrimaryInterface(IfCollector
collector) //suspecteventprocessor.java
      |...
      1238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
      |...
```

```
1266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
      ...
1266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
      ...
1355. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
      ...
1355. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
      ...
1411. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
```

```
...
1411. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
...
1427. if (!useExistingNode)
```

[Table of Contents] [Description]

Path 2233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
...
1427. if (!useExistingNode)
```

[Table of Contents] [Description]

Path 2234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
...
1438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
...
1438. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
```

```
throws SQLException //suspecteventprocessor.java
    ...
1490. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
throws SQLException //suspecteventprocessor.java
    ...
1490. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
throws SQLException //suspecteventprocessor.java
    ...
1492. if (stmt != null) rs.close();
```

[Table of Contents] [Description]

Path 2239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1456. static List getPrimarySnmpInterfaceFromDb(Connection dbc, DbNodeEntry node)
throws SQLException //suspecteventprocessor.java
    ...
1492. if (stmt != null) rs.close();
```

[Table of Contents] [Description]

Path 2240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
    ...
1522. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1522. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
     ...
1534. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
     ...
1549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
     ...
1549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
     ...
1554. if(stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
     ...
```

```
1554. if(stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
...
1563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
...
1563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
...
1578. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
...
1578. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1583. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1518. static void setPrimarySnmpInterface(Connection dbc, DbNodeEntry node,
InetAddress newPrimarySnmpIf, InetAddress oldPrimarySnmpIf) throws
SQLException //suspecteventprocessor.java
      ...
1583. if (stmt != null) stmt.close();
```

[Table of Contents] [Description]

Path 2256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1601. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPrimary, InetAddress newPrimary) //suspecteventprocessor.java
      ...
1620. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1601. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPrimary, InetAddress newPrimary) //suspecteventprocessor.java
      ...
1620. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1601. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPrimary, InetAddress newPrimary) //suspecteventprocessor.java
```

```
...
1645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1601. private void generateSnmpDataCollectionEvents(DbNodeEntry nodeEntry,
InetAddress oldPrimary, InetAddress newPrimary) //suspecteventprocessor.java
...
1645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
...
1666. if (oldPrimaryIf != null)
```

[Table of Contents] [Description]

Path 2261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
...
1670. if (newPrimaryIf != null)
```

[Table of Contents] [Description]

Path 2262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
...
1673. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
      ...
1673. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
      ...
1724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1662. private void createAndSendPrimarySnmpInterfaceChangedEvent(int nodeId,
InetAddress newPrimaryIf, InetAddress oldPrimaryIf) //suspecteventprocessor.java
      ...
1724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1740. private void createAndSendReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //suspecteventprocessor.java
      ...
1743. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1740. private void createAndSendReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //suspecteventprocessor.java
      ...
1743. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1740. private void createAndSendReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //suspecteventprocessor.java
      |...
      1764. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1740. private void createAndSendReinitializePrimarySnmpInterfaceEvent(int nodeId,
InetAddress primarySnmpIf) //suspecteventprocessor.java
      |...
      1764. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1784. private void sendInterfaceEvents(DbNodeEntry node, boolean useExistingNode,
InetAddress ifaddr, IfCollector collector) //suspecteventprocessor.java
      |...
      1792. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1784. private void sendInterfaceEvents(DbNodeEntry node, boolean useExistingNode,
InetAddress ifaddr, IfCollector collector) //suspecteventprocessor.java
      |...
      1792. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1784. private void sendInterfaceEvents(DbNodeEntry node, boolean useExistingNode,
InetAddress ifaddr, IfCollector collector) //suspecteventprocessor.java
      |
```

```
...
1803. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1784. private void sendInterfaceEvents(DbNodeEntry node, boolean useExistingNode,
InetAddress ifaddr, IfCollector collector) //suspecteventprocessor.java
...
1803. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1784. private void sendInterfaceEvents(DbNodeEntry node, boolean useExistingNode,
InetAddress ifaddr, IfCollector collector) //suspecteventprocessor.java
...
1811. if (useExistingNode)
```

[Table of Contents] [Description]

Path 2275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1848. private void createAndSendNodeAddedEvent(DbNodeEntry
nodeEntry) //suspecteventprocessor.java
...
1901. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1848. private void createAndSendNodeAddedEvent(DbNodeEntry
nodeEntry) //suspecteventprocessor.java
...
1901. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1918. private void createAndSendDuplicateIpaddressEvent(int nodeId, String
ipAddr) //suspecteventprocessor.java
      ...
1965. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1918. private void createAndSendDuplicateIpaddressEvent(int nodeId, String
ipAddr) //suspecteventprocessor.java
      ...
1965. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1981. private void createAndSendNodeGainedInterfaceEvent(int nodeId, InetAddress
ipAddr) //suspecteventprocessor.java
      ...
2027. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1981. private void createAndSendNodeGainedInterfaceEvent(int nodeId, InetAddress
ipAddr) //suspecteventprocessor.java
      ...
2027. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2048. private void createAndSendNodeGainedServiceEvent(DbNodeEntry nodeEntry,
InetAddress ipAddr, String svcName, String qualifier) //suspecteventprocessor.java
      ...
2135. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
2048. private void createAndSendNodeGainedServiceEvent(DbNodeEntry nodeEntry,
InetAddress ipAddr, String svcName, String qualifier) //suspecteventprocessor.java
      ...
2135. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
150. if (bannerResult != null)
```

[Table of Contents] [Description]

Path 2286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
```

```
      ...
150.  if (bannerResult != null)
```

[Table of Contents] [Description]

Path 2287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
156.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
156.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
186.  if (socket != null)
```

[Table of Contents] [Description]

Path 2290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      ...
186.  if (socket != null)
```

[Table of Contents] [Description]

Path 2291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //tcpplugin.java
      |...
      258. if (port == -1)
```

[Table of Contents] [Description]

Path 2292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //tcpplugin.java
      |...
      276. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 2293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //tcpplugin.java
      |...
      276. if (bannerResult != null && bannerResult.length() > 0)
```

[Table of Contents] [Description]

Path 2294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      |342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      |342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      |350. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      |350. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |
```

```
      ...
387.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      ...
387.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      ...
404.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      ...
404.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      ...
412.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      412. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      417. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      417. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      482. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //iftable.java
      |...
      482. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |325. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |325. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |
```

```
      ...
366.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      ...
366.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      ...
384.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      ...
384.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      ...
392.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |392. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |397. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |397. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |436. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |436. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |492. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ifxtable.java
      |...
      |492. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
247. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ipaddrtable.java
      |...
      |280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
247. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ipaddrtable.java
      |...
      |280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
247. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ipaddrtable.java
      |
```

```
      ...
351.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
247.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //ipaddrtable.java
      ...
351.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
375.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //systemgroup.java
      ...
393.  if (errIndex == 1)
```

[Table of Contents] [Description]

Path 2331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302.  private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
      ...
322.  if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302.  private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
      ...
322.  if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
      ...
333. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
      ...
333. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
352. private void nodeGainedServiceHandler(Event
event) //broadcasteventprocessor.java
      ...
357. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
352. private void nodeGainedServiceHandler(Event
event) //broadcasteventprocessor.java
      ...
357. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      385. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      385. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
      390. if (!event.getService().equals("SNMP") && !event.getService().equals
      ("SNMPv1") && !event.getService().equals("SNMPv2"))
```

[Table of Contents] [Description]

Path 2340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
      409. if (parmValue == null)
```

[Table of Contents] [Description]

Path 2341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
      452. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
```

```
452. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
...
468. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
...
468. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
488. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
488. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
```

```
event) //broadcasteventprocessor.java
      ...
492. if (event.getInterface() == null)
```

[Table of Contents] [Description]

Path 2348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
509. if (parmValue == null)
```

[Table of Contents] [Description]

Path 2349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
554. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
554. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
563. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
563. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
570. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
570. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
      ...
593. if (!(cSvc.getNodeId() == nodeId))
```

[Table of Contents] [Description]

Path 2356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
      ...
614. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
       ...
614. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
625. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
640. if (!(cSvc.getNodeId() == nodeId && addr.getHostName().equals(ipAddr)))
```

[Table of Contents] [Description]

Path 2359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
625. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
661. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
625. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
661. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
       ...
675. if (!event.getService().equals("SNMP") && !event.getService().equals
```

```
("SNMPv1") && !event.getService().equals("SNMPv2"))
```

[Table of Contents] [Description]

Path 2362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
     ...
695. if (!(cSvc.getNodeId() == nodeId && addr.getHostName().equals(ipAddr) &&
cSvc.getServiceName().equals(svcName)))
```

[Table of Contents] [Description]

Path 2363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
     ...
698. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
     ...
698. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
     ...
719. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
```

```
...
719. if (log().isDebugEnabled())
```

[Table of Contents] [Description]

Path 2367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
765. private boolean isBlank(String string) //broadcasteventprocessor.java
...
767. if (string == null || string.trim().length() == 0) blank = true;
```

[Table of Contents] [Description]

Path 2368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. CollectableService(int dbNodeId, InetAddress address, String svcName,
org.opennms.netmgt.config.collectd.Package pkg) //collectableservice.java
...
180. if (svc == null)
```

[Table of Contents] [Description]

Path 2369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
268. public boolean isReady() //collectableservice.java
...
271. if (!Collectd.getInstance().isSchedulingCompleted())
```

[Table of Contents] [Description]

Path 2370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
288. private void sendEvent(String uei) //collectableservice.java
...
312. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
288. private void sendEvent(String uei) //collectableservice.java
       ...
312. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
329. if (processUpdates() == ABORT_COLLECTION)
```

[Table of Contents] [Description]

Path 2373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
346. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
346. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //collectableservice.java
       ...
364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. private boolean scheduledOutage() //collectableservice.java
       ...
428. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 2378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. private boolean scheduledOutage() //collectableservice.java
       ...
428. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 2379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
452. if (!m_updates.hasUpdates())
```

[Table of Contents] [Description]

Path 2380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
461. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
461. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
479. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
479. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
491. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
491. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
537. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
537. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
540. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
540. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
563. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
593. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
593. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
445. private boolean processUpdates() //collectableservice.java
       ...
597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public synchronized void init() //collectd.java
       ...
153. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public synchronized void init() //collectd.java
       ...
153. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public synchronized void init() //collectd.java
       ...
159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public synchronized void init() //collectd.java
       ...
159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. private void createEventProcessor(final Category log) //collectd.java
       ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. private void createEventProcessor(final Category log) //collectd.java
       ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. private void createEventProcessor(final Category log) //collectd.java
       ...
187. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. private void createEventProcessor(final Category log) //collectd.java
       ...
187. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void run() //collectd.java
      |
      ...
208. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void run() //collectd.java
      |
      ...
208. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private void createScheduler(final Category log, CollectdConfiguration
config) //collectd.java
      |
      ...
223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private void createScheduler(final Category log, CollectdConfiguration
config) //collectd.java
      |
      ...
223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private void createScheduler(final Category log, CollectdConfiguration
config) //collectd.java
      |
      ...
228. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private void createScheduler(final Category log, CollectdConfiguration
config) //collectd.java
      |...
      |228. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
      |...
      |239. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
      |...
      |239. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
      |...
      |260. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
      |...
      |260. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
264. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
264. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
268. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
268. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
272. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
272. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
276. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
276. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
284. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
284. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
292. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
     ...
292. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
     ...
334. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
     ...
334. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
     ...
338. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
      |...
      |338. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
      |...
      |342. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. private void loadscheduledOutagesConfigFactory(final Category
log) //collectd.java
      |...
      |342. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      |...
      |354. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      |...
      |354. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      ...
358. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      ...
358. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      ...
362. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private void loadConfigFactory(final Category log) //collectd.java
      ...
362. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
380. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
380. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
391. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
391. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized void start() //collectd.java
      ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. public synchronized void stop() //collectd.java
      ...
415. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. public synchronized void stop() //collectd.java
      ...
415. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public synchronized void pause() //collectd.java
      423. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 2451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public synchronized void pause() //collectd.java
       ...
431. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public synchronized void pause() //collectd.java
       ...
431. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
438. public synchronized void resume() //collectd.java
439. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 2454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
438. public synchronized void resume() //collectd.java
       ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
438. public synchronized void resume() //collectd.java
       ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
     ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
     ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
     ...
551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
     ...
551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
     ...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
        ...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
        ...
599. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
        ...
599. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
        ...
607. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
        ...
607. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
      |...
      |655. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
581. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //collectd.java
      |...
      |655. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. void setAttribute(String property, Object value) //collectorupdates.java
      |117. if (m_properties == null)
```

[Table of Contents] [Description]

Path 2469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. Object getAttribute(String property) //collectorupdates.java
      |...
      |130. if (m_properties != null)
```

[Table of Contents] [Description]

Path 2470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. protected String getFullOid(String ifIndex) //datasource.java
      |...
      |119. if (this.getOid() == null)
```

[Table of Contents] [Description]

Path 2471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. protected String getFullOid(String ifIndex) //datasource.java
      ...
123. if (this.getInstance().equals(MibObject.INSTANCE_IFINDEX))
```

[Table of Contents] [Description]

Path 2472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public boolean equals(Object object) //mibobject.java
256. if (object == null)
```

[Table of Contents] [Description]

Path 2473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public boolean equals(Object object) //mibobject.java
      ...
267. if (m_oid.equals(aMibObject.getOid()))
```

[Table of Contents] [Description]

Path 2474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
261. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
261. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
271. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
271. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
275. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
275. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
279. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
279. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
290. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
290. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
294. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
294. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
298. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
298. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
310. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
310. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
314. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
314. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
318. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
318. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
322. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
322. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
326. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
326. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 2498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
334. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
334. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
     ...
343. if (m_rrdPath == null)
```

[Table of Contents] [Description]

Path 2501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
351. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
351. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
358. if (!f.isDirectory())
```

[Table of Contents] [Description]

Path 2504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
365. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      ...
365. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |
      ...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public void initialize(Map parameters) //snmpcollector.java
      |
      ...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
402. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 2509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
418. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
418. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
426. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
426. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
433. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
433. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
```

```
parameters) //snmpcollector.java
    ...
450. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
    ...
450. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
    ...
460. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
    ...
460. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
    ...
492. if (rs.wasNull())
```

[Table of Contents] [Description]

Path 2520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
495. if (rs.wasNull())
```

[Table of Contents] [Description]

Path 2521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
498. if (str != null)
```

[Table of Contents] [Description]

Path 2522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
508. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
508. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
527. if (nodeID == -1)
```

[Table of Contents] [Description]

Path 2527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
530. if (primaryIfIndex == -1)
```

[Table of Contents] [Description]

Path 2528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
536. if (isSnmpPrimary != DbIpInterfaceEntry.SNMP_PRIMARY)
```

[Table of Contents] [Description]

Path 2529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
```

```
...
553. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
553. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
559. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
559. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
566. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
566. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
571. if (sysoid == null)
```

[Table of Contents] [Description]

Path 2536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
616. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
616. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
623. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
      623. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
      663. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
      663. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
      ...
      674. if (currSNMPPriValue == null)
```

[Table of Contents] [Description]

Path 2543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |
```

```
...
674. if (currSNMPPriValue == null)
```

[Table of Contents] [Description]

Path 2544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
676. else if (currSNMPPriValue.equals("P"))
```

[Table of Contents] [Description]

Path 2545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
678. else if (currSNMPPriValue.equals("S") && snmppriCollType.equals("P"))
```

[Table of Contents] [Description]

Path 2546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
678. else if (currSNMPPriValue.equals("S") && snmppriCollType.equals("P"))
```

[Table of Contents] [Description]

Path 2547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
680. else if (currSNMPPriValue.equals("C") && (snmppriCollType.equals("P") ||
snmppriCollType.equals("S")))
```

[Table of Contents] [Description]

Path 2548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
680. else if (currSNMPPriValue.equals("C") && (snmppriCollType.equals("P") ||
snmppriCollType.equals("S")))
```

[Table of Contents] [Description]

Path 2549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
680. else if (currSNMPPriValue.equals("C") && (snmppriCollType.equals("P") ||
snmppriCollType.equals("S")))
```

[Table of Contents] [Description]

Path 2550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
695. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
695. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
```

```
730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
735. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
735. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
...
773. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
773. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
780. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
780. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
787. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
787. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |795. if (ifMap.size() < 1)
```

[Table of Contents] [Description]

Path 2563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |811. if (!ifInfo.getOidList().isEmpty())
```

[Table of Contents] [Description]

Path 2564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |824. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 2565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |824. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 2566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
```

```
829. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
829. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
840. if (forceVersion.equals("snmpv2"))
```

[Table of Contents] [Description]

Path 2569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
842. else if (forceVersion.equals("snmpv1"))
```

[Table of Contents] [Description]

Path 2570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
     ...
865. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      ...
865. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
904. if (peer == null)
```

[Table of Contents] [Description]

Path 2573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
915. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
915. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      ...
935. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     |935. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     |941. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     |941. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     |963. if (nodeInfo == null)
```

[Table of Contents] [Description]

Path 2580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
```

```
968. if (ifMap == null)
```

[Table of Contents] [Description]

Path 2581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
980. if (!ifInfo.getOidList().isEmpty())
```

[Table of Contents] [Description]

Path 2582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1002. if (tmp != null)
```

[Table of Contents] [Description]

Path 2583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1028. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1028. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     1079. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     1079. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     1098. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     1098. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
     1107. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
1107. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
1126. if (nodeCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
1126. if (nodeCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
1126. if (nodeCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     |...
```

```
1152. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1152. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1171. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1171. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1180. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      |
      ...
1180. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      |
      ...
1200. if (ifCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      |
      ...
1200. if (ifCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      |
      ...
1200. if (ifCollector.timedout() == true)
```

[Table of Contents] [Description]

Path 2603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
      |
      ...
1217. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
     ...
1217. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
     ...
1329. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
     ...
1329. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
     ...
1357. if (snmpCollectorEntries == null)
```

[Table of Contents] [Description]

Path 2608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
     ...
```

```
1374. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
...
1374. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
...
1389. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
...
1389. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
...
1394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1403. if (ifInfo.getDsList() == null)
```

[Table of Contents] [Description]

Path 2615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1429. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1278. private boolean updateRRDs(String collectionName, NetworkInterface iface,
SnmpNodeCollector nodeCollector, SnmpIfCollector ifCollector) //snmpcollector.java
      ...
1429. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1463. public String getRRDValue(DataSource ds, SNMPCollectorEntry collectorEntry)
throws IllegalArgumentException //snmpcollector.java
      ...
1468. if (ds.getOid() == null)
```

[Table of Contents] [Description]

Path 2618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1463. public String getRRDValue(DataSource ds, SNMPCollectorEntry collectorEntry)
throws IllegalArgumentException //snmpcollector.java
      ...
1472. if (ds.getInstance().equals(MibObject.INSTANCE_IFINDEX))
```

[Table of Contents] [Description]

Path 2619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1463. public String getRRDValue(DataSource ds, SNMPCollectorEntry collectorEntry)
throws IllegalArgumentException //snmpcollector.java
      ...
1480. if (snmpVar == null)
```

[Table of Contents] [Description]

Path 2620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1463. public String getRRDValue(DataSource ds, SNMPCollectorEntry collectorEntry)
throws IllegalArgumentException //snmpcollector.java
      ...
1484. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1463. public String getRRDValue(DataSource ds, SNMPCollectorEntry collectorEntry)
throws IllegalArgumentException //snmpcollector.java
      ...
1484. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
      ...
```

```
1549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
...
1549. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
...
1563. if (m_host != null)
```

[Table of Contents] [Description]

Path 2625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
...
1563. if (m_host != null)
```

[Table of Contents] [Description]

Path 2626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
...
1572. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1544. private void generateForceRescanEvent(String ifAddress, EventProxy
eventProxy) //snmpcollector.java
      ...
1572. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1610. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1610. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1616. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1616. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1658. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
      ...
1658. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1687. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1687. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 2636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1693. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1693. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 2638:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1743. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2639:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
      ...
1743. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2640:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public SNMPCollectorEntry(SnmpVarBind[] vars, List objList, String
ifIndex) //snmpcollectorentry.java
      ...
126. if (ifIndex != null)
```

[Table of Contents] [Description]

Path 2641:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public SNMPCollectorEntry(SnmpVarBind[] vars, List objList, String
ifIndex) //snmpcollectorentry.java
      ...
126. if (ifIndex != null)
```

[Table of Contents] [Description]

Path 2642:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     ...
156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2643:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     ...
156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2644:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     ...
172. if (mibObject.getInstance().equals(MibObject.INSTANCE_IFINDEX))
```

[Table of Contents] [Description]

Path 2645:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     ...
185. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2646:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     ...
185. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2647:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     |...
     188. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2648:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     |...
     188. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2649:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     |...
     197. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2650:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public void update(SnmpVarBind[] vars, String
ifIndex) //snmpcollectorentry.java
     |...
     197. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2651:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
     |
```

```
...
262. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2652:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
...
262. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2653:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
...
270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2654:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
...
270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2655:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
...
277. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2656:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
     ...
277. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2657:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
     ...
317. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2658:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public SnmpIfCollector(SnmpSession session, Signaler signaler, String
primaryIfIndex, Map ifMap, int ifCount, int maxVarsPerPdu) //snmpifcollector.java
     ...
317. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2659:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public static SnmpObjectId stop_oid(String oid) //snmpifcollector.java
     ...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2660:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public static SnmpObjectId stop_oid(String oid) //snmpifcollector.java
     ...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2661:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2662:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2663:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
441. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2664:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
441. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2665:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2666:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. private SnmpPduPacket getNextSnmpV1Pdu(String ifAddress) //snmpifcollector.java
      ...
447. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2667:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
487. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2668:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
487. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2669:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2670:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
499. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2671:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
557. if (instance.equals("0"))
```

[Table of Contents] [Description]

Path 2672:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
568. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2673:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
476. private SnmpPduPacket getNextSnmpV2Pdu(String ifAddress) //snmpifcollector.java
      ...
568. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2674:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
610. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2675:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
610. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2676:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |623. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2677:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |623. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2678:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |636. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2679:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |636. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2680:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |
```

```
...
647. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2681:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
...
647. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2682:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
...
661. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2683:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
...
661. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2684:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
...
691. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2685:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      691. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2686:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      694. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2687:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      694. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2688:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      704. if (m_responses >= m_numInterfaces)
```

[Table of Contents] [Description]

Path 2689:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      734. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2690:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
     |...
     734. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2691:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
     |...
     776. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2692:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
     |...
     776. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2693:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
     |...
     794. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2694:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
     |
```

```
      ...
794.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2695:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
824.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2696:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
824.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2697:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
835.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2698:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      ...
835.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2699:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |844. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2700:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |844. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2701:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |855. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2702:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |855. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2703:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |860. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2704:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpifcollector.java
      |...
      |860. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2705:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      |...
      |907. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2706:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      |...
      |907. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2707:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      |...
      |913. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2708:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      |
```

```
      ...
913.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2709:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902.  public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      ...
917.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2710:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902.  public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      ...
917.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2711:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902.  public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      ...
922.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2712:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
902.  public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpifcollector.java
      ...
922.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2713:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
945. public void snmpTimeoutError(SnmpSession session, SnmpSyntax
pdu) //snmpifcollector.java
      |...
      |950. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2714:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
945. public void snmpTimeoutError(SnmpSession session, SnmpSyntax
pdu) //snmpifcollector.java
      |...
      |950. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2715:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
978. public boolean timedout() //snmpifcollector.java
      |979. if (m_error)
```

[Table of Contents] [Description]

Path 2716:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1005. private static List buildV2CombinedOidList(Map ifMap) //snmpifcollector.java
      |...
      |1021. if (!allOids.contains(oid))
```

[Table of Contents] [Description]

Path 2717:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1005. private static List buildV2CombinedOidList(Map ifMap) //snmpifcollector.java
      |...
      |1021. if (!allOids.contains(oid))
```

[Table of Contents] [Description]

Path 2718:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. public SnmpNodeCollector(SnmpSession session, Signaler signaler, List objList,
int maxVarsPerPu) //snmpnodecollector.java
     ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2719:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. public SnmpNodeCollector(SnmpSession session, Signaler signaler, List objList,
int maxVarsPerPu) //snmpnodecollector.java
     ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2720:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. public SnmpNodeCollector(SnmpSession session, Signaler signaler, List objList,
int maxVarsPerPu) //snmpnodecollector.java
     ...
211. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2721:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. public SnmpNodeCollector(SnmpSession session, Signaler signaler, List objList,
int maxVarsPerPu) //snmpnodecollector.java
     ...
211. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2722:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public static SnmpObjectId stop_oid(String oid) //snmpnodecollector.java
     ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2723:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public static SnmpObjectId stop_oid(String oid) //snmpnodecollector.java
       ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2724:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
     ...
344. if (lastValue == 0)
```

[Table of Contents] [Description]

Path 2725:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
     ...
349. if (instancePrefix == null)
```

[Table of Contents] [Description]

Path 2726:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
     ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2727:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
     ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2728:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
     ...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2729:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. public SnmpPduPacket getNextPdu(String ifAddress) //snmpnodecollector.java
       ...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2730:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
       ...
404. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2731:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
       ...
404. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2732:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
       ...
417. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2733:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
       ...
417. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2734:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     |...
     |430. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2735:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     |...
     |430. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2736:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     |...
     |442. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2737:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     |...
     |442. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2738:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
```

```
pdu) //snmpnodecollector.java
     ...
449. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2739:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
449. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2740:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2741:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2742:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
466. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2743:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
466. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2744:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2745:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2746:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2747:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
     ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2748:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      |...
      |492. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2749:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      |...
      |492. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2750:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      |...
      |512. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2751:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      |...
      |512. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2752:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
```

```
      ...
518.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2753:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      ...
518.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2754:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      ...
524.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2755:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      ...
524.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2756:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396.  public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      ...
549.  if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2757:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
396. public void snmpReceivedPdu(SnmpSession session, int command, SnmpPduPacket
pdu) //snmpnodecollector.java
      |...
      549. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 2758:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
      |...
      575. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2759:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
      |...
      575. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2760:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
      |...
      581. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2761:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
      |...
      581. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2762:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
     |...
     585. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2763:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
     |...
     585. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2764:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
     |...
     590. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2765:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
570. public void snmpInternalError(SnmpSession session, int error, SnmpSyntax
pdu) //snmpnodecollector.java
     |...
     590. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2766:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
613. public void snmpTimeoutError(SnmpSession session, SnmpSyntax
pdu) //snmpnodecollector.java
     |
```

```
      ...
618.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2767:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
613. public void snmpTimeoutError(SnmpSession session, SnmpSyntax
pdu) //snmpnodecollector.java
      ...
618.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2768:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. public boolean timedout() //snmpnodecollector.java
      647.  if (m_error)
```

[Table of Contents] [Description]

Path 2769:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public void snmpReceivedPdu(SnmpSession sess, int command, SnmpPduPacket
pkt) //snmpresponsehandler.java
      ...
70.  if (((SnmpPduRequest) pkt).getErrorStatus() == SnmpPduPacket.ErrNoError)
```

[Table of Contents] [Description]

Path 2770:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
      ...
95.  if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 2771:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
      ...
      95. if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 2772:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public static synchronized ActiondConfigFactory getInstance
() //actiondconfigfactory.java
      144. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2773:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. private CapsdConfigFactory(String configFile) throws IOException,
MarshalException, ValidationException //capsdconfigfactory.java
      ...
      545. if (!m_urlMap.containsKey(url))
```

[Table of Contents] [Description]

Path 2774:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. private CapsdConfigFactory(String configFile) throws IOException,
MarshalException, ValidationException //capsdconfigfactory.java
      ...
      545. if (!m_urlMap.containsKey(url))
```

[Table of Contents] [Description]

Path 2775:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
603. public static synchronized CapsdConfigFactory getInstance
() //capsdconfigfactory.java
      604. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2776:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      716. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2777:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      716. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2778:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      722. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2779:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      722. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2780:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2781:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2782:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      738. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2783:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      738. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2784:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      739. if (insStmt != null) insStmt.close();
```

[Table of Contents] [Description]

Path 2785:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
```

```
...
739. if (insStmt != null) insStmt.close();
```

[Table of Contents] [Description]

Path 2786:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
...
740. if (nxtStmt != null) nxtStmt.close();
```

[Table of Contents] [Description]

Path 2787:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
...
740. if (nxtStmt != null) nxtStmt.close();
```

[Table of Contents] [Description]

Path 2788:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
...
741. if (loadStmt != null) loadStmt.close();
```

[Table of Contents] [Description]

Path 2789:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
...
741. if (loadStmt != null) loadStmt.close();
```

[Table of Contents] [Description]

Path 2790:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
    ...
742. if (delFromOutagesStmt != null) delFromOutagesStmt.close();
```

[Table of Contents] [Description]

Path 2791:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
    ...
742. if (delFromOutagesStmt != null) delFromOutagesStmt.close();
```

[Table of Contents] [Description]

Path 2792:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
    ...
743. if (delFromIfServicesStmt != null) delFromIfServicesStmt.close();
```

[Table of Contents] [Description]

Path 2793:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
638. public void syncServices(Connection conn) throws
SQLException //capsdconfigfactory.java
    ...
743. if (delFromIfServicesStmt != null) delFromIfServicesStmt.close();
```

[Table of Contents] [Description]

Path 2794:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
    ...
779. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2795:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      779. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2796:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      791. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2797:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      791. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2798:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
      ...
      834. if (str != null)
```

[Table of Contents] [Description]

Path 2799:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
```

```
      ...
      841. if (result != null) result.close();
```

[Table of Contents] [Description]

Path 2800:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      841. if (result != null) result.close();
```

[Table of Contents] [Description]

Path 2801:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      843. if (ipRetStmt != null) ipRetStmt.close();
```

[Table of Contents] [Description]

Path 2802:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      843. if (ipRetStmt != null) ipRetStmt.close();
```

[Table of Contents] [Description]

Path 2803:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      929. if (ipPkg != null)
```

[Table of Contents] [Description]

Path 2804:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      932. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2805:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      932. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2806:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      937. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2807:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      937. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2808:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
      964. if (str != null)
```

[Table of Contents] [Description]

Path 2809:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      977. if (!svcToBePolled)
```

[Table of Contents] [Description]

Path 2810:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      981. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2811:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      981. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2812:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      986. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2813:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |
```

```
...
986. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2814:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1016. if (ifUpdateStmt != null) ifUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2815:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1016. if (ifUpdateStmt != null) ifUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2816:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1017. if (allSvcUpdateStmt !=null) allSvcUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2817:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1017. if (allSvcUpdateStmt !=null) allSvcUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2818:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      1018. if (svcRetStmt != null) svcRetStmt.close();
```

[Table of Contents] [Description]

Path 2819:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      1018. if (svcRetStmt != null) svcRetStmt.close();
```

[Table of Contents] [Description]

Path 2820:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      1019. if (svcUpdateStmt != null) svcUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2821:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      1019. if (svcUpdateStmt != null) svcUpdateStmt.close();
```

[Table of Contents] [Description]

Path 2822:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      |...
      1051. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2823:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1051. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2824:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1080. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2825:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1080. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2826:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1085. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2827:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
```

```
...
1085. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2828:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1088. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2829:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1088. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2830:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1097. if (str != null)
```

[Table of Contents] [Description]

Path 2831:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1103. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2832:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1103. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2833:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1124. if (result != null) result.close();
```

[Table of Contents] [Description]

Path 2834:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1124. if (result != null) result.close();
```

[Table of Contents] [Description]

Path 2835:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1125. if (ipRetStmt != null) ipRetStmt.close();
```

[Table of Contents] [Description]

Path 2836:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1125. if (ipRetStmt != null) ipRetStmt.close();
```

[Table of Contents] [Description]

Path 2837:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1133. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2838:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1133. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2839:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1139. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2840:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1139. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2841:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
```

```
...
1159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2842:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2843:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1221. if (CollectdConfigFactory.getInstance().lookupInterfaceServicePair
(lwIf.getAddress(), "SNMP") || CollectdConfigFactory.getInstance
().lookupInterfaceServicePair(lwIf.getAddress(), "SNMPv1") ||
CollectdConfigFactory.getInstance().lookupInterfaceServicePair(lwIf.getAddress
(), "SNMPv2"))
```

[Table of Contents] [Description]

Path 2844:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1221. if (CollectdConfigFactory.getInstance().lookupInterfaceServicePair
(lwIf.getAddress(), "SNMP") || CollectdConfigFactory.getInstance
().lookupInterfaceServicePair(lwIf.getAddress(), "SNMPv1") ||
CollectdConfigFactory.getInstance().lookupInterfaceServicePair(lwIf.getAddress
(), "SNMPv2"))
```

[Table of Contents] [Description]

Path 2845:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
```

```
1223. else if ( 'C' != lwIf.getSnmpPrimaryState())
```

[Table of Contents] [Description]

Path 2846:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1223. else if ( 'C' != lwIf.getSnmpPrimaryState())
```

[Table of Contents] [Description]

Path 2847:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1233. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2848:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1233. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2849:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
...
1246. if (updateStmt != null) updateStmt.close();
```

[Table of Contents] [Description]

Path 2850:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1246. if (updateStmt != null) updateStmt.close();
```

[Table of Contents] [Description]

Path 2851:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1252. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2852:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1040. public synchronized void syncSnmpPrimaryState(Connection conn) throws
SQLException //capsdconfigfactory.java
      ...
1252. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2853:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
      ...
1361. if (!found)
```

[Table of Contents] [Description]

Path 2854:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
      ...
1376. if (scan.equals("off"))
```

[Table of Contents] [Description]

Path 2855:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1289. public ProtocolInfo[] getProtocolSpecification(InetAddress
address) //capsdconfigfactory.java
      |...
      1409. if (scan.equals("off"))
```

[Table of Contents] [Description]

Path 2856:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1461. public SmbAuth getSmbAuth(String target) //capsdconfigfactory.java
      |...
      1467. if (a.getContent() != null && a.getContent().equalsIgnoreCase(target))
```

[Table of Contents] [Description]

Path 2857:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |...
      1497. if (mgt.getPolicy() == null || mgt.getPolicy().equalsIgnoreCase
      ("managed"))
```

[Table of Contents] [Description]

Path 2858:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |...
      1522. if (mgt.getPolicy() == null || mgt.getPolicy().equalsIgnoreCase
      ("managed"))
```

[Table of Contents] [Description]

Path 2859:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
      |
```

```
...
1551. if (mgt.getPolicy() == null || mgt.getPolicy().equalsIgnoreCase
("managed"))
```

[Table of Contents] [Description]

Path 2860:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
...
1568. if (found_denial)
```

[Table of Contents] [Description]

Path 2861:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1481. public boolean isAddressUnmanaged(InetAddress
target) //capsdconfigfactory.java
...
1570. else if (found_accept)
```

[Table of Contents] [Description]

Path 2862:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
...
1654. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2863:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
...
1654. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2864:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
      ...
1672. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2865:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
      ...
1672. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2866:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
      ...
1674. if (s != null) s.close();
```

[Table of Contents] [Description]

Path 2867:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1646. public boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //capsdconfigfactory.java
      ...
1674. if (s != null) s.close();
```

[Table of Contents] [Description]

Path 2868:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
      ...
1686. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2869:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
      ...
1686. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2870:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
      ...
1694. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2871:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
      ...
1694. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2872:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
      ...
1705. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2873:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
```

```
...
1705. if (ifIndex != -1)
```

[Table of Contents] [Description]

Path 2874:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
...
1715. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2875:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
...
1715. if (rs != null) rs.close();
```

[Table of Contents] [Description]

Path 2876:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
...
1717. if (stmt != null) rs.close();
```

[Table of Contents] [Description]

Path 2877:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1683. public int getInterfaceDbNodeId(Connection dbConn, InetAddress ifAddress, int
ifIndex) throws SQLException //capsdconfigfactory.java
...
1717. if (stmt != null) rs.close();
```

[Table of Contents] [Description]

Path 2878:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1727. public long getRescanFrequency() //capsdconfigfactory.java
      ...
1730. if (m_config.hasRescanFrequency())
```

[Table of Contents] [Description]

Path 2879:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1743. public long getInitialSleepTime() //capsdconfigfactory.java
      ...
1746. if (m_config.hasInitialSleepTime())
```

[Table of Contents] [Description]

Path 2880:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1775. public boolean getAbortProtocolScansFlag() //capsdconfigfactory.java
      ...
1779. if (abortProperty != null && abortProperty.equals("true"))
```

[Table of Contents] [Description]

Path 2881:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1785. public boolean getDeletePropagationEnabled() //capsdconfigfactory.java
      ...
1789. if (propagationProperty != null && propagationProperty.equals("false"))
```

[Table of Contents] [Description]

Path 2882:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public static synchronized CatFactory getInstance() //categoryfactory.java
     150. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2883:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. private void createPackageIpListMap() //collectdconfigfactory.java
     ...
191. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2884:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. private void createPackageIpListMap() //collectdconfigfactory.java
     ...
191. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2885:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
319. public static synchronized CollectdConfigFactory getInstance
() //collectdconfigfactory.java
     320. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2886:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
400. public synchronized boolean interfaceInPackage(String iface, Package
pkg) //collectdconfigfactory.java
     ...
405. if (!filterPassed)
```

[Table of Contents] [Description]

Path 2887:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
438. public synchronized boolean serviceInPackageAndEnabled(String svcName,
org.opennms.netmgt.config.collectd.Package pkg) //collectdconfigfactory.java
     ...
448. if (status.equals("on"))
```

[Table of Contents] [Description]

Path 2888:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public synchronized InetAddress determinePrimarySnmpInterface(List addressList,
boolean strict) //collectdconfigfactory.java
     ...
525. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2889:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public synchronized InetAddress determinePrimarySnmpInterface(List addressList,
boolean strict) //collectdconfigfactory.java
     ...
525. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2890:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
507. public synchronized InetAddress determinePrimarySnmpInterface(List addressList,
boolean strict) //collectdconfigfactory.java
     ...
530. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2891:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public synchronized InetAddress compareAndSelectPrimaryCollectionInterface
(String svcName, InetAddress currentIf, InetAddress oldPrimary, String method,
boolean strict) //collectdconfigfactory.java
     ...
568. if (lookupInterfaceServicePair(currentIf.getHostAddress(), svcName))
```

[Table of Contents] [Description]

Path 2892:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public synchronized InetAddress compareAndSelectPrimaryCollectionInterface
(String svcName, InetAddress currentIf, InetAddress oldPrimary, String method,
boolean strict) //collectdconfigfactory.java
     ...
574. if (oldPrimary == null)
```

[Table of Contents] [Description]

Path 2893:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public synchronized InetAddress compareAndSelectPrimaryCollectionInterface
(String svcName, InetAddress currentIf, InetAddress oldPrimary, String method,
boolean strict) //collectdconfigfactory.java
      |...
      |586. if (lookupInterfaceServicePair(currentIf.getHostAddress(), svcName))
```

[Table of Contents] [Description]

Path 2894:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public synchronized InetAddress compareAndSelectPrimaryCollectionInterface
(String svcName, InetAddress currentIf, InetAddress oldPrimary, String method,
boolean strict) //collectdconfigfactory.java
      |...
      |599. if (lookupInterfaceServicePair(currentIf.getHostAddress(), svcName))
```

[Table of Contents] [Description]

Path 2895:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. public synchronized InetAddress compareAndSelectPrimaryCollectionInterface
(String svcName, InetAddress currentIf, InetAddress oldPrimary, String method,
boolean strict) //collectdconfigfactory.java
      |...
      |608. if (newPrimary != null)
```

[Table of Contents] [Description]

Path 2896:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
641. private boolean hasSpecific(Package pkg, long
addr) //collectdconfigfactory.java
      |...
      |646. if (speca == addr)
```

[Table of Contents] [Description]

Path 2897:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private boolean interfaceInFilter(String iface, Package pkg, Category
log) //collectdconfigfactory.java
     |...
     |680. if (!filterPassed)
```

[Table of Contents] [Description]

Path 2898:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private boolean interfaceInFilter(String iface, Package pkg, Category
log) //collectdconfigfactory.java
     |...
     |680. if (!filterPassed)
```

[Table of Contents] [Description]

Path 2899:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public static synchronized void init() throws IOException, MarshalException,
ValidationException, ClassNotFoundException //databaseconnectionfactory.java
     |...
     |133. if (in != null) in.close();
```

[Table of Contents] [Description]

Path 2900:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public static synchronized void init() throws IOException, MarshalException,
ValidationException, ClassNotFoundException //databaseconnectionfactory.java
     |...
     |133. if (in != null) in.close();
```

[Table of Contents] [Description]

Path 2901:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public static synchronized DbConnectionFactory getInstance
() //databaseconnectionfactory.java
     |153. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2902:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
198. public static synchronized DatabaseSchemaConfigFactory getInstance
() //databaseschemaconfigfactory.java
     |199. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2903:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public String constructJoinExprForTable(Table
t) //databaseschemaconfigfactory.java
     |...
     |310. if (i != 0)
```

[Table of Contents] [Description]

Path 2904:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
303. public String constructJoinExprForTable(Table
t) //databaseschemaconfigfactory.java
     |...
     |310. if (i != 0)
```

[Table of Contents] [Description]

Path 2905:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public Join[] getPrimaryJoinsForTable(Table
t) //databaseschemaconfigfactory.java
     |...
     |361. if (j != null)
```

[Table of Contents] [Description]

Path 2906:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public Join[] getPrimaryJoinsForTable(Table
t) //databaseschemaconfigfactory.java
     |...
     |361. if (j != null)
```

[Table of Contents] [Description]

Path 2907:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
220. public static synchronized DataCollectionConfigFactory getInstance
() //datacollectionconfigfactory.java
     221. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2908:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
     266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2909:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
     266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2910:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
     270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2911:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
```

```
...
270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2912:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
...
332. if (sysChoice.getSysoid() != null)
```

[Table of Contents] [Description]

Path 2913:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2914:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2915:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
...
352. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2916:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
352. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2917:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
373. if (addrList != null && addrList.size() > 0 || maskList != null &&
maskList.size() > 0)
```

[Table of Contents] [Description]

Path 2918:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2919:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2920:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     ...
394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2921:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     |...
     394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2922:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     |...
     406. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2923:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public List getMibObjectList(String cName, String aSysoid, String anAddress,
int ifType) //datacollectionconfigfactory.java
     |...
     406. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2924:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
     |...
     467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2925:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
```

```
...
467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2926:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
...
510. if (groupIfType.indexOf(',') != -1)
```

[Table of Contents] [Description]

Path 2927:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
...
527. if (ifTypeStr.equals(groupIfType))
```

[Table of Contents] [Description]

Path 2928:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
...
555. if (nextComma == -1)
```

[Table of Contents] [Description]

Path 2929:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
...
569. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2930:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
      ...
569. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2931:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
      ...
574. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2932:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
451. private void processGroupName(String cName, String groupName, int ifType, List
mibObjectList) //datacollectionconfigfactory.java
      ...
574. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2933:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. public int getStep(String cName) //datacollectionconfigfactory.java
      ...
621. if (collection != null)
```

[Table of Contents] [Description]

Path 2934:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
635. public List getRRAList(String cName) //datacollectionconfigfactory.java
      ...
637. if (collection != null)
```

[Table of Contents] [Description]

Path 2935:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
652. public String getSnmpStorageFlag(String
cName) //datacollectionconfigfactory.java
      |
      ...
      654. if (collection != null)
```

[Table of Contents] [Description]

Path 2936:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public int getMaxVarsPerPdu(String cName) //datacollectionconfigfactory.java
      |
      ...
      671. if (collection != null)
```

[Table of Contents] [Description]

Path 2937:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
14. public DbConfiguration(Database database) //dbconfiguration.java
      |
      ...
      18. if (parms[i].getName().equals("user"))
```

[Table of Contents] [Description]

Path 2938:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
14. public DbConfiguration(Database database) //dbconfiguration.java
      |
      ...
      18. if (parms[i].getName().equals("user"))
```

[Table of Contents] [Description]

Path 2939:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
14. public DbConfiguration(Database database) //dbconfiguration.java
      |
      ...
      20. else if (parms[i].getName().equals("password"))
```

[Table of Contents] [Description]

Path 2940:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
14. public DbConfiguration(Database database) //dbconfiguration.java
      ...
      20. else if (parms[i].getName().equals("password"))
```

[Table of Contents] [Description]

Path 2941:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public Collection getTargetCommands(Path path, int index, String target) throws
IOException, MarshalException, ValidationException //destinationpathmanager.java
      ...
      124. if (targets[i].getName().equals(target))
```

[Table of Contents] [Description]

Path 2942:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. public boolean pathHasTarget(Path path, String target) throws IOException,
MarshalException, ValidationException //destinationpathmanager.java
      ...
      162. if (curTarget.getName().equals(target))
```

[Table of Contents] [Description]

Path 2943:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public static synchronized DhcpdConfigFactory getInstance
() //dhcpdconfigfactory.java
      142. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2944:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public static synchronized DiscoveryConfigFactory getInstance
() //discoveryconfigfactory.java
      148. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2945:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. static synchronized public EventconfFactory getInstance
() //eventconffactory.java
     |121. if (!initialized)
```

[Table of Contents] [Description]

Path 2946:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. public List getEvents(String uei) //eventconffactory.java
     |...
     |194. if (events.size() > 0)
```

[Table of Contents] [Description]

Path 2947:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
276. public synchronized void removeEvents(String uei) throws MarshalException,
ValidationException, IOException //eventconffactory.java
     |...
     |280. if (events == null)
```

[Table of Contents] [Description]

Path 2948:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
276. public synchronized void removeEvents(String uei) throws MarshalException,
ValidationException, IOException //eventconffactory.java
     |...
     |284. if (m_events.removeAll(events) == true)
```

[Table of Contents] [Description]

Path 2949:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
295. public synchronized void removeEvent(String uei, Mask mask) throws
MarshalException, IOException, ValidationException //eventconffactory.java
     |...
     |299. if (events == null)
```

[Table of Contents] [Description]

Path 2950:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
295. public synchronized void removeEvent(String uei, Mask mask) throws
MarshalException, IOException, ValidationException //eventconffactory.java
     ...
310. if (m_events.remove(current) == true)
```

[Table of Contents] [Description]

Path 2951:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public static synchronized EventdConfigFactory getInstance
() //eventdconfigfactory.java
145. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2952:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public static synchronized EventsArchiverConfigFactory getInstance
() //eventsarchiverconfigfactory.java
145. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2953:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
     ...
212. if (!m_inUse)
```

[Table of Contents] [Description]

Path 2954:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
```

```
...
212. if (!m_inUse)
```

[Table of Contents] [Description]

Path 2955:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
...
218. if (m_delegate.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 2956:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
...
218. if (m_delegate.getAutoCommit() == false)
```

[Table of Contents] [Description]

Path 2957:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
...
224. if (isTracing)
```

[Table of Contents] [Description]

Path 2958:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
...
224. if (isTracing)
```

[Table of Contents] [Description]

Path 2959:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
      |...
      |233. if (isTracing)
```

[Table of Contents] [Description]

Path 2960:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
      |...
      |233. if (isTracing)
```

[Table of Contents] [Description]

Path 2961:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
      |...
      |239. if (isTracing)
```

[Table of Contents] [Description]

Path 2962:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
      |...
      |239. if (isTracing)
```

[Table of Contents] [Description]

Path 2963:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
      |...
      |256. if (isTracing)
```

[Table of Contents] [Description]

Path 2964:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
208. public synchronized void close() throws
SQLException //legacydbconnectionfactory.java
     ...
256. if (isTracing)
```

[Table of Contents] [Description]

Path 2965:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. public synchronized boolean isClosed() throws
SQLException //legacydbconnectionfactory.java
     ...
274. if (m_inUse)
```

[Table of Contents] [Description]

Path 2966:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     ...
894. if (isTracing)
```

[Table of Contents] [Description]

Path 2967:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     ...
894. if (isTracing)
```

[Table of Contents] [Description]

Path 2968:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
```

```
...
903. if (isTracing)
```

[Table of Contents] [Description]

Path 2969:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
...
903. if (isTracing)
```

[Table of Contents] [Description]

Path 2970:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
...
912. if (isTracing)
```

[Table of Contents] [Description]

Path 2971:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
...
912. if (isTracing)
```

[Table of Contents] [Description]

Path 2972:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
...
918. if (isTracing)
```

[Table of Contents] [Description]

Path 2973:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     ...
     918. if (isTracing)
```

[Table of Contents] [Description]

Path 2974:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     ...
     936. if (isTracing)
```

[Table of Contents] [Description]

Path 2975:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
877. public Connection getConnection() throws
SQLException //legacydbconnectionfactory.java
     ...
     936. if (isTracing)
```

[Table of Contents] [Description]

Path 2976:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. protected Range removeSpecificFromRange(final String
spec) //mergeablerange.java
     ...
     176. if (!coversSpecific(spec))
```

[Table of Contents] [Description]

Path 2977:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
86. static synchronized public NotifdConfigFactory getInstance
() //notifdconfigfactory.java
     87. if (!initialized)
```

[Table of Contents] [Description]

Path 2978:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. public static String getPrettyStatus() throws IOException, MarshalException,
ValidationException //notifdconfigfactory.java
     |129. if (!initialized)
```

[Table of Contents] [Description]

Path 2979:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. public static String getPrettyStatus() throws IOException, MarshalException,
ValidationException //notifdconfigfactory.java
     |...
     |136. if (status.equals("on"))
```

[Table of Contents] [Description]

Path 2980:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. public static String getPrettyStatus() throws IOException, MarshalException,
ValidationException //notifdconfigfactory.java
     |...
     |138. else if (status.equals("off"))
```

[Table of Contents] [Description]

Path 2981:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public boolean matchNotificationParameters(Event event, Notification
notification) //notifdconfigmanager.java
     |...
     |213. if (parmValue == null)
```

[Table of Contents] [Description]

Path 2982:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. static synchronized public NotificationFactory getInstance
() //notificationfactory.java
     |99. if (!initialized)
```

[Table of Contents] [Description]

Path 2983:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public Notification[] getNotifForEvent(Event event) throws IOException,
MarshalException, ValidationException //notificationmanager.java
     ...
156. if (!matchAll)
```

[Table of Contents] [Description]

Path 2984:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
493. public String getServiceNoticeStatus(String nodeID, String ipaddr, String
service) throws SQLException //notificationmanager.java
     ...
511. if (notify == null)
```

[Table of Contents] [Description]

Path 2985:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
527. public void updateNoticeWithUserInfo(String userId, int noticeId, String media,
String contactInfo, String autoNotify) throws
SQLException //notificationmanager.java
     ...
529. if (noticeId < 0) return;
```

[Table of Contents] [Description]

Path 2986:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. public static synchronized OpennmsServerConfigFactory getInstance
() //opennmsserverconfigfactory.java
     141. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2987:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
162. public synchronized boolean verifyServer() //opennmsserverconfigfactory.java
     |
```

```
...
164. if (flag.equals("true"))
```

[Table of Contents] [Description]

Path 2988:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
145. public static synchronized OutageManagerConfig getInstance
() //outagemanagerconfigfactory.java
     146. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2989:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public synchronized boolean deletePropagation
() //outagemanagerconfigfactory.java
     ...
     180. if (deletePropagation != null && deletePropagation.equals("false"))
```

[Table of Contents] [Description]

Path 2990:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. public static synchronized PollerConfigFactory getInstance
() //pollerconfigfactory.java
     258. if (!m_loaded)
```

[Table of Contents] [Description]

Path 2991:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. public synchronized boolean getXmlrpc() //pollerconfigfactory.java
     ...
     330. if (flag.equals("true"))
```

[Table of Contents] [Description]

Path 2992:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. public synchronized boolean pollAllIfNoCriticalServiceDefined
() //pollerconfigfactory.java
      |...
      |362. if (flag.equals("true"))
```

[Table of Contents] [Description]

Path 2993:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. public synchronized boolean nodeOutageProcessingEnabled
() //pollerconfigfactory.java
      |...
      |373. if (status.equals("on"))
```

[Table of Contents] [Description]

Path 2994:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
386. public synchronized boolean serviceUnresponsiveEnabled
() //pollerconfigfactory.java
      |...
      |388. if (enabled.equals("true"))
```

[Table of Contents] [Description]

Path 2995:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
      |...
      |424. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2996:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
      |...
      |424. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2997:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
       ...
428. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2998:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
       ...
428. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 2999:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
       ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3000:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
399. private void createPackageIpListMap() //pollerconfigfactory.java
       ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3001:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
       ...
483. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3002:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
483. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3003:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
486. if (!filterPassed)
```

[Table of Contents] [Description]

Path 3004:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
515. if (speca == addr)
```

[Table of Contents] [Description]

Path 3005:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
551. public synchronized boolean serviceInPackageAndEnabled(String svcName,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
558. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3006:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
551. public synchronized boolean serviceInPackageAndEnabled(String svcName,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
558. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3007:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
551. public synchronized boolean serviceInPackageAndEnabled(String svcName,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
      ...
571. if (status.equals("on"))
```

[Table of Contents] [Description]

Path 3008:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
610. public synchronized org.opennms.netmgt.config.poller.Package
getFirstPackageMatch(String ipaddr) //pollerconfigfactory.java
      ...
616. if (inPkg)
```

[Table of Contents] [Description]

Path 3009:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
663. public synchronized boolean isPolled(String ipaddr) //pollerconfigfactory.java
      ...
669. if (inPkg)
```

[Table of Contents] [Description]

Path 3010:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
147. public static synchronized PollOutagesConfigFactory getInstance
() //polloutagesconfigfactory.java
148. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3011:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. private void setOutCalTime(Calendar outCal, String
timeStr) //polloutagesconfigmanager.java
      ...
```

```
128. if (tempDate == null)
```

[Table of Contents] [Description]

Path 3012:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. private void setOutCalTime(Calendar outCal, String
timeStr) //polloutagesconfigmanager.java
...
151. if (tempDate == null)
```

[Table of Contents] [Description]

Path 3013:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
201. public synchronized String getOutageType(String
name) //polloutagesconfigmanager.java
...
203. if (out == null)
```

[Table of Contents] [Description]

Path 3014:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
217. public synchronized Time[] getOutageTimes(String
name) //polloutagesconfigmanager.java
...
219. if (out == null)
```

[Table of Contents] [Description]

Path 3015:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized Interface[] getInterfaces(String
name) //polloutagesconfigmanager.java
...
235. if (out == null)
```

[Table of Contents] [Description]

Path 3016:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. public synchronized boolean isInterfaceInOutage(String linterface, String
outName) //polloutagesconfigmanager.java
     |...
     |253. if (out == null)
```

[Table of Contents] [Description]

Path 3017:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
269. public synchronized boolean isInterfaceInOutage(String linterface, Outage
out) //polloutagesconfigmanager.java
     |270. if (out == null)
```

[Table of Contents] [Description]

Path 3018:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
294. public synchronized boolean isTimeInOutage(Calendar cal, String
outName) //polloutagesconfigmanager.java
     |...
     |296. if (out == null)
```

[Table of Contents] [Description]

Path 3019:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
312. public synchronized boolean isTimeInOutage(long time, String
outName) //polloutagesconfigmanager.java
     |...
     |314. if (out == null)
```

[Table of Contents] [Description]

Path 3020:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
     |...
     |335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3021:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3022:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |338. if (out == null)
```

[Table of Contents] [Description]

Path 3023:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |368. if (cal.get(Calendar.DAY_OF_WEEK) == dayInMap.intValue())
```

[Table of Contents] [Description]

Path 3024:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |378. if (cal.get(Calendar.DAY_OF_MONTH) == intOTimeDay)
```

[Table of Contents] [Description]

Path 3025:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |387. if (oTimeDay != null && !inOutage)
```

[Table of Contents] [Description]

Path 3026:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |395. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3027:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |395. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3028:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized boolean isTimeInOutage(Calendar cal, Outage
out) //polloutagesconfigmanager.java
      |...
      |404. if (curCalTime >= outCalBeginTime && curCalTime < outCalEndTime)
```

[Table of Contents] [Description]

Path 3029:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
474. public synchronized Node[] getNodeIds(String
name) //polloutagesconfigmanager.java
      |...
      |476. if (out == null)
```

[Table of Contents] [Description]

Path 3030:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
494. public synchronized boolean isNodeIdInOutage(long lnodeid, String
outName) //polloutagesconfigmanager.java
```

```
      ...
      496. if (out == null)
```

[Table of Contents] [Description]

Path 3031:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
502. public synchronized Calendar getEndOfOutage(String
outName) //polloutagesconfigmanager.java
      ...
      504. if (out == null)
```

[Table of Contents] [Description]

Path 3032:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
517. public synchronized Calendar getEndOfOutage(Outage
out) //polloutagesconfigmanager.java
      ...
      542. if (cal.get(Calendar.DAY_OF_WEEK) == dayInMap.intValue())
```

[Table of Contents] [Description]

Path 3033:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
517. public synchronized Calendar getEndOfOutage(Outage
out) //polloutagesconfigmanager.java
      ...
      551. if (cal.get(Calendar.DAY_OF_MONTH) == intOTimeDay)
```

[Table of Contents] [Description]

Path 3034:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
517. public synchronized Calendar getEndOfOutage(Outage
out) //polloutagesconfigmanager.java
      ...
      570. if (curCalTime >= outCalBeginTime && curCalTime < outCalEndTime)
```

[Table of Contents] [Description]

Path 3035:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
588. public synchronized boolean isNodeIdInOutage(long lnodeid, Outage
out) //polloutagesconfigmanager.java
     589. if (out == null)
```

[Table of Contents] [Description]

Path 3036:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
     ...
     94. if (hIndex != -1)
```

[Table of Contents] [Description]

Path 3037:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
     ...
     102. if (hIndex >= mIndex)
```

[Table of Contents] [Description]

Path 3038:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
     ...
     113. if (mIndex >= sIndex)
```

[Table of Contents] [Description]

Path 3039:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
     ...
     117. if (hIndex >= sIndex)
```

[Table of Contents] [Description]

Path 3040:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
      |...
      |130. if (hrStr != null)
```

[Table of Contents] [Description]

Path 3041:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
      |...
      |133. if (minStr != null)
```

[Table of Contents] [Description]

Path 3042:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private long parseRollingWindow(String rolling) throws
IllegalArgumentException //rtcconfigfactory.java
      |...
      |136. if (secStr != null)
```

[Table of Contents] [Description]

Path 3043:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
216. public static synchronized RTCConfigFactory getInstance
() //rtcconfigfactory.java
      |217. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3044:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public static synchronized ScriptdConfigFactory getInstance
() //scriptdconfigfactory.java
      |150. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3045:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public static synchronized void init() throws IOException, MarshalException,
ValidationException //serviceconfigfactory.java
     ...
135. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3046:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public static synchronized void init() throws IOException, MarshalException,
ValidationException //serviceconfigfactory.java
     ...
135. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3047:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. public static synchronized ServiceConfigFactory getInstance
() //serviceconfigfactory.java
     171. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3048:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. private void purgeOtherDefs(final MergeableDefinition updatedDef, final
Definition eventDef) //snmpconfigmanager.java
     ...
122. if (def.getConfigDef() == updatedDef.getConfigDef()) continue;
```

[Table of Contents] [Description]

Path 3049:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. public SnmpEventInfo(Event event) //snmpeventinfo.java
     ...
```

```
107. if (parmValue == null) continue;
```

[Table of Contents] [Description]

Path 3050:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
212. if (getCommunityString() != null) definition.setReadCommunity
(getCommunityString());
```

[Table of Contents] [Description]

Path 3051:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
212. if (getCommunityString() != null) definition.setReadCommunity
(getCommunityString());
```

[Table of Contents] [Description]

Path 3052:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
216. if (getRetryCount() != 0) definition.setRetry(getRetryCount());
```

[Table of Contents] [Description]

Path 3053:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
216. if (getRetryCount() != 0) definition.setRetry(getRetryCount());
```

[Table of Contents] [Description]

Path 3054:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
217. if (getTimeout() != 0) definition.setTimeout(getTimeout());
```

[Table of Contents] [Description]

Path 3055:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
217. if (getTimeout() != 0) definition.setTimeout(getTimeout());
```

[Table of Contents] [Description]

Path 3056:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
218. if (getPort() != 0) definition.setPort(getPort());
```

[Table of Contents] [Description]

Path 3057:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
210. public Definition createDef() //snmpeventinfo.java
     ...
218. if (getPort() != 0) definition.setPort(getPort());
```

[Table of Contents] [Description]

Path 3058:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
268. private boolean isBlank(String string) //snmpeventinfo.java
     ...
270. if (string == null || string.trim().length() == 0) blank = true;
```

[Table of Contents] [Description]

Path 3059:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public static synchronized String marshallConfig() //snmppeerfactory.java
     ...
222. if (writer != null) writer.close();
```

[Table of Contents] [Description]

Path 3060:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public static synchronized String marshallConfig() //snmppeerfactory.java
     ...
222. if (writer != null) writer.close();
```

[Table of Contents] [Description]

Path 3061:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public static synchronized SnmpPeerFactory getInstance() //snmppeerfactory.java
     239. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3062:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
     ...
333. if (def.getVersion().equals("v1"))
```

[Table of Contents] [Description]

Path 3063:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
     ...
333. if (def.getVersion().equals("v1"))
```

[Table of Contents] [Description]

Path 3064:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
335. else if (def.getVersion().equals("v2c"))
```

[Table of Contents] [Description]

Path 3065:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
335. else if (def.getVersion().equals("v2c"))
```

[Table of Contents] [Description]

Path 3066:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
339. if (supportedSnmpVersion == SnmpSMI.SNMPV1 || supportedSnmpVersion ==
SnmpSMI.SNMPV2)
```

[Table of Contents] [Description]

Path 3067:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
339. if (supportedSnmpVersion == SnmpSMI.SNMPV1 || supportedSnmpVersion ==
SnmpSMI.SNMPV2)
```

[Table of Contents] [Description]

Path 3068:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
339. if (supportedSnmpVersion == SnmpSMI.SNMPV1 || supportedSnmpVersion ==
SnmpSMI.SNMPV2)
```

[Table of Contents] [Description]

Path 3069:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
385. if (def.hasPort())
```

[Table of Contents] [Description]

Path 3070:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
321. private SnmpPeer create(InetAddress addr, Definition def, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
385. if (def.hasPort())
```

[Table of Contents] [Description]

Path 3071:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
499. if (m_config.getReadCommunity() != null)
```

[Table of Contents] [Description]

Path 3072:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
      ...
499. if (m_config.getReadCommunity() != null)
```

[Table of Contents] [Description]

Path 3073:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
```

```
...
502. if (m_config.getWriteCommunity() != null)
```

[Table of Contents] [Description]

Path 3074:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
...
502. if (m_config.getWriteCommunity() != null)
```

[Table of Contents] [Description]

Path 3075:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
...
506. if (m_config.hasRetry())
```

[Table of Contents] [Description]

Path 3076:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
...
506. if (m_config.hasRetry())
```

[Table of Contents] [Description]

Path 3077:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
...
509. if (m_config.hasTimeout())
```

[Table of Contents] [Description]

Path 3078:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
424. public synchronized SnmpPeer getPeer(InetAddress host, int
supportedSnmpVersion) //snmppeerfactory.java
     ...
509. if (m_config.hasTimeout())
```

[Table of Contents] [Description]

Path 3079:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. private void createPackageIpListMap() //threshdconfigfactory.java
     ...
175. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3080:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. private void createPackageIpListMap() //threshdconfigfactory.java
     ...
175. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3081:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. public static synchronized ThreshdConfigFactory getInstance
() //threshdconfigfactory.java
     302. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3082:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.threshd.Package pkg) //threshdconfigfactory.java
     ...
394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3083:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.threshd.Package pkg) //threshdconfigfactory.java
     ...
394. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3084:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.threshd.Package pkg) //threshdconfigfactory.java
     ...
397. if (!filterPassed)
```

[Table of Contents] [Description]

Path 3085:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.threshd.Package pkg) //threshdconfigfactory.java
     ...
426. if (speca == addr)
```

[Table of Contents] [Description]

Path 3086:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public synchronized boolean serviceInPackageAndEnabled(String svcName,
org.opennms.netmgt.config.threshd.Package pkg) //threshdconfigfactory.java
     ...
472. if (status.equals("on"))
```

[Table of Contents] [Description]

Path 3087:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
177. public static synchronized ThresholdingConfigFactory getInstance
() //thresholdingconfigfactory.java
     178. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3088:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. public String getRrdRepository(String
groupName) //thresholdingconfigfactory.java
      |...
      |198. if (group == null)
```

[Table of Contents] [Description]

Path 3089:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
218. public Collection getThresholds(String
groupName) //thresholdingconfigfactory.java
      |...
      |220. if (group == null)
```

[Table of Contents] [Description]

Path 3090:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. public static synchronized TrapdConfigFactory getInstance
() //trapdconfigfactory.java
      |149. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3091:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public boolean isUserOnDuty(String user, Calendar time) throws IOException,
MarshalException, ValidationException //usermanager.java
      |...
      |162. if (!m_dutySchedules.containsKey(user))
```

[Table of Contents] [Description]

Path 3092:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public boolean isUserOnDuty(String user, Calendar time) throws IOException,
MarshalException, ValidationException //usermanager.java
      |...
```

```
174. if (result)
```

[Table of Contents] [Description]

Path 3093:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
243. public String getContactInfo(String userID, String command) throws IOException,
MarshalException, ValidationException //usermanager.java
     ...
247. if (user == null)
```

[Table of Contents] [Description]

Path 3094:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
270. public String getEmail(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
     ...
275. if (user == null)
```

[Table of Contents] [Description]

Path 3095:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. public String getPagerEmail(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
     ...
303. if (user == null)
```

[Table of Contents] [Description]

Path 3096:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
327. public String getXMPPAddress(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
     ...
332. if (user == null)
```

[Table of Contents] [Description]

Path 3097:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. public String getNumericPage(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
      |...
      361. if (user == null)
```

[Table of Contents] [Description]

Path 3098:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. public String getTextPin(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
      |...
      389. if (user == null)
```

[Table of Contents] [Description]

Path 3099:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
412. public String getTextPage(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
      |...
      417. if (user == null)
```

[Table of Contents] [Description]

Path 3100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public String getNumericPin(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
      |...
      445. if (user == null)
```

[Table of Contents] [Description]

Path 3101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
623. private String hexToString(byte[] data) //usermanager.java
      |...
      627. if ((data.length % 2) != 0)
```

[Table of Contents] [Description]

Path 3102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
653. public boolean comparePasswords(String userID, String
aPassword) //usermanager.java
     ...
655. if (user == null)
```

[Table of Contents] [Description]

Path 3103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public static synchronized VacuumdConfigFactory getInstance
() //vacuumdconfigfactory.java
     147. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. static synchronized public ViewFactory getInstance() //viewfactory.java
     119. if (!initialized)
```

[Table of Contents] [Description]

Path 3105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
247. public static synchronized VulnscandConfigFactory getInstance
() //vulnscandconfigfactory.java
     248. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
309. public static boolean isInterfaceInDB(Connection dbConn, InetAddress ifAddress)
throws SQLException //vulnscandconfigfactory.java
     ...
320. if (rs.next())
```

[Table of Contents] [Description]

Path 3107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
502. public long getRescanFrequency() //vulnscandconfigfactory.java
      |...
      |505. if (m_config.hasRescanFrequency())
```

[Table of Contents] [Description]

Path 3108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
518. public long getInitialSleepTime() //vulnscandconfigfactory.java
      |...
      |521. if (m_config.hasInitialSleepTime())
```

[Table of Contents] [Description]

Path 3109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. public static synchronized XmlrpcdConfigFactory getInstance
() //xmlrpcdconfigfactory.java
      |155. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized boolean eventSubscribed(String
uei) //xmlrpcdconfigfactory.java
      |...
      |190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized boolean eventSubscribed(String
uei) //xmlrpcdconfigfactory.java
      |...
      |190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public void run() //client.java
       ...
128. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public void run() //client.java
       ...
128. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public void run() //client.java
       ...
181. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public void run() //client.java
       ...
181. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public synchronized void start() //client.java
       212. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |277. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |277. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |311. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |311. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |318. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void run() //client.java
       |...
       |318. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
183. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
183. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
190. if (m_worker != null && m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
196. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
     ...
196. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
222. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
222. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
231. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
231. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized void stop() //dhcpd.java
     |282. if (m_worker == null)
```

[Table of Contents] [Description]

Path 3143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
     |...
     |359. if (m_status != STARTING)
```

[Table of Contents] [Description]

Path 3144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
     |...
     |362. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
     |...
     |362. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
     |...
     |367. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
367. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
386. } else if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
409. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
409. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
416. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
416. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
421. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public void run() //dhcpd.java
       |
       ...
421. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
442. public void update(Observable inst, Object ignored) //dhcpd.java
       ...
444. if (m_clients != null)
```

[Table of Contents] [Description]

Path 3158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
442. public void update(Observable inst, Object ignored) //dhcpd.java
       ...
444. if (m_clients != null)
```

[Table of Contents] [Description]

Path 3159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
       ...
236. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
       ...
236. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
       ...
240. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
      |...
      240. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public void close() //poller.java
      |...
      283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public void close() //poller.java
      |...
      283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      358. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      358. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      375. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      375. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |...
      386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |
      ...
      386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |
      ...
      411. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |
      ...
      411. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      |
      ...
      430. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
```

```
...
430. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
...
449. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
...
449. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
...
463. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
...
463. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      ...
      471. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      ...
      471. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      ...
      474. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      ...
      474. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private static void setHwAddress(String hwAddressStr) //poller.java
      ...
      509. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private static void setHwAddress(String hwAddressStr) //poller.java
       ...
509. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private static void setHwAddress(String hwAddressStr) //poller.java
       ...
520. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private static void setHwAddress(String hwAddressStr) //poller.java
       ...
520. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public synchronized void start() //receiver.java
      77. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
145. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
145. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
151. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
151. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
158. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public synchronized void start() //receiver2.java
      77. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver2.java
       |
       ...
145. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver2.java
       |
       ...
145. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver2.java
       |
       ...
151. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver2.java
       |
       ...
151. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 3200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void run() //receiver2.java
       |
       ...
158. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      102. if (eventUei == null)
```

[Table of Contents] [Description]

Path 3202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      |
      ...
      116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
      136. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
      136. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
      191. if (c != null)
```

[Table of Contents] [Description]

Path 3209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
      191. if (c != null)
```

[Table of Contents] [Description]

Path 3210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      227. if (m_manager != null)
```

[Table of Contents] [Description]

Path 3211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      257. if (ctest != null)
```

[Table of Contents] [Description]

Path 3212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      257. if (ctest != null)
```

[Table of Contents] [Description]

Path 3213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      294. if (s.hasTimeout())
```

[Table of Contents] [Description]

Path 3214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      296. else if (cfg.hasTimeout())
```

[Table of Contents] [Description]

Path 3215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      300. if (s.hasRetries())
```

[Table of Contents] [Description]

Path 3216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
302. else if (cfg.hasRetries())
```

[Table of Contents] [Description]

Path 3217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
317. if (ir.hasTimeout())
```

[Table of Contents] [Description]

Path 3218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
319. else if (cfg.hasTimeout())
```

[Table of Contents] [Description]

Path 3219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
323. if (ir.hasRetries())
```

[Table of Contents] [Description]

Path 3220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
325. else if (cfg.hasRetries())
```

[Table of Contents] [Description]

Path 3221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
342. if (url.hasTimeout())
```

[Table of Contents] [Description]

Path 3222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
344. else if (cfg.hasTimeout())
```

[Table of Contents] [Description]

Path 3223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
348. if (url.hasRetries())
```

[Table of Contents] [Description]

Path 3224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
       ...
350. else if (cfg.hasRetries())
```

[Table of Contents] [Description]

Path 3225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
       ...
475. if (m_manager != null)
```

[Table of Contents] [Description]

Path 3226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
      |...
      |475. if (m_manager != null)
```

[Table of Contents] [Description]

Path 3227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
500. public synchronized void pause() //discovery.java
      |501. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
513. public synchronized void resume() //discovery.java
      |514. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 3229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. IPAddressRangeGenerator(long start, long end) //ipaddressrange.java
      |128. if (start > end)
```

[Table of Contents] [Description]

Path 3230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public Object nextElement() //ipaddressrange.java
      |154. if (m_next > m_end)
```

[Table of Contents] [Description]

Path 3231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
309. boolean contains(long ipAddr) //ipaddressrange.java
      |310. if (ipAddr >= m_begin && ipAddr <= m_end)
```

[Table of Contents] [Description]

Path 3232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
438. static long byteToLong(byte b) //ipaddressrange.java
      |...
      |440. if (r < 0)
```

[Table of Contents] [Description]

Path 3233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public static long convertToLong(byte[] addr) //ipaddressrange.java
      |466. if (addr == null)
```

[Table of Contents] [Description]

Path 3234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public boolean hasNext() //ipgenerator.java
      |...
      |148. if (!rc)
```

[Table of Contents] [Description]

Path 3235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public boolean hasNext() //ipgenerator.java
      |...
      |148. if (!rc)
```

[Table of Contents] [Description]

Path 3236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public Object next() //ipgenerator.java
      |...
      |168. if (i.hasNext())
```

[Table of Contents] [Description]

Path 3237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public Object next() //ipgenerator.java
       ...
168. if (i.hasNext())
```

[Table of Contents] [Description]

Path 3238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public Object next() //ipgenerator.java
       ...
174. if (obj == null)
```

[Table of Contents] [Description]

Path 3239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
       ...
226. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
       ...
226. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
       ...
232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
236. if (m_isInitial)
```

[Table of Contents] [Description]

Path 3244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
236. if (m_isInitial)
```

[Table of Contents] [Description]

Path 3245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
248. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
```

```
InterruptedException //ipgenerator.java
    ...
248. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
    ...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
    ...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
    ...
263. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
    ...
263. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      |266. if (m_isInitial)
```

[Table of Contents] [Description]

Path 3252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      |266. if (m_isInitial)
```

[Table of Contents] [Description]

Path 3253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      |278. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      |278. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      |282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |
      ...
      282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |
      ...
      286. if (m_isRestarted)
```

[Table of Contents] [Description]

Path 3258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |
      ...
      286. if (m_isRestarted)
```

[Table of Contents] [Description]

Path 3259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |
      ...
      298. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
```

```
      ...
298. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
309. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
309. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
313. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      ...
313. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      316. if (m_isRestarted)
```

[Table of Contents] [Description]

Path 3266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. private synchronized boolean doWait(long maxWait) throws
InterruptedException //ipgenerator.java
      |...
      316. if (m_isRestarted)
```

[Table of Contents] [Description]

Path 3267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public synchronized Object remove(long timeout) throws InterruptedException,
FifoQueueException //ipgenerator.java
      |...
      443. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public synchronized Object remove(long timeout) throws InterruptedException,
FifoQueueException //ipgenerator.java
      |...
      443. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public synchronized Object remove(long timeout) throws InterruptedException,
FifoQueueException //ipgenerator.java
      |...
      453. if (m_iter.hasNext())
```

[Table of Contents] [Description]

Path 3270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. private/* synchronized */boolean poll(IPPollAddress addr) throws IOException,
InterruptedException //pingmanager.java
     ...
190. if (wtime <= 0)
```

[Table of Contents] [Description]

Path 3271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. private/* synchronized */boolean poll(IPPollAddress addr) throws IOException,
InterruptedException //pingmanager.java
     ...
193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. private/* synchronized */boolean poll(IPPollAddress addr) throws IOException,
InterruptedException //pingmanager.java
     ...
193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public synchronized void start() //pingmanager.java
     243. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. public synchronized void stop() //pingmanager.java
     260. if (m_worker == null)
```

[Table of Contents] [Description]

Path 3275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized void pause() //pingmanager.java
     |282. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 3276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
292. public synchronized void resume() //pingmanager.java
     |293. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 3277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
314. public synchronized int getStatus() //pingmanager.java
     |315. if (m_worker != null && !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
     |...
     |362. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
     |...
     |362. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
367. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
367. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
373. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
400. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       |
       ...
411. if (m_status != STOP_PENDING)
```

[Table of Contents] [Description]

Path 3290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       ...
411. if (m_status != STOP_PENDING)
```

[Table of Contents] [Description]

Path 3291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       ...
411. if (m_status != STOP_PENDING)
```

[Table of Contents] [Description]

Path 3292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       ...
418. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void run() //pingmanager.java
       ...
418. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
432. public synchronized boolean isPinging(InetAddress addr) //pingmanager.java
       433. if (addr == null)
```

[Table of Contents] [Description]

Path 3295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
477. PingManager(FifoQueue addressQ, FifoQueue discoveredQ, short filterId, int
threads, int pktsPerSecond) throws IOException //pingmanager.java
     |478. if (threads < 1)
```

[Table of Contents] [Description]

Path 3296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
477. PingManager(FifoQueue addressQ, FifoQueue discoveredQ, short filterId, int
threads, int pktsPerSecond) throws IOException //pingmanager.java
     |...
     |481. if (pktsPerSecond < 1)
```

[Table of Contents] [Description]

Path 3297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
521. public synchronized void start() //pingmanager.java
     |522. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
     |...
     |559. if (m_worker == null)
```

[Table of Contents] [Description]

Path 3299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
     |...
     |563. if (m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
       ...
567. } else if (m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
582. public synchronized void pause() //pingmanager.java
583. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 3302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
605. public synchronized void resume() //pingmanager.java
606. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 3303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
624. public synchronized int getStatus() //pingmanager.java
625. if (m_worker != null && !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. public void run() //pingmanager.java
       ...
686. if (doAdd)
```

[Table of Contents] [Description]

Path 3305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
646. public void run() //pingmanager.java
       ...
```

```
686. if (doAdd)
```

[Table of Contents] [Description]

Path 3306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. public synchronized void start() //suspecteventgenerator.java
     129. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
145. public synchronized void stop() //suspecteventgenerator.java
     146. if (m_worker == null)
```

[Table of Contents] [Description]

Path 3308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized int getStatus() //suspecteventgenerator.java
     161. if (m_worker != null && !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
     ...
     193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
     ...
     193. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
       ...
204. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
       ...
214. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
       ...
214. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
       ...
271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public void run() //suspecteventgenerator.java
       ...
271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public static void loadConfiguration(Reader rdr) throws IOException,
MarshalException, ValidationException //eventconfigurationmanager.java
      ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public static void loadConfiguration(Reader rdr) throws IOException,
MarshalException, ValidationException //eventconfigurationmanager.java
      ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
      ...
170. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
      ...
170. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
      ...
176. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
       ...
176. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
       ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
       ...
182. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
       ...
187. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public void stop() //eventd.java
       ...
187. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void init() //eventd.java
      ...
263. if (tempConn != null)
```

[Table of Contents] [Description]

Path 3327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void init() //eventd.java
      ...
263. if (tempConn != null)
```

[Table of Contents] [Description]

Path 3328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void init() //eventd.java
      ...
309. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void init() //eventd.java
      ...
309. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
341. public void pause() //eventd.java
      342. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
341. public void pause() //eventd.java
       ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
341. public void pause() //eventd.java
       ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
369. public void resume() //eventd.java
370. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 3334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
369. public void resume() //eventd.java
       ...
389. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
369. public void resume() //eventd.java
       ...
389. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. private static org.opennms.netmgt.xml.event.Snmp transform
(org.opennms.netmgt.xml.eventconf.Snmp src) //eventexpander.java
     ...
201. if (src.hasGeneric())
```

[Table of Contents] [Description]

Path 3337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. private static org.opennms.netmgt.xml.event.Snmp transform
(org.opennms.netmgt.xml.eventconf.Snmp src) //eventexpander.java
     ...
201. if (src.hasGeneric())
```

[Table of Contents] [Description]

Path 3338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. private static org.opennms.netmgt.xml.event.Snmp transform
(org.opennms.netmgt.xml.eventconf.Snmp src) //eventexpander.java
     ...
203. if (src.hasSpecific())
```

[Table of Contents] [Description]

Path 3339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. private static org.opennms.netmgt.xml.event.Snmp transform
(org.opennms.netmgt.xml.eventconf.Snmp src) //eventexpander.java
     ...
203. if (src.hasSpecific())
```

[Table of Contents] [Description]

Path 3340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
410. private static org.opennms.netmgt.xml.eventconf.Event lookup(Snmp
snmpInfo) //eventexpander.java
     ...
415. if (snmpInfo == null)
```

[Table of Contents] [Description]

Path 3341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
469. private static org.opennms.netmgt.xml.eventconf.Event lookup(Event
event) //eventexpander.java
      |470. if (event == null)
```

[Table of Contents] [Description]

Path 3342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
469. private static org.opennms.netmgt.xml.eventconf.Event lookup(Event
event) //eventexpander.java
      |...
      |519. if ((eConf.getMask() == null))
```

[Table of Contents] [Description]

Path 3343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
469. private static org.opennms.netmgt.xml.eventconf.Event lookup(Event
event) //eventexpander.java
      |...
      |534. if (!snmpEidIsPartOfMask)
```

[Table of Contents] [Description]

Path 3344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
555. private static void expandParms(Autoaction[] autoactions, Event
event) //eventexpander.java
      |...
      |568. if (expanded)
```

[Table of Contents] [Description]

Path 3345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
555. private static void expandParms(Autoaction[] autoactions, Event
event) //eventexpander.java
      |
```

```
...
568. if (expanded)
```

[Table of Contents] [Description]

Path 3346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. private static void expandParms(Operaction[] operactions, Event
event) //eventexpander.java
...
588. if (expanded)
```

[Table of Contents] [Description]

Path 3347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. private static void expandParms(Operaction[] operactions, Event
event) //eventexpander.java
...
588. if (expanded)
```

[Table of Contents] [Description]

Path 3348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
...
678. if (EventConfigurationManager.isSecureTag("mask"))
```

[Table of Contents] [Description]

Path 3349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
...
678. if (EventConfigurationManager.isSecureTag("mask"))
```

[Table of Contents] [Description]

Path 3350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
686. if (e.getUei() == null)
```

[Table of Contents] [Description]

Path 3351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
686. if (e.getUei() == null)
```

[Table of Contents] [Description]

Path 3352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
691. if (e.getSnmp() == null && econf.getSnmp() != null)
```

[Table of Contents] [Description]

Path 3353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
691. if (e.getSnmp() == null && econf.getSnmp() != null)
```

[Table of Contents] [Description]

Path 3354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
696. if (EventConfigurationManager.isSecureTag("descr"))
```

[Table of Contents] [Description]

Path 3355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
696. if (EventConfigurationManager.isSecureTag("descr"))
```

[Table of Contents] [Description]

Path 3356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
698. if (e.getDescr() == null && econf.getDescr() != null)
```

[Table of Contents] [Description]

Path 3357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
698. if (e.getDescr() == null && econf.getDescr() != null)
```

[Table of Contents] [Description]

Path 3358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
703. if (EventConfigurationManager.isSecureTag("logmsg"))
```

[Table of Contents] [Description]

Path 3359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
703. if (EventConfigurationManager.isSecureTag("logmsg"))
```

[Table of Contents] [Description]

Path 3360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
705. if (e.getLogmsg() == null && econf.getLogmsg() != null)
```

[Table of Contents] [Description]

Path 3361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
705. if (e.getLogmsg() == null && econf.getLogmsg() != null)
```

[Table of Contents] [Description]

Path 3362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
710. if (EventConfigurationManager.isSecureTag("severity"))
```

[Table of Contents] [Description]

Path 3363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
710. if (EventConfigurationManager.isSecureTag("severity"))
```

[Table of Contents] [Description]

Path 3364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
712. if (e.getSeverity() == null && econf.getSeverity() != null)
```

[Table of Contents] [Description]

Path 3365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
712. if (e.getSeverity() == null && econf.getSeverity() != null)
```

[Table of Contents] [Description]

Path 3366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
717. if (EventConfigurationManager.isSecureTag("correlation"))
```

[Table of Contents] [Description]

Path 3367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
717. if (EventConfigurationManager.isSecureTag("correlation"))
```

[Table of Contents] [Description]

Path 3368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
719. if (e.getCorrelation() == null && econf.getCorrelation() != null)
```

[Table of Contents] [Description]

Path 3369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
719. if (e.getCorrelation() == null && econf.getCorrelation() != null)
```

[Table of Contents] [Description]

Path 3370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
724. if (EventConfigurationManager.isSecureTag("operinstruct"))
```

[Table of Contents] [Description]

Path 3371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
724. if (EventConfigurationManager.isSecureTag("operinstruct"))
```

[Table of Contents] [Description]

Path 3372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
726. if (e.getOperinstruct() == null && econf.getOperinstruct() != null)
```

[Table of Contents] [Description]

Path 3373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
726. if (e.getOperinstruct() == null && econf.getOperinstruct() != null)
```

[Table of Contents] [Description]

Path 3374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
731. if (EventConfigurationManager.isSecureTag("autoaction"))
```

[Table of Contents] [Description]

Path 3375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
731. if (EventConfigurationManager.isSecureTag("autoaction"))
```

[Table of Contents] [Description]

Path 3376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
743. if (EventConfigurationManager.isSecureTag("operaction"))
```

[Table of Contents] [Description]

Path 3377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
743. if (EventConfigurationManager.isSecureTag("operaction"))
```

[Table of Contents] [Description]

Path 3378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
755. if (EventConfigurationManager.isSecureTag("autoacknowledge"))
```

[Table of Contents] [Description]

Path 3379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
755. if (EventConfigurationManager.isSecureTag("autoacknowledge"))
```

[Table of Contents] [Description]

Path 3380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
757. if (e.getAutoacknowledge() == null && econf.getAutoacknowledge() != null)
```

[Table of Contents] [Description]

Path 3381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
757. if (e.getAutoacknowledge() == null && econf.getAutoacknowledge() != null)
```

[Table of Contents] [Description]

Path 3382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
762. if (EventConfigurationManager.isSecureTag("loggroup"))
```

[Table of Contents] [Description]

Path 3383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
762. if (EventConfigurationManager.isSecureTag("loggroup"))
```

[Table of Contents] [Description]

Path 3384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
764. if (e.getLoggroupCount() == 0 && econf.getLoggroupCount() > 0)
```

[Table of Contents] [Description]

Path 3385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
       ...
764. if (e.getLoggroupCount() == 0 && econf.getLoggroupCount() > 0)
```

[Table of Contents] [Description]

Path 3386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
       ...
769. if (EventConfigurationManager.isSecureTag("tticket"))
```

[Table of Contents] [Description]

Path 3387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
       ...
769. if (EventConfigurationManager.isSecureTag("tticket"))
```

[Table of Contents] [Description]

Path 3388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
       ...
771. if (e.getTticket() == null && econf.getTticket() != null)
```

[Table of Contents] [Description]

Path 3389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
       ...
771. if (e.getTticket() == null && econf.getTticket() != null)
```

[Table of Contents] [Description]

Path 3390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
776. if (EventConfigurationManager.isSecureTag("forward"))
```

[Table of Contents] [Description]

Path 3391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
776. if (EventConfigurationManager.isSecureTag("forward"))
```

[Table of Contents] [Description]

Path 3392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
788. if (EventConfigurationManager.isSecureTag("script"))
```

[Table of Contents] [Description]

Path 3393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
788. if (EventConfigurationManager.isSecureTag("script"))
```

[Table of Contents] [Description]

Path 3394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
     ...
800. if (EventConfigurationManager.isSecureTag("mouseovertext"))
```

[Table of Contents] [Description]

Path 3395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
      ...
800. if (EventConfigurationManager.isSecureTag("mouseovertext"))
```

[Table of Contents] [Description]

Path 3396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
      ...
802. if (e.getMouseovertext() == null && econf.getMouseovertext() != null)
```

[Table of Contents] [Description]

Path 3397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
674. public static synchronized void expandEvent(Event e) //eventexpander.java
      ...
802. if (e.getMouseovertext() == null && econf.getMouseovertext() != null)
```

[Table of Contents] [Description]

Path 3398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
89. public void run() //eventhandler.java
      ...
92. if (m_eventLog == null)
```

[Table of Contents] [Description]

Path 3399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
      ...
146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
     |...
     146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
     |...
     153. if (obj == null)
```

[Table of Contents] [Description]

Path 3402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
     |...
     167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void run() //eventipcmanagerdefaultimpl.java
     |...
     167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
     |...
     282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      ...
282. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      ...
286. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      ...
286. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      ...
299. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      ...
299. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      |...
      |324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      |...
      |324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      |...
      |333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public synchronized void broadcastNow(Event
event) //eventipcmanagerdefaultimpl.java
      |...
      |333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. public synchronized void addEventListener(EventListener listener, List
ueilist) //eventipcmanagerdefaultimpl.java
      |...
```

```
372. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. public synchronized void addEventListener(EventListener listener, List
ueilist) //eventipcmanagerdefaultimpl.java
...
372. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. public synchronized void addEventListener(EventListener listener, List
ueilist) //eventipcmanagerdefaultimpl.java
...
404. if (!listenersList.contains(listener))
```

[Table of Contents] [Description]

Path 3417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. public synchronized void addEventListener(EventListener listener, List
ueilist) //eventipcmanagerdefaultimpl.java
...
404. if (!listenersList.contains(listener))
```

[Table of Contents] [Description]

Path 3418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public synchronized void addEventListener(EventListener listener, String
uei) //eventipcmanagerdefaultimpl.java
...
442. if (!listenersList.contains(listener))
```

[Table of Contents] [Description]

Path 3419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public synchronized void addEventListener(EventListener listener, String
uei) //eventipcmanagerdefaultimpl.java
      ...
442. if (!listenersList.contains(listener))
```

[Table of Contents] [Description]

Path 3420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public static synchronized EventIpcManagerFactory getInstance
() //eventipcmanagerfactory.java
      88. if (!m_loaded)
```

[Table of Contents] [Description]

Path 3421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
      281. if (pvalue == null)
```

[Table of Contents] [Description]

Path 3422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
      ...
288. if (value instanceof String)
```

[Table of Contents] [Description]

Path 3423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
      ...
290. else if (value instanceof Number)
```

[Table of Contents] [Description]

Path 3424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     |...
     292. else if (value instanceof SnmpInt32)
```

[Table of Contents] [Description]

Path 3425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     |...
     294. else if (value instanceof SnmpUInt32)
```

[Table of Contents] [Description]

Path 3426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     |...
     296. else if (value instanceof SnmpCounter64)
```

[Table of Contents] [Description]

Path 3427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     |...
     298. else if (value instanceof SnmpIPAddress)
```

[Table of Contents] [Description]

Path 3428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     |...
     300. else if (value instanceof SnmpOctetString)
```

[Table of Contents] [Description]

Path 3429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     ...
302. else if (value instanceof SnmpObjectId)
```

[Table of Contents] [Description]

Path 3430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
280. public static String getValueAsString(Value pvalue) //eventutil.java
     ...
305. if (value instanceof String)
```

[Table of Contents] [Description]

Path 3431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
447. if (nodeLabel != null)
```

[Table of Contents] [Description]

Path 3432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
490. if (ifAlias != null)
```

[Table of Contents] [Description]

Path 3433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
500. if (info == null)
```

[Table of Contents] [Description]

Path 3434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
504. if (info.getIdtext() != null)
```

[Table of Contents] [Description]

Path 3435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
504. if (info.getIdtext() != null)
```

[Table of Contents] [Description]

Path 3436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
504. if (info.getIdtext() != null)
```

[Table of Contents] [Description]

Path 3437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
512. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
512. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
512. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
518. if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
518. if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
518. if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
577. if (assetdescr != null)
```

[Table of Contents] [Description]

Path 3444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
591. if (assetcomment != null)
```

[Table of Contents] [Description]

Path 3445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
596. if (event.getParms() != null
```

[Table of Contents] [Description]

Path 3446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
608. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
612. if (parmValueStr == null)
```

[Table of Contents] [Description]

Path 3448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
625. if (event.getParms() != null
```

[Table of Contents] [Description]

Path 3449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
638. if (parmName == null)
```

[Table of Contents] [Description]

Path 3450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
652. if (event.getParms() != null
```

[Table of Contents] [Description]

Path 3451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
665. if (parmName == null)
```

[Table of Contents] [Description]

Path 3452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
669. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
769. public static String expandParms(String inp, Event event) //eventutil.java
     ...
816. if (ret.length() > 0 && !retStr.equals(inp))
```

[Table of Contents] [Description]

Path 3454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
        ...
276. if (name == null)
```

[Table of Contents] [Description]

Path 3455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
        ...
282. if (id != -1)
```

[Table of Contents] [Description]

Path 3456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
        ...
299. if (id != -1)
```

[Table of Contents] [Description]

Path 3457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. private int getServiceID(String name) throws SQLException //eventwriter.java
        ...
299. if (id != -1)
```

[Table of Contents] [Description]

Path 3458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
325. private String getHostName(String hostip) throws
SQLException //eventwriter.java
        ...
344. if (hostname == null)
```

[Table of Contents] [Description]

Path 3459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      456. if (event.getSnmp() != null)
```

[Table of Contents] [Description]

Path 3462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      456. if (event.getSnmp() != null)
```

[Table of Contents] [Description]

Path 3463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |
      ...
      456. if (event.getSnmp() != null)
```

[Table of Contents] [Description]

Path 3464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |...
      |566. if (event.getTticket().getState().equals("on"))
```

[Table of Contents] [Description]

Path 3465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |...
      |600. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. private void add(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |...
      |600. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. public void persistEvent(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |...
      |705. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
672. public void persistEvent(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      |...
```

```
705. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
66. private void findServer() throws
InstanceNotFoundException //eventhandlermbeanproxy.java
    ...
80. if (m_mbserver == null)
```

[Table of Contents] [Description]

Path 3470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public boolean equals(Object obj) //eventhandlermbeanproxy.java
    ...
125. if (p.m_mbserver.equals(m_mbserver) && p.m_listener.equals(m_listener))
```

[Table of Contents] [Description]

Path 3471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public synchronized void start() //tcpeventreceiver.java
158. if (m_status != START_PENDING && m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public synchronized void stop() //tcpeventreceiver.java
192. if (m_status == STOPPED)
```

[Table of Contents] [Description]

Path 3473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. public void addEventHandler(EventHandler handler) //tcpeventreceiver.java
    ...
255. if (!m_handlers.contains(handler))
```

[Table of Contents] [Description]

Path 3474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
253. public void addEventHandler(EventHandler handler) //tcpeventreceiver.java
       ...
255. if (!m_handlers.contains(handler))
```

[Table of Contents] [Description]

Path 3475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
279. public void setPort(Integer port) //tcpeventreceiver.java
       280. if (m_status != START_PENDING && m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
305. public void setEventsPerConnection(Integer number) //tcpeventreceiver.java
       306. if (m_status != START_PENDING && m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. private void closeStream() throws IOException //tcprecordhandler.java
       ...
153. if (m_out != null)
```

[Table of Contents] [Description]

Path 3478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. private void closeStream() throws IOException //tcprecordhandler.java
       ...
153. if (m_out != null)
```

[Table of Contents] [Description]

Path 3479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
166. if (isTracing)
```

[Table of Contents] [Description]

Path 3480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
166. if (isTracing)
```

[Table of Contents] [Description]

Path 3481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
181. if (isTracing)
```

[Table of Contents] [Description]

Path 3484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
181. if (isTracing)
```

[Table of Contents] [Description]

Path 3485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
185. if (isTracing)
```

[Table of Contents] [Description]

Path 3486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
185. if (isTracing)
```

[Table of Contents] [Description]

Path 3487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
190. if (isTracing)
```

[Table of Contents] [Description]

Path 3488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. private void newStream() throws IOException //tcprecordhandler.java
       ...
190. if (isTracing)
```

[Table of Contents] [Description]

Path 3489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void forward(char ch) throws IOException //tcprecordhandler.java
       ...
201. if (m_out != null)
```

[Table of Contents] [Description]

Path 3490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void forward(char ch) throws IOException //tcprecordhandler.java
       ...
201. if (m_out != null)
```

[Table of Contents] [Description]

Path 3491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void forward(char ch) throws IOException //tcprecordhandler.java
       ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void forward(char ch) throws IOException //tcprecordhandler.java
       ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. int next(char ch) throws IOException //tcprecordhandler.java
       ...
230. if (ch == '<')
```

[Table of Contents] [Description]

Path 3494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. int next(char ch) throws IOException //tcprecordhandler.java
       ...
237. if (ch == '/')
```

[Table of Contents] [Description]

Path 3495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. int next(char ch) throws IOException //tcprecordhandler.java
       ...
244. if (ch == 'l')
```

[Table of Contents] [Description]

Path 3496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. int next(char ch) throws IOException //tcprecordhandler.java
       ...
246. else if (Character.isLetterOrDigit(ch))
```

[Table of Contents] [Description]

Path 3497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. int next(char ch) throws IOException //tcprecordhandler.java
       ...
253. if (ch == ':')
```

[Table of Contents] [Description]

Path 3498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
251. int next(char ch) throws IOException //tcprecordhandler.java
       ...
255. else if (Character.isLetterOrDigit(ch))
```

[Table of Contents] [Description]

Path 3499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
260. int next(char ch) throws IOException //tcprecordhandler.java
       ...
262. if (ch == 'l')
```

[Table of Contents] [Description]

Path 3500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. int next(char ch) throws IOException //tcprecordhandler.java
       ...
269. if (ch == 'o')
```

[Table of Contents] [Description]

Path 3501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
274. int next(char ch) throws IOException //tcprecordhandler.java
       ...
276. if (ch == 'g')
```

[Table of Contents] [Description]

Path 3502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
313. boolean isAlive() //tcprecordhandler.java
       ...
315. if (m_context != null)
```

[Table of Contents] [Description]

Path 3503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
       ...
328. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
      ...
328. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
      ...
333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
      ...
333. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
      ...
338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. void stop() throws InterruptedException //tcprecordhandler.java
      ...
338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
363. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
363. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
379. if (!m_stop)
```

[Table of Contents] [Description]

Path 3512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
379. if (!m_stop)
```

[Table of Contents] [Description]

Path 3513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
417. if (!m_stop)
```

[Table of Contents] [Description]

Path 3518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
417. if (!m_stop)
```

[Table of Contents] [Description]

Path 3519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
446. if (m_out != null)
```

[Table of Contents] [Description]

Path 3520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
446. if (m_out != null)
```

[Table of Contents] [Description]

Path 3521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
449. if (!m_stop)
```

[Table of Contents] [Description]

Path 3522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
449. if (!m_stop)
```

[Table of Contents] [Description]

Path 3523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
      ...
455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
174. if (isTracing)
```

[Table of Contents] [Description]

Path 3526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
174. if (isTracing)
```

[Table of Contents] [Description]

Path 3527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
181. if (isTracing)
```

[Table of Contents] [Description]

Path 3528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
181. if (isTracing)
```

[Table of Contents] [Description]

Path 3529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
187. if (isTracing)
```

[Table of Contents] [Description]

Path 3530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
187. if (isTracing)
```

[Table of Contents] [Description]

Path 3531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
205. if (isTracing)
```

[Table of Contents] [Description]

Path 3532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
205. if (isTracing)
```

[Table of Contents] [Description]

Path 3533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
210. if (isTracing)
```

[Table of Contents] [Description]

Path 3534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
210. if (isTracing)
```

[Table of Contents] [Description]

Path 3535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
217. if (isTracing)
```

[Table of Contents] [Description]

Path 3536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
170. void stop() throws InterruptedException //tcpserver.java
       ...
217. if (isTracing)
```

[Table of Contents] [Description]

Path 3537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. boolean isAlive() //tcpserver.java
       ...
226. if (m_context != null)
```

[Table of Contents] [Description]

Path 3538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
256. if (isTracing)
```

[Table of Contents] [Description]

Path 3539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
256. if (isTracing)
```

[Table of Contents] [Description]

Path 3540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
260. if (m_tcpSock != null)
```

[Table of Contents] [Description]

Path 3541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
260. if (m_tcpSock != null)
```

[Table of Contents] [Description]

Path 3542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
263. if (isTracing)
```

[Table of Contents] [Description]

Path 3543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
263. if (isTracing)
```

[Table of Contents] [Description]

Path 3544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
266. if (isTracing)
```

[Table of Contents] [Description]

Path 3545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
266. if (isTracing)
```

[Table of Contents] [Description]

Path 3546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
270. if (isTracing)
```

[Table of Contents] [Description]

Path 3547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
270. if (isTracing)
```

[Table of Contents] [Description]

Path 3548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
278. if (isTracing)
```

[Table of Contents] [Description]

Path 3549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
278. if (isTracing)
```

[Table of Contents] [Description]

Path 3550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
289. if (!m_stop)
```

[Table of Contents] [Description]

Path 3551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
289. if (!m_stop)
```

[Table of Contents] [Description]

Path 3552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
292. if (isTracing)
```

[Table of Contents] [Description]

Path 3553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
292. if (isTracing)
```

[Table of Contents] [Description]

Path 3554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
309. if (isTracing && !ioInterrupted)
```

[Table of Contents] [Description]

Path 3555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
309. if (isTracing && !ioInterrupted)
```

[Table of Contents] [Description]

Path 3556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
325. if (isTracing)
```

[Table of Contents] [Description]

Path 3557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
325. if (isTracing)
```

[Table of Contents] [Description]

Path 3558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
341. if (isTracing)
```

[Table of Contents] [Description]

Path 3559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
341. if (isTracing)
```

[Table of Contents] [Description]

Path 3560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
366. if (!t.isAlive())
```

[Table of Contents] [Description]

Path 3561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
366. if (!t.isAlive())
```

[Table of Contents] [Description]

Path 3562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
375. if (isTracing)
```

[Table of Contents] [Description]

Path 3563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
375. if (isTracing)
```

[Table of Contents] [Description]

Path 3564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
385. if (isTracing)
```

[Table of Contents] [Description]

Path 3565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
385. if (isTracing)
```

[Table of Contents] [Description]

Path 3566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. boolean isAlive() //tcpstreamhandler.java
       ...
145. if (m_context != null)
```

[Table of Contents] [Description]

Path 3567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. void stop() throws InterruptedException //tcpstreamhandler.java
       ...
158. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. void stop() throws InterruptedException //tcpstreamhandler.java
       ...
158. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. void stop() throws InterruptedException //tcpstreamhandler.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. void stop() throws InterruptedException //tcpstreamhandler.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
      ...
190. if (isTracing)
```

[Table of Contents] [Description]

Path 3572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
      ...
190. if (isTracing)
```

[Table of Contents] [Description]

Path 3573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
      ...
195. if (isTracing)
```

[Table of Contents] [Description]

Path 3574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
195. if (isTracing)
```

[Table of Contents] [Description]

Path 3575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
199. if (isTracing)
```

[Table of Contents] [Description]

Path 3576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
199. if (isTracing)
```

[Table of Contents] [Description]

Path 3577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
254. if (o instanceof Throwable)
```

[Table of Contents] [Description]

Path 3578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
289. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
289. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
       ...
354. if (de1 != null && de2 != null)
```

[Table of Contents] [Description]

Path 3581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
       ...
356. else if (de1 == null && de2 != null)
```

[Table of Contents] [Description]

Path 3582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
       ...
358. else if (de1 != null && de2 == null)
```

[Table of Contents] [Description]

Path 3583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
395. if (isTracing)
```

[Table of Contents] [Description]

Path 3584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
395. if (isTracing)
```

[Table of Contents] [Description]

Path 3585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
485. if (isTracing)
```

[Table of Contents] [Description]

Path 3586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
485. if (isTracing)
```

[Table of Contents] [Description]

Path 3587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
490. if (isTracing)
```

[Table of Contents] [Description]

Path 3588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
490. if (isTracing)
```

[Table of Contents] [Description]

Path 3589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
499. if (isTracing)
```

[Table of Contents] [Description]

Path 3590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
499. if (isTracing)
```

[Table of Contents] [Description]

Path 3591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
504. if (isTracing)
```

[Table of Contents] [Description]

Path 3592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
504. if (isTracing)
```

[Table of Contents] [Description]

Path 3593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
510. if (isTracing)
```

[Table of Contents] [Description]

Path 3594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
510. if (isTracing)
```

[Table of Contents] [Description]

Path 3595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
156. public synchronized void start() //udpeventreceiver.java
157. if (m_status != START_PENDING)
```

[Table of Contents] [Description]

Path 3596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. public synchronized void stop() //udpeventreceiver.java
198. if (m_status == STOPPED)
```

[Table of Contents] [Description]

Path 3597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. public void setPort(Integer port) //udpeventreceiver.java
236. if (m_status == STARTING || m_status == RUNNING || m_status ==
STOP_PENDING)
```

[Table of Contents] [Description]

Path 3598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public void addEventHandler(EventHandler handler) //udpeventreceiver.java
       ...
256. if (!m_handlers.contains(handler))
```

[Table of Contents] [Description]

Path 3599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public void addEventHandler(EventHandler handler) //udpeventreceiver.java
       ...
256. if (!m_handlers.contains(handler))
```

[Table of Contents] [Description]

Path 3600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. void stop() throws InterruptedException //udpprocessor.java
       ...
118. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. void stop() throws InterruptedException //udpprocessor.java
       ...
118. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. void stop() throws InterruptedException //udpprocessor.java
       ...
124. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
114. void stop() throws InterruptedException //udpprocessor.java
       ...
124. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
144. if (isTracing)
```

[Table of Contents] [Description]

Path 3605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
144. if (isTracing)
```

[Table of Contents] [Description]

Path 3606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
147. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
147. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
154. if (isTracing)
```

[Table of Contents] [Description]

Path 3609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
      |
154. if (isTracing)
```

[Table of Contents] [Description]

Path 3610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
      |
166. if (isTracing)
```

[Table of Contents] [Description]

Path 3611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
      |
166. if (isTracing)
```

[Table of Contents] [Description]

Path 3612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
      |
172. if (isTracing)
```

[Table of Contents] [Description]

Path 3613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
      |
172. if (isTracing)
```

[Table of Contents] [Description]

Path 3614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
180. if (isTracing)
```

[Table of Contents] [Description]

Path 3615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
180. if (isTracing)
```

[Table of Contents] [Description]

Path 3616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
201. if (isTracing)
```

[Table of Contents] [Description]

Path 3617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
201. if (isTracing)
```

[Table of Contents] [Description]

Path 3618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |
      ...
238. if (isTracing)
```

[Table of Contents] [Description]

Path 3619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |...
      238. if (isTracing)
```

[Table of Contents] [Description]

Path 3620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |...
      248. if (isTracing)
```

[Table of Contents] [Description]

Path 3621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
132. public void run() //udpprocessor.java
      |...
      248. if (isTracing)
```

[Table of Contents] [Description]

Path 3622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
165. void ackEvent(Event e) //udpreceivedevent.java
      |166. if (!m_ackEvents.contains(e))
```

[Table of Contents] [Description]

Path 3623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
165. void ackEvent(Event e) //udpreceivedevent.java
      |166. if (!m_ackEvents.contains(e))
```

[Table of Contents] [Description]

Path 3624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. void stop() throws InterruptedException //udpreceiver.java
      ...
99. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. void stop() throws InterruptedException //udpreceiver.java
      ...
99. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. void stop() throws InterruptedException //udpreceiver.java
      ...
105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. void stop() throws InterruptedException //udpreceiver.java
      ...
105. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
132. if (isTracing)
```

[Table of Contents] [Description]

Path 3629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
132. if (isTracing)
```

[Table of Contents] [Description]

Path 3630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
135. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
135. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
148. if (isTracing)
```

[Table of Contents] [Description]

Path 3633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
148. if (isTracing)
```

[Table of Contents] [Description]

Path 3634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
       ...
160. if (isTracing)
```

[Table of Contents] [Description]

Path 3635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
       ...
160. if (isTracing)
```

[Table of Contents] [Description]

Path 3636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
       ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
       ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3638:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
       ...
183. if (isTracing && !ioInterrupted)
```

[Table of Contents] [Description]

Path 3639:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
183. if (isTracing && !ioInterrupted)
```

[Table of Contents] [Description]

Path 3640:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
197. if (isTracing)
```

[Table of Contents] [Description]

Path 3641:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
197. if (isTracing)
```

[Table of Contents] [Description]

Path 3642:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
213. if (isTracing)
```

[Table of Contents] [Description]

Path 3643:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void run() //udpreceiver.java
      ...
213. if (isTracing)
```

[Table of Contents] [Description]

Path 3644:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. void stop() throws InterruptedException //udpuuidsender.java
      |...
      |119. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3645:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. void stop() throws InterruptedException //udpuuidsender.java
      |...
      |119. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3646:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. void stop() throws InterruptedException //udpuuidsender.java
      |...
      |125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3647:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. void stop() throws InterruptedException //udpuuidsender.java
      |...
      |125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3648:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |149. if (isTracing)
```

[Table of Contents] [Description]

Path 3649:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |149. if (isTracing)
```

[Table of Contents] [Description]

Path 3650:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |152. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3651:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |152. } else if (isTracing)
```

[Table of Contents] [Description]

Path 3652:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |162. if (isTracing)
```

[Table of Contents] [Description]

Path 3653:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      |...
      |162. if (isTracing)
```

[Table of Contents] [Description]

Path 3654:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3655:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
176. if (isTracing)
```

[Table of Contents] [Description]

Path 3656:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
211. if (isTracing)
```

[Table of Contents] [Description]

Path 3657:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
211. if (isTracing)
```

[Table of Contents] [Description]

Path 3658:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
      ...
267. if (isTracing)
```

[Table of Contents] [Description]

Path 3659:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public void run() //udpuuidsender.java
        ...
267. if (isTracing)
```

[Table of Contents] [Description]

Path 3660:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. private org.opennms.netmgt.xml.eventconf.Event getMatchInKeyList(List keylist,
org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
        ...
123. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3661:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. private org.opennms.netmgt.xml.eventconf.Event getMatchInKeyList(List keylist,
org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
        ...
123. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3662:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. private boolean eventValuePassesMaskValue(String eventvalue, List
maskValues) //eventconfdata.java
        ...
179. if (eventvalue.startsWith(keyvalue.substring(0, len - 1)))
```

[Table of Contents] [Description]

Path 3663:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. private void updateUeiToKeyListMap(EventKey eventKey,
org.opennms.netmgt.xml.eventconf.Event event) //eventconfdata.java
        ...
200. if (!keylist.contains(eventKey))
```

[Table of Contents] [Description]

Path 3664:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. private void updateUeiToKeyListMap(EventKey eventKey,
org.opennms.netmgt.xml.eventconf.Event event) //eventconfdata.java
      ...
200. if (!keylist.contains(eventKey))
```

[Table of Contents] [Description]

Path 3665:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized org.opennms.netmgt.xml.eventconf.Event getEvent
(org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
      ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3666:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized org.opennms.netmgt.xml.eventconf.Event getEvent
(org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
      ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3667:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized org.opennms.netmgt.xml.eventconf.Event getEvent
(org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
      ...
328. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3668:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
281. public synchronized org.opennms.netmgt.xml.eventconf.Event getEvent
(org.opennms.netmgt.xml.event.Event event) //eventconfdata.java
      ...
328. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3669:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
310. public void evaluateHashCode() //eventkey.java
       ...
313. if (isEmpty())
```

[Table of Contents] [Description]

Path 3670:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
343. public int compareTo(Object o) //eventkey.java
344. if (!(o instanceof EventKey))
```

[Table of Contents] [Description]

Path 3671:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
357. public int hashCode() //eventkey.java
358. if (m_hashCode != -1111)
```

[Table of Contents] [Description]

Path 3672:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. private int evaluateHashCode(String value) //eventmaskvaluelist.java
       ...
230. if (value == null)
```

[Table of Contents] [Description]

Path 3673:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. private int evaluateHashCode(String value) //eventmaskvaluelist.java
       ...
239. if (tc != '.')
```

[Table of Contents] [Description]

Path 3674:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. private int evaluateHashCode(String value) //eventmaskvaluelist.java
     ...
239. if (tc != '.')
```

[Table of Contents] [Description]

Path 3675:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. private int evaluateHashCode(String value) //eventmaskvaluelist.java
     ...
253. if (i != 0)
```

[Table of Contents] [Description]

Path 3676:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public void evaluateHashCode() //eventmaskvaluelist.java
     ...
274. if (obj instanceof String)
```

[Table of Contents] [Description]

Path 3677:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. public int hashCode() //eventmaskvaluelist.java
     288. if (m_hashCode != -1111)
```

[Table of Contents] [Description]

Path 3678:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
78. public static String format(Autoaction[] autoacts, int sz) //autoaction.java
    ...
83. if (!first)
```

[Table of Contents] [Description]

Path 3679:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
78. public static String format(Autoaction[] autoacts, int sz) //autoaction.java
      ...
83. if (!first)
```

[Table of Contents] [Description]

Path 3680:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public static String format(List strings, int maxlen) //constants.java
      183. if (strings == null)
```

[Table of Contents] [Description]

Path 3681:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public static String format(List strings, int maxlen) //constants.java
      ...
193. if (!first)
```

[Table of Contents] [Description]

Path 3682:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public static String format(List strings, int maxlen) //constants.java
      ...
193. if (!first)
```

[Table of Contents] [Description]

Path 3683:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public static String format(String[] strings, int maxlen) //constants.java
      223. if (strings == null || strings.length <= 0)
```

[Table of Contents] [Description]

Path 3684:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public static String format(String[] strings, int maxlen) //constants.java
     |...
     |232. if (!first)
```

[Table of Contents] [Description]

Path 3685:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public static String format(String[] strings, int maxlen) //constants.java
     |...
     |232. if (!first)
```

[Table of Contents] [Description]

Path 3686:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
257. public static String format(String string, int maxlen) //constants.java
     |258. if (string == null)
```

[Table of Contents] [Description]

Path 3687:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
     |...
     |282. if (sev.equalsIgnoreCase("normal"))
```

[Table of Contents] [Description]

Path 3688:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
     |...
     |284. else if (sev.equalsIgnoreCase("warning"))
```

[Table of Contents] [Description]

Path 3689:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
```

```
...
286. else if (sev.equalsIgnoreCase("minor"))
```

[Table of Contents] [Description]

Path 3690:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
...
288. else if (sev.equalsIgnoreCase("major"))
```

[Table of Contents] [Description]

Path 3691:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
...
290. else if (sev.equalsIgnoreCase("critical"))
```

[Table of Contents] [Description]

Path 3692:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public static int getSeverity(String sev) //constants.java
...
292. else if (sev.equalsIgnoreCase("cleared"))
```

[Table of Contents] [Description]

Path 3693:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
78. public static String format(org.opennms.netmgt.xml.event.Forward[] forwards, int
sz) //forward.java
...
83. if (!first)
```

[Table of Contents] [Description]

Path 3694:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
78. public static String format(org.opennms.netmgt.xml.event.Forward[] forwards, int
sz) //forward.java
      |
      ...
      83. if (!first)
```

[Table of Contents] [Description]

Path 3695:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
55. public static String format(Logmsg msg) //logmessage.java
      |
      ...
      60. if (fmsg.length() >= 256)
```

[Table of Contents] [Description]

Path 3696:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public static String format(List opacts, int sz) //operatoraction.java
      |
      ...
      85. if (!first)
```

[Table of Contents] [Description]

Path 3697:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public static String format(List opacts, int sz) //operatoraction.java
      |
      ...
      85. if (!first)
```

[Table of Contents] [Description]

Path 3698:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
59. public static String format(Parms parms) //parameter.java
      |
      ...
      67. if (!first)
```

[Table of Contents] [Description]

Path 3699:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
59. public static String format(Parms parms) //parameter.java
      |...
      |67. if (!first)
```

[Table of Contents] [Description]

Path 3700:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public static String format(Snmp info, int maxlen) //snmpinfo.java
      |66. if (info == null)
```

[Table of Contents] [Description]

Path 3701:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public static String format(Snmp info, int maxlen) //snmpinfo.java
      |...
      |82. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3702:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public static String format(Snmp info, int maxlen) //snmpinfo.java
      |...
      |82. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3703:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public static String format(Snmp info, int maxlen) //snmpinfo.java
      |...
      |82. if (info.hasSpecific())
```

[Table of Contents] [Description]

Path 3704:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public static String format(Snmp info, int maxlen) //snmpinfo.java
```

```
      ...
88.  if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3705:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65.  public static String format(Snmp info, int maxlen) //snmpinfo.java
      ...
88.  if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3706:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65.  public static String format(Snmp info, int maxlen) //snmpinfo.java
      ...
88.  if (info.hasGeneric())
```

[Table of Contents] [Description]

Path 3707:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65.  public static String format(Snmp info, int maxlen) //snmpinfo.java
      ...
94.  if (info.getCommunity() != null)
```

[Table of Contents] [Description]

Path 3708:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65.  public static String format(Snmp info, int maxlen) //snmpinfo.java
      ...
94.  if (info.getCommunity() != null)
```

[Table of Contents] [Description]

Path 3709:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65.  public static String format(Snmp info, int maxlen) //snmpinfo.java
```

```
...
94. if (info.getCommunity() != null)
```

[Table of Contents] [Description]

Path 3710:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3711:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
...
273. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3712:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. public void printStackTrace() //filterparseexception.java
94. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3713:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. public void printStackTrace() //filterparseexception.java
94. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3714:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void printStackTrace(PrintStream ps) //filterparseexception.java
107. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3715:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void printStackTrace(PrintStream ps) //filterparseexception.java
     |107. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3716:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void printStackTrace(PrintWriter pw) //filterparseexception.java
     |120. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3717:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
119. public void printStackTrace(PrintWriter pw) //filterparseexception.java
     |120. if (m_delegate != null)
```

[Table of Contents] [Description]

Path 3718:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. private String validateIdent(String ident) //sqltranslation.java
     |...
     |186. if (tableForIdent != null)
```

[Table of Contents] [Description]

Path 3719:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. private String validateIdent(String ident) //sqltranslation.java
     |...
     |197. if (tableForIdent != null)
```

[Table of Contents] [Description]

Path 3720:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. private String validateIdent(String ident) //sqltranslation.java
      |
      ...
      199. if (expr != null)
```

[Table of Contents] [Description]

Path 3721:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. private String validateIdent(String ident) //sqltranslation.java
      |
      ...
      211. if (tableForIdent != null)
```

[Table of Contents] [Description]

Path 3722:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
183. private String validateIdent(String ident) //sqltranslation.java
      |
      ...
      213. if (expr != null)
```

[Table of Contents] [Description]

Path 3723:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. private String addColumnToStatement(String colName) //sqltranslation.java
      |
      ...
      237. if (t == null)
```

[Table of Contents] [Description]

Path 3724:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private String addColumnToStatement(Table t, String
colName) //sqltranslation.java
      |
      ...
      265. if (joinTable == null)
```

[Table of Contents] [Description]

Path 3725:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private String addColumnToStatement(Table t, String
colName) //sqltranslation.java
      |
      ...
      268. if (m_tables.size() == 0)
```

[Table of Contents] [Description]

Path 3726:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private String addColumnToStatement(Table t, String
colName) //sqltranslation.java
      |
      ...
      268. if (m_tables.size() == 0)
```

[Table of Contents] [Description]

Path 3727:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private String addColumnToStatement(Table t, String
colName) //sqltranslation.java
      |
      ...
      268. if (m_tables.size() == 0)
```

[Table of Contents] [Description]

Path 3728:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
307. public void checkIPNum(String octet) //sqltranslation.java
      |
      ...
      310. if (ipnum < 0 || ipnum > 255)
```

[Table of Contents] [Description]

Path 3729:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
service) //sqltranslation.java
      |
      ...
      394. if (needAnd)
```

[Table of Contents] [Description]

Path 3730:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
service) //sqltranslation.java
     |...
     394. if (needAnd)
```

[Table of Contents] [Description]

Path 3731:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
service) //sqltranslation.java
     |...
     402. if (needAnd)
```

[Table of Contents] [Description]

Path 3732:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
service) //sqltranslation.java
     |...
     402. if (needAnd)
```

[Table of Contents] [Description]

Path 3733:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
service) //sqltranslation.java
     |...
     410. if (needAnd)
```

[Table of Contents] [Description]

Path 3734:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
383. public void setConstraintTranslation(long nodeId, String ipaddr, String
```

```
service) //sqltranslation.java
    ...
410. if (needAnd)
```

[Table of Contents] [Description]

Path 3735:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
696. public String getStatement() //sqltranslation.java
    ...
698. if (m_node == null)
```

[Table of Contents] [Description]

Path 3736:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public synchronized Collection waitForAnticipated(long
millis) //eventanticipator.java
    ...
111. if (m_anticipatedEvents.isEmpty())
```

[Table of Contents] [Description]

Path 3737:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int getPollStatus() //mockcontainer.java
    ...
108. if (element.getPollStatus() == ServiceMonitor.SERVICE_AVAILABLE)
```

[Table of Contents] [Description]

Path 3738:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void drop() //mockdatabase.java
246. if (m_server != null)
```

[Table of Contents] [Description]

Path 3739:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public void drop() //mockdatabase.java
     |246. if (m_server != null)
```

[Table of Contents] [Description]

Path 3740:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
359. public Integer getServiceID(String serviceName) //mockdatabase.java
     |360. if (serviceName == null) return new Integer(-1);
```

[Table of Contents] [Description]

Path 3741:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
436. public void writeEvent(Event e) //mockdatabase.java
     |...
     |439. if (e.getCreationTime() == null)
```

[Table of Contents] [Description]

Path 3742:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
436. public void writeEvent(Event e) //mockdatabase.java
     |...
     |439. if (e.getCreationTime() == null)
```

[Table of Contents] [Description]

Path 3743:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. private boolean eventMatches(Event e) //mockeventipcmanager.java
     |69. if (m_ueiList == null)
```

[Table of Contents] [Description]

Path 3744:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. public void visitService(MockService svc) //mockinterface.java
     |...
```

```
104. if (svc.getPollStatus() == ServiceMonitor.SERVICE_AVAILABLE)
```

[Table of Contents] [Description]

Path 3745:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
401. public void testPollerConfig() //mocknetworktest.java
     ...
430. if ("ICMP".equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3746:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
401. public void testPollerConfig() //mocknetworktest.java
     ...
430. if ("ICMP".equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3747:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
401. public void testPollerConfig() //mocknetworktest.java
     ...
432. else if ("HTTP".equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3748:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
401. public void testPollerConfig() //mocknetworktest.java
     ...
432. else if ("HTTP".equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3749:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
401. public void testPollerConfig() //mocknetworktest.java
```

```
...
432. else if ("HTTP".equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3750:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. public void assertSuccess() throws Throwable //mocknetworktest.java
     620. if (m_t != null)
```

[Table of Contents] [Description]

Path 3751:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
706. public void assertSuccess() throws Throwable //mocknetworktest.java
     707. if (m_t != null)
```

[Table of Contents] [Description]

Path 3752:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public void addDowntime(long interval, long begin, long end, boolean
delete) //mockpollerconfig.java
     ...
     90. if (end >= 0)
```

[Table of Contents] [Description]

Path 3753:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
85. public void addDowntime(long interval, long begin, long end, boolean
delete) //mockpollerconfig.java
     ...
     90. if (end >= 0)
```

[Table of Contents] [Description]

Path 3754:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. private void addServiceMonitor(String name, ServiceMonitor
```

```
monitor) //mockpollerconfig.java
      165. if (!hasServiceMonitor(name))
```

[Table of Contents] [Description]

Path 3755:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. private void addServiceMonitor(String name, ServiceMonitor
monitor) //mockpollerconfig.java
      165. if (!hasServiceMonitor(name))
```

[Table of Contents] [Description]

Path 3756:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
200. private Service findService(Package pkg, String
svcName) //mockpollerconfig.java
      ...
      204. if (svcName.equals(svc.getName()))
```

[Table of Contents] [Description]

Path 3757:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public Package getPackage(String name) //mockpollerconfig.java
      ...
      225. if (pkg.getName().equals(name))
```

[Table of Contents] [Description]

Path 3758:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
264. public boolean interfaceInPackage(String iface, Package
pkg) //mockpollerconfig.java
      ...
      268. if (ipAddr.equals(iface))
```

[Table of Contents] [Description]

Path 3759:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
307. public boolean serviceInPackageAndEnabled(String svcName, Package
pkg) //mockpollerconfig.java
      |...
      |311. if (svc.getName().equals(svcName))
```

[Table of Contents] [Description]

Path 3760:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
351. public void setPollInterval(Package pkg, String svcName, long
interval) //mockpollerconfig.java
      |...
      |353. if (svc == null)
```

[Table of Contents] [Description]

Path 3761:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. public static boolean eventsMatch(Event e1, Event e2) //mockutil.java
      |195. if (e1 == e2)
```

[Table of Contents] [Description]

Path 3762:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. public static boolean eventsMatch(Event e1, Event e2) //mockutil.java
      |...
      |197. if (e1 == null || e2 == null)
```

[Table of Contents] [Description]

Path 3763:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
227. public static void printEvent(String prefix, Event event) //mockutil.java
      |228. if (!printEnabled()) return;
```

[Table of Contents] [Description]

Path 3764:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
227. public static void printEvent(String prefix, Event event) //mockutil.java
       ...
       230. if (prefix == null)
```

[Table of Contents] [Description]

Path 3765:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. public static void printEvents(String prefix, Collection
events) //mockutil.java
       236. if (!printEnabled()) return ;
```

[Table of Contents] [Description]

Path 3766:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. public static void println(String string) //mockutil.java
       262. if (MockUtil.printEnabled())
```

[Table of Contents] [Description]

Path 3767:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. public static void println(String string) //mockutil.java
       262. if (MockUtil.printEnabled())
```

[Table of Contents] [Description]

Path 3768:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
80. public boolean equals(Object o) //outage.java
       81. if (! (o instanceof Outage)) return false;
```

[Table of Contents] [Description]

Path 3769:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. private synchronized boolean anticipates(Map pending, MockService
svc) //outageanticipator.java
```

```
...
112. if (outage.isForService(svc))
```

[Table of Contents] [Description]

Path 3770:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
134. protected synchronized void removeFromOutageList(Map outageMap, Event
outageEvent, Outage outage) //outageanticipator.java
...
137. if (list == null) return;
```

[Table of Contents] [Description]

Path 3771:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
209. public synchronized boolean checkAnticipated() //outageanticipator.java
...
217. if (m_pendingOpens.size() != 0 || m_pendingCloses.size() != 0)
```

[Table of Contents] [Description]

Path 3772:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public synchronized void poll(MockService service) //pollanticipator.java
...
81. if (m_anticipatedPolls.contains(service))
```

[Table of Contents] [Description]

Path 3773:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public synchronized void poll(MockService service) //pollanticipator.java
...
81. if (m_anticipatedPolls.contains(service))
```

[Table of Contents] [Description]

Path 3774:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public synchronized void poll(MockService service) //pollanticipator.java
      ...
81. if (m_anticipatedPolls.contains(service))
```

[Table of Contents] [Description]

Path 3775:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public synchronized void poll(MockService service) //pollanticipator.java
      ...
86. if (m_anticipatedPolls.isEmpty())
```

[Table of Contents] [Description]

Path 3776:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public synchronized void poll(MockService service) //pollanticipator.java
      ...
86. if (m_anticipatedPolls.isEmpty())
```

[Table of Contents] [Description]

Path 3777:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public synchronized Collection waitForAnticipated(long
millis) //pollanticipator.java
      ...
121. if (m_anticipatedPolls.isEmpty())
```

[Table of Contents] [Description]

Path 3778:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public void onEvent(Event event) //broadcasteventprocessor.java
160. if (event == null)
```

[Table of Contents] [Description]

Path 3779:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3780:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3781:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      |...
      602. if (finalTextMessage == null)
```

[Table of Contents] [Description]

Path 3782:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      |...
      602. if (finalTextMessage == null)
```

[Table of Contents] [Description]

Path 3783:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      |...
      602. if (finalTextMessage == null)
```

[Table of Contents] [Description]

Path 3784:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      ...
608. if (finalNumericMessage == null)
```

[Table of Contents] [Description]

Path 3785:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      ...
608. if (finalNumericMessage == null)
```

[Table of Contents] [Description]

Path 3786:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      ...
608. if (finalNumericMessage == null)
```

[Table of Contents] [Description]

Path 3787:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      ...
614. if (finalSubjectLine == null)
```

[Table of Contents] [Description]

Path 3788:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
      ...
614. if (finalSubjectLine == null)
```

[Table of Contents] [Description]

Path 3789:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
575. Map buildParameterMap(Notification notification, Event event, int
noticeId) //broadcasteventprocessor.java
     ...
614. if (finalSubjectLine == null)
```

[Table of Contents] [Description]

Path 3790:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
822. public String scheduledOutage(long nodeId, String
theInterface) //broadcasteventprocessor.java
     ...
843. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3791:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
822. public String scheduledOutage(long nodeId, String
theInterface) //broadcasteventprocessor.java
     ...
843. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3792:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public void run() //defaultqueuehandler.java
     ...
127. if (m_status == STOP_PENDING || m_status == STOPPED)
```

[Table of Contents] [Description]

Path 3793:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
118. public void run() //defaultqueuehandler.java
     ...
144. if (m_status == RESUME_PENDING)
```

[Table of Contents] [Description]

Path 3794:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public synchronized void stop() //defaultqueuehandler.java
     |230. if (m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3795:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. Growl( String inAppName, NSData inImageData, NSArray inAllNotes, //growl.java
     |...
     |159. if ( registerNow )
```

[Table of Contents] [Description]

Path 3796:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. Growl( String inAppName, NSData inImageData, NSArray inAllNotes, //growl.java
     |...
     |159. if ( registerNow )
```

[Table of Contents] [Description]

Path 3797:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public synchronized void stop() //notifd.java
     |...
     |253. if (m_eventReader != null)
```

[Table of Contents] [Description]

Path 3798:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public synchronized void stop() //notifd.java
     |...
     |253. if (m_eventReader != null)
```

[Table of Contents] [Description]

Path 3799:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
283. public synchronized void pause() //notifd.java
     |284. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3800:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. public synchronized void resume() //notifd.java
     |302. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 3801:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
66. protected XMPPNotificationManager() //xmppnotificationmanager.java
     |...
     |85. if (this.props.containsKey("xmpp.useSSL")
```

[Table of Contents] [Description]

Path 3802:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. public synchronized Collection waitForAnticipated(long
millis) //notificationanticipator.java
     |...
     |145. if (m_anticipated.isEmpty())
```

[Table of Contents] [Description]

Path 3803:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
     |...
     |129. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3804:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
      ...
129. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3805:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
      ...
139. if (log.isDebugEnabled()) log.debug("handleInterfaceDown: " + svc + "
already down");
```

[Table of Contents] [Description]

Path 3806:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicinterface.java
      ...
139. if (log.isDebugEnabled()) log.debug("handleInterfaceDown: " + svc + "
already down");
```

[Table of Contents] [Description]

Path 3807:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
      ...
124. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3808:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
```

```
      ...
124.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3809:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
      ...
133.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3810:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. public void openOutages(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicnode.java
      ...
133.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3811:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. public boolean openOutage(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicservice.java
      ...
129.  if (openOutageExists(dbConn))
```

[Table of Contents] [Description]

Path 3812:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. public boolean openOutage(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicservice.java
      ...
137.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3813:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. public boolean openOutage(Connection dbConn, long eventID, String eventTime)
throws SQLException //basicservice.java
     ...
137. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3814:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
50. static public boolean countQueryIsPositive(PreparedStatement openStmt) throws
SQLException //dbutil.java
     ...
51. if (DbUtil.executeCountQuery(openStmt) > 0)
```

[Table of Contents] [Description]

Path 3815:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public static void close(Connection dbConn) //dbutil.java
     ...
77. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3816:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public static void close(Connection dbConn) //dbutil.java
     ...
77. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3817:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. public static void rollback(Connection dbConn, String msg, SQLException
se) //dbutil.java
     ...
89. if (dbConn != null) dbConn.rollback();
```

[Table of Contents] [Description]

Path 3818:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
84. public static void rollback(Connection dbConn, String msg, SQLException
se) //dbutil.java
    ...
89. if (dbConn != null) dbConn.rollback();
```

[Table of Contents] [Description]

Path 3819:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public static void commit(Connection dbConn, String successMsg, String
failMsg) //dbutil.java
    ...
100. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3820:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public static void commit(Connection dbConn, String successMsg, String
failMsg) //dbutil.java
    ...
100. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3821:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
    ...
162. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3822:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
    ...
162. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3823:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
     ...
203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3824:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
     ...
203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3825:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
     ...
243. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3826:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public OutageEventEntry findCacheMatch(long eventId, long nodeId, String
ipAddr, long serviceId, String eventTime, int type) //outageeventcache.java
     ...
243. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3827:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
```

```
      ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3828:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3829:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3830:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3831:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
170. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3832:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
      170. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3833:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
      177. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3834:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. private void closeOutages(java.sql.Connection dbConn) throws
SQLException //outagemanager.java
      ...
      177. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3835:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
      ...
      216. if (m_eventMgr == null)
```

[Table of Contents] [Description]

Path 3836:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
      ...
      219. if (m_outageMgrConfig == null)
```

[Table of Contents] [Description]

Path 3837:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
       ...
256. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3838:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
       ...
256. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3839:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
       ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3840:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public void init() //outagemanager.java
       ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3841:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. public void start() //outagemanager.java
       ...
321. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3842:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. public void start() //outagemanager.java
       ...
321. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3843:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. public void pause() //outagemanager.java
340. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3844:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. public void pause() //outagemanager.java
       ...
349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3845:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. public void pause() //outagemanager.java
       ...
349. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3846:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. public void resume() //outagemanager.java
357. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 3847:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. public void resume() //outagemanager.java
       ...
366. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3848:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
356. public void resume() //outagemanager.java
       ...
366. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3849:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3850:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3851:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3852:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3853:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3854:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public void stop() //outagemanager.java
       ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3855:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. public void start() //outagemgreventprocessor.java
       ...
90. if (m_eventListenerRegistered) return;
```

[Table of Contents] [Description]

Path 3856:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
       145. if (event == null)
```

[Table of Contents] [Description]

Path 3857:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
     |...
     149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3858:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
     |...
     149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3859:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
     |...
     154. if (uei == null)
```

[Table of Contents] [Description]

Path 3860:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
     |...
     159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3861:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public void onEvent(Event event) //outagemgreventprocessor.java
     |...
     159. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3862:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
133. OutageChecker(MockService svc, Event lostSvcEvent, Event
regainedSvcEvent) //outagetest.java
      |...
      |140. if (m_regainedSvcEvent != null)
```

[Table of Contents] [Description]

Path 3863:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. private Integer getRegainedEventId() //outagetest.java
      |153. if (m_regainedSvcEvent == null)
```

[Table of Contents] [Description]

Path 3864:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
      |...
      |140. if (name == null)
```

[Table of Contents] [Description]

Path 3865:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
      |...
      |146. if (id != -1)
```

[Table of Contents] [Description]

Path 3866:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
      |...
      |170. if (serviceStmt != null)
```

[Table of Contents] [Description]

Path 3867:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
     ...
170. if (serviceStmt != null)
```

[Table of Contents] [Description]

Path 3868:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
     ...
174. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3869:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
     ...
174. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3870:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
     ...
183. if (id != -1)
```

[Table of Contents] [Description]

Path 3871:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. private long getServiceID(String name) throws SQLException //outagewriter.java
     ...
183. if (id != -1)
```

[Table of Contents] [Description]

Path 3872:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |...
      |396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3873:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |...
      |396. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3874:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
393. private void handleInterfaceReparented(String ipAddr, Parms
eventParms) //outagewriter.java
      |...
      |416. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3875:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
506. private Event createEvent(String uei, java.util.Date eventDate, long nodeID,
String ipAddr, String serviceName) //outagewriter.java
      |...
      |516. if (ipAddr != null)
```

[Table of Contents] [Description]

Path 3876:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
506. private Event createEvent(String uei, java.util.Date eventDate, long nodeID,
String ipAddr, String serviceName) //outagewriter.java
      |...
      |516. if (ipAddr != null)
```

[Table of Contents] [Description]

Path 3877:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
506. private Event createEvent(String uei, java.util.Date eventDate, long nodeID,
String ipAddr, String serviceName) //outagewriter.java
      ...
519. if (serviceName != null)
```

[Table of Contents] [Description]

Path 3878:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
506. private Event createEvent(String uei, java.util.Date eventDate, long nodeID,
String ipAddr, String serviceName) //outagewriter.java
      ...
519. if (serviceName != null)
```

[Table of Contents] [Description]

Path 3879:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
      ...
536. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3880:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
      ...
536. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3881:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
      ...
541. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3882:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       |
       ...
541. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3883:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       |
       ...
550. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3884:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       |
       ...
550. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3885:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       |
       ...
557. if (m_event.hasDbid())
```

[Table of Contents] [Description]

Path 3886:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       |
       ...
562. if (m_event.hasNodeid())
```

[Table of Contents] [Description]

Path 3887:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       ...
569. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3888:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
532. private void processEvent() //outagewriter.java
       ...
569. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3889:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. static private long byteToLong(byte b) //packet.java
       ...
106. if (r < 0)
```

[Table of Contents] [Description]

Path 3890:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public final void computeChecksum() //packet.java
       ...
261. if ((m_pad.length % 1) == 1)
```

[Table of Contents] [Description]

Path 3891:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. public final void computeChecksum() //packet.java
       ...
261. if ((m_pad.length % 1) == 1)
```

[Table of Contents] [Description]

Path 3892:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. public final int loadFromBuffer(byte[] buf, int offset) //packet.java
     |302. if ((buf.length - offset) < getNetworkSize())
```

[Table of Contents] [Description]

Path 3893:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. public final int loadFromBuffer(byte[] buf, int offset) //packet.java
     |...
     |306. if (!isEchoReply() && !isEchoRequest())
```

[Table of Contents] [Description]

Path 3894:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
301. public final int loadFromBuffer(byte[] buf, int offset) //packet.java
     |...
     |333. if (m_pad == null)
```

[Table of Contents] [Description]

Path 3895:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
357. public final int storeToBuffer(byte[] buf, int offset) //packet.java
     |358. if ((buf.length - offset) < getNetworkSize())
```

[Table of Contents] [Description]

Path 3896:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. static Reply create(DatagramPacket packet) //reply.java
     |...
     |150. if (pkt.getReceivedTime() == 0)
```

[Table of Contents] [Description]

Path 3897:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. static Reply create(DatagramPacket packet) //reply.java
      ...
150. if (pkt.getReceivedTime() == 0)
```

[Table of Contents] [Description]

Path 3898:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public final synchronized void start() //replyreceiver.java
      232. if (m_worker != null)
```

[Table of Contents] [Description]

Path 3899:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
249. public final synchronized void stop() //replyreceiver.java
      250. if (m_worker == null)
```

[Table of Contents] [Description]

Path 3900:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
249. public final synchronized void stop() //replyreceiver.java
      ...
254. if (m_status != STOPPED)
```

[Table of Contents] [Description]

Path 3901:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
268. public final synchronized void pause() //replyreceiver.java
      269. if (m_worker == null || !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3902:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
278. public final synchronized void resume() //replyreceiver.java
     279. if (m_worker == null || !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 3903:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. public final synchronized int getStatus() //replyreceiver.java
     299. if (m_status == RUNNING && m_paused)
```

[Table of Contents] [Description]

Path 3904:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public final void run() //replyreceiver.java
     ...
     319. if (m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 3905:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public final void run() //replyreceiver.java
     ...
     326. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3906:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public final void run() //replyreceiver.java
     ...
     326. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3907:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
171. public void onEvent(Event e) //defaultpollcontext.java
     |
```

```
...
182. if (pollEvent.isPending()) break;
```

[Table of Contents] [Description]

Path 3908:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public boolean activeServiceExists(String whichEvent, int nodeId, String ipAddr,
String serviceName) //defaultquerymanager.java
...
116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3909:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public boolean activeServiceExists(String whichEvent, int nodeId, String ipAddr,
String serviceName) //defaultquerymanager.java
...
116. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3910:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
...
169. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 3911:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
...
169. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 3912:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
     ...
177. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 3913:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
     ...
177. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 3914:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
     ...
185. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 3915:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
     ...
185. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 3916:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. public List getActiveServiceIdsForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
     ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3917:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
197. public List getActiveServiceIdsForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3918:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3919:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3920:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      ...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3921:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
```

```
...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3922:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
...
257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3923:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
...
257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3924:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
...
286. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3925:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
...
286. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3926:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
      ...
295. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3927:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
      ...
295. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3928:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
      ...
305. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3929:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
      ...
305. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3930:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
319. public int getServiceCountForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      ...
332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3931:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
319. public int getServiceCountForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
      |
      ...
      332. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3932:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public List getInterfacesWithService(String svcName) throws
SQLException //defaultquerymanager.java
      |
      ...
      352. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3933:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public List getInterfacesWithService(String svcName) throws
SQLException //defaultquerymanager.java
      |
      ...
      352. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3934:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
379. public Date getServiceLostDate(int nodeId, String ipAddr, String svcName, int
serviceId) //defaultquerymanager.java
      |
      ...
      432. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3935:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
379. public Date getServiceLostDate(int nodeId, String ipAddr, String svcName, int
serviceId) //defaultquerymanager.java
      |
```

```
...
432. if (dbConn != null)
```

[Table of Contents] [Description]

Path 3936:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3937:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3938:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
...
242. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 3939:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
...
242. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 3940:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
```

```
...
251. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3941:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
228. public synchronized void start() //poller.java
...
251. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3942:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public synchronized void stop() //poller.java
...
268. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3943:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public synchronized void stop() //poller.java
...
268. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3944:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. public synchronized void pause() //poller.java
273. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 3945:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. public synchronized void pause() //poller.java
```

```
...
281. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3946:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
272. public synchronized void pause() //poller.java
...
281. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3947:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. public synchronized void resume() //poller.java
286. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 3948:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. public synchronized void resume() //poller.java
...
294. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3949:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. public synchronized void resume() //poller.java
...
294. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3950:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
329. public void visitElement(PollableElement element) //poller.java
```

```
330. if (!element.hasOpenOutage())
```

[Table of Contents] [Description]

Path 3951:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
329. public void visitElement(PollableElement element) //poller.java
330. if (!element.hasOpenOutage())
```

[Table of Contents] [Description]

Path 3952:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
416. private void scheduleService(int nodeId, String ipAddr, String serviceName,
Number svcLostEventId, Date date, String svcLostUei) //poller.java
...
450. if (svcLostEventId == null)
```

[Table of Contents] [Description]

Path 3953:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
416. private void scheduleService(int nodeId, String ipAddr, String serviceName,
Number svcLostEventId, Date date, String svcLostUei) //poller.java
...
450. if (svcLostEventId == null)
```

[Table of Contents] [Description]

Path 3954:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. private Package findPackageForService(String ipAddr, String
serviceName) //poller.java
...
477. if (pollableServiceInPackage(ipAddr, serviceName, pkg))
```

[Table of Contents] [Description]

Path 3955:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
483. private boolean pollableServiceInPackage(String ipAddr, String serviceName,
Package pkg) //poller.java
      484. if (!m_pollerConfig.serviceInPackageAndEnabled(serviceName, pkg)) return
      false;
```

[Table of Contents] [Description]

Path 3956:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
483. private boolean pollableServiceInPackage(String ipAddr, String serviceName,
Package pkg) //poller.java
      ...
      487. if (inPkg) return true;
```

[Table of Contents] [Description]

Path 3957:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      ...
      501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3958:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      ...
      501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3959:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      ...
      513. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3960:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      |...
      |513. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3961:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      |...
      |518. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3962:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
497. public boolean packageIncludesIfAndSvc(Package pkg, String ipAddr, String
svcName) //poller.java
      |...
      |518. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3963:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
      |...
      |597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3964:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
      |...
```

```
597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3965:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
...
605. if (address != null)
```

[Table of Contents] [Description]

Path 3966:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
...
605. if (address != null)
```

[Table of Contents] [Description]

Path 3967:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
...
608. if (svcName != null)
```

[Table of Contents] [Description]

Path 3968:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
594. public Event createEvent(String uei, int nodeId, InetAddress address, String
svcName, java.util.Date date) //poller.java
...
608. if (svcName != null)
```

[Table of Contents] [Description]

Path 3969:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private void interfaceReparentedHandler(Event
event) //pollereventprocessor.java
      |
      ...
      215. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3970:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private void interfaceReparentedHandler(Event
event) //pollereventprocessor.java
      |
      ...
      215. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3971:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private void interfaceReparentedHandler(Event
event) //pollereventprocessor.java
      |
      ...
      219. if (event.getInterface() == null)
```

[Table of Contents] [Description]

Path 3972:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private void interfaceReparentedHandler(Event
event) //pollereventprocessor.java
      |
      ...
      238. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3973:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
330. private void nodeDeletedHandler(Event event) //pollereventprocessor.java
      |
      ...
      349. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3974:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
330. private void nodeDeletedHandler(Event event) //pollereventprocessor.java
      ...
357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3975:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
330. private void nodeDeletedHandler(Event event) //pollereventprocessor.java
      ...
357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3976:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      ...
414. if (parmValue == null)
```

[Table of Contents] [Description]

Path 3977:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      ...
422. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3978:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. private void interfaceDeletedHandler(Event event) //pollereventprocessor.java
      ...
422. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3979:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
512. PollerEventProcessor(Poller poller) //pollereventprocessor.java
      ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3980:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
512. PollerEventProcessor(Poller poller) //pollereventprocessor.java
      ...
519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 3981:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
546. public void onEvent(Event event) //pollereventprocessor.java
      547. if (event == null)
```

[Table of Contents] [Description]

Path 3982:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. OutageChecker(MockService svc, Event lostSvcEvent, Event
regainedSvcEvent) //pollertest.java
      ...
854. if (m_regainedSvcEvent != null)
```

[Table of Contents] [Description]

Path 3983:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
866. private Integer getRegainedEventId() //pollertest.java
      867. if (m_regainedSvcEvent == null)
```

[Table of Contents] [Description]

Path 3984:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. public boolean scheduledSuspension() //mockinterval.java
      ...
85. if (suspension.contains(m_timer.getCurrentTime()))
```

[Table of Contents] [Description]

Path 3985:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void setEventMgr(EventIpcManager eventMgr) //mockpollcontext.java
88. if (m_eventMgr != null)
```

[Table of Contents] [Description]

Path 3986:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void setEventMgr(EventIpcManager eventMgr) //mockpollcontext.java
88. if (m_eventMgr != null)
```

[Table of Contents] [Description]

Path 3987:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void setEventMgr(EventIpcManager eventMgr) //mockpollcontext.java
      ...
91. if (m_eventMgr != null)
```

[Table of Contents] [Description]

Path 3988:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void setEventMgr(EventIpcManager eventMgr) //mockpollcontext.java
      ...
91. if (m_eventMgr != null)
```

[Table of Contents] [Description]

Path 3989:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutage(final PollableService pSvc, final PollEvent
svcLostEvent) //mockpollcontext.java
     |
     ...
126. if (svcLostEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3990:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutage(final PollableService pSvc, final PollEvent
svcLostEvent) //mockpollcontext.java
     |
     ...
126. if (svcLostEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3991:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public void openOutage(final PollableService pSvc, final PollEvent
svcLostEvent) //mockpollcontext.java
     |
     ...
126. if (svcLostEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3992:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. public void resolveOutage(final PollableService pSvc, final PollEvent
svcRegainEvent) //mockpollcontext.java
     |
     ...
145. if (svcRegainEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3993:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. public void resolveOutage(final PollableService pSvc, final PollEvent
svcRegainEvent) //mockpollcontext.java
     |
     ...
145. if (svcRegainEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3994:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
139. public void resolveOutage(final PollableService pSvc, final PollEvent
svcRegainEvent) //mockpollcontext.java
      |...
      |145. if (svcRegainEvent instanceof PendingPollEvent)
```

[Table of Contents] [Description]

Path 3995:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
178. public synchronized void onEvent(Event e) //mockpollcontext.java
      |...
      |189. if (pollEvent.isPending()) break;
```

[Table of Contents] [Description]

Path 3996:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. public long getNextTime() //mockscheduler.java
      |80. if (m_scheduleEntries.isEmpty())
```

[Table of Contents] [Description]

Path 3997:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public long next() //mockscheduler.java
      |88. if (m_scheduleEntries.isEmpty())
```

[Table of Contents] [Description]

Path 3998:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public long next() //mockscheduler.java
      |...
      |96. if (entries.isEmpty())
```

[Table of Contents] [Description]

Path 3999:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public long next() //mockscheduler.java
      |...
      |96. if (entries.isEmpty())
```

[Table of Contents] [Description]

Path 4000:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
102. public long tick(int step) //mockscheduler.java
      |103. if (m_scheduleEntries.isEmpty())
```

[Table of Contents] [Description]

Path 4001:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
      |...
      |116. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4002:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
      |...
      |141. if (timeout == 0)
```

[Table of Contents] [Description]

Path 4003:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
      |...
      |149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4004:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
     |...
     |149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4005:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dhcpmonitor.java
     |...
     |96. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4006:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dhcpmonitor.java
     |...
     |122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4007:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dhcpmonitor.java
     |...
     |122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4008:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     |...
```

```
117. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4009:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
188. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4010:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
188. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4011:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
207. if (socket != null)
```

[Table of Contents] [Description]

Path 4012:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
     ...
207. if (socket != null)
```

[Table of Contents] [Description]

Path 4013:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
130. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4014:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
145. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4015:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
145. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4016:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4017:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4018:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      |...
      |173. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4019:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      |...
      |173. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4020:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      |...
      |209. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4021:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      |...
      |209. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4022:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      |...
```

```
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4023:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4024:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
...
223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4025:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
...
223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4026:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
...
228. if (socket != null)
```

[Table of Contents] [Description]

Path 4027:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
      228. if (socket != null)
```

[Table of Contents] [Description]

Path 4028:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
      232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4029:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dominoiiopmonitor.java
      ...
      232. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4030:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private String retrieveIORText(String host, int port) throws
IOException //dominoiiopmonitor.java
      ...
      273. if (!IOR.startsWith("IOR:"))
```

[Table of Contents] [Description]

Path 4031:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
      ...
      152. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4032:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     |...
     |180. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4033:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     |...
     |180. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4034:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     |...
     |209. if (banner == null)
```

[Table of Contents] [Description]

Path 4035:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     |...
     |231. if (banner == null)
```

[Table of Contents] [Description]

Path 4036:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     |...
```

```
266. if (response == null)
```

[Table of Contents] [Description]

Path 4037:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
283. if (response == null)
```

[Table of Contents] [Description]

Path 4038:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
309. if (response == null)
```

[Table of Contents] [Description]

Path 4039:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
315. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4040:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
...
315. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4041:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
320. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4042:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
320. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4043:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4044:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4045:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
341. if (response == null)
```

[Table of Contents] [Description]

Path 4046:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
364. if (response == null)
```

[Table of Contents] [Description]

Path 4047:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
456. if (socket != null)
```

[Table of Contents] [Description]

Path 4048:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
     ...
456. if (socket != null)
```

[Table of Contents] [Description]

Path 4049:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
112. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4050:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
```

```
122. if (0 < timeout && timeout < 1000)
```

[Table of Contents] [Description]

Path 4051:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4052:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4053:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
165. if (args == null)
```

[Table of Contents] [Description]

Path 4054:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
     ...
182. if (!scriptoutput.equals(""))
```

[Table of Contents] [Description]

Path 4055:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
      ...
182. if (!scriptoutput.equals(""))
```

[Table of Contents] [Description]

Path 4056:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
      ...
182. if (!scriptoutput.equals(""))
```

[Table of Contents] [Description]

Path 4057:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
      ...
186. if (!scripterror.equals(""))
```

[Table of Contents] [Description]

Path 4058:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
      ...
186. if (!scripterror.equals(""))
```

[Table of Contents] [Description]

Path 4059:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
      ...
188. if (strBannerMatch == null || strBannerMatch.equals("*"))
```

[Table of Contents] [Description]

Path 4060:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     |...
     |124. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4061:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     |...
     |207. if (line == null)
```

[Table of Contents] [Description]

Path 4062:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     |...
     |245. if (line == null)
```

[Table of Contents] [Description]

Path 4063:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     |...
     |258. if (responseIndex != -1)
```

[Table of Contents] [Description]

Path 4064:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
     |...
```

```
266. if (!bResponseTextFound)
```

[Table of Contents] [Description]

Path 4065:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
...
292. if (socket != null)
```

[Table of Contents] [Description]

Path 4066:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
...
292. if (socket != null)
```

[Table of Contents] [Description]

Path 4067:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
...
314. if (i == 0)
```

[Table of Contents] [Description]

Path 4068:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
...
314. if (i == 0)
```

[Table of Contents] [Description]

Path 4069:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
      ...
      314. if (i == 0)
```

[Table of Contents] [Description]

Path 4070:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
      ...
      129. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4071:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
      ...
      220. if (line == null)
```

[Table of Contents] [Description]

Path 4072:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
      ...
      223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4073:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
      ...
      223. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4074:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     |...
     |280. if (line == null)
```

[Table of Contents] [Description]

Path 4075:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     |...
     |293. if (responseIndex != -1)
```

[Table of Contents] [Description]

Path 4076:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     |...
     |301. if (!bResponseTextFound)
```

[Table of Contents] [Description]

Path 4077:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     |...
     |329. if (socket != null)
```

[Table of Contents] [Description]

Path 4078:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
     |...
```

```
329. if (socket != null)
```

[Table of Contents] [Description]

Path 4079:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
352. if (i == 0)
```

[Table of Contents] [Description]

Path 4080:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
352. if (i == 0)
```

[Table of Contents] [Description]

Path 4081:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
...
352. if (i == 0)
```

[Table of Contents] [Description]

Path 4082:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
...
270. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4083:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
      ...
      357. if (invertResult.equals("true") && serviceStatus ==
      ServiceMonitor.SERVICE_AVAILABLE)
```

[Table of Contents] [Description]

Path 4084:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
267. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //icmpmonitor.java
      ...
      359. else if (invertResult.equals("true"))
```

[Table of Contents] [Description]

Path 4085:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
      ...
      146. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4086:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
      ...
      172. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4087:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
      ...
      172. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4088:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     |...
     |202. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4089:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     |...
     |202. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4090:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     |...
     |257. if (socket != null)
```

[Table of Contents] [Description]

Path 4091:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
     |...
     |257. if (socket != null)
```

[Table of Contents] [Description]

Path 4092:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void initialize(PollerConfig pollerConfig, Map
parameters) //ipv4latencymonitor.java
     |...
```

```
108. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4093:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void initialize(PollerConfig pollerConfig, Map
parameters) //ipv4latencymonitor.java
    ...
108. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4094:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void initialize(PollerConfig pollerConfig, Map
parameters) //ipv4latencymonitor.java
    ...
113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4095:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. public void initialize(PollerConfig pollerConfig, Map
parameters) //ipv4latencymonitor.java
    ...
113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4096:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public void initialize(NetworkInterface iface) //ipv4latencymonitor.java
    169. if (!(iface.getAddress() instanceof InetAddress))
```

[Table of Contents] [Description]

Path 4097:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
136. public void initialize(NetworkInterface iface) //ipv4monitor.java
     |137. if (!(iface.getAddress() instanceof InetAddress))
```

[Table of Contents] [Description]

Path 4098:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. public synchronized Object getAttribute(String
property) //ipv4networkinterface.java
     |...
     |125. if (m_properties != null)
```

[Table of Contents] [Description]

Path 4099:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
154. public synchronized Object setAttribute(String property, Object
value) //ipv4networkinterface.java
     |155. if (m_properties == null)
```

[Table of Contents] [Description]

Path 4100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
     |...
     |76. if ( storedProcedure == null )
```

[Table of Contents] [Description]

Path 4101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
     |...
     |81. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
    ...
81. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
    ...
86. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
    ...
86. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
    ...
155. if (addressObject instanceof InetAddress)
```

[Table of Contents] [Description]

Path 4106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
    ...
157. else if (addressObject instanceof String)
```

[Table of Contents] [Description]

Path 4107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
178. if (socket != null)
```

[Table of Contents] [Description]

Path 4108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
137. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ldapmonitor.java
     ...
178. if (socket != null)
```

[Table of Contents] [Description]

Path 4109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //nrpemonitor.java
     ...
120. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
     ...
108. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
```

```
...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
...
184. if (socket != null)
```

[Table of Contents] [Description]

Path 4114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
...
184. if (socket != null)
```

[Table of Contents] [Description]

Path 4115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
...
124. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
178. if (banner == null)
```

[Table of Contents] [Description]

Path 4119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
215. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
215. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     |...
     224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     |...
     224. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     |...
     228. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     |...
     228. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
     |
```

```
...
233. if (socket != null)
```

[Table of Contents] [Description]

Path 4126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
...
233. if (socket != null)
```

[Table of Contents] [Description]

Path 4127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
...
237. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
...
98. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      ...
      161. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      ...
      161. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      ...
      164. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      ...
      164. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      ...
      167. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smbmonitor.java
      |
      ...
      167. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
      |
      ...
      162. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
      |
      ...
      186. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
      |
      ...
      186. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
      |
```

```
...
216. if (banner == null)
```

[Table of Contents] [Description]

Path 4140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
238. if (banner == null)
```

[Table of Contents] [Description]

Path 4141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
242. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
242. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
261. if (response == null)
```

[Table of Contents] [Description]

Path 4144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     |...
     |283. if (response == null)
```

[Table of Contents] [Description]

Path 4145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     |...
     |298. if (response == null)
```

[Table of Contents] [Description]

Path 4146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     |...
     |320. if (response == null)
```

[Table of Contents] [Description]

Path 4147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     |...
     |350. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     |...
     |350. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
354. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
354. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
363. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
363. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
```

```
...
368. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
368. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
373. if (socket != null)
```

[Table of Contents] [Description]

Path 4156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
373. if (socket != null)
```

[Table of Contents] [Description]

Path 4157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
...
377. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
377. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
     ...
163. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
     ...
163. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
     ...
167. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
     ...
167. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
      ...
171. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. public void initialize(PollerConfig pollerConfig, Map
parameters) //snmpmonitor.java
      ...
171. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. public void initialize(NetworkInterface iface) //snmpmonitor.java
      ...
200. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. public void initialize(NetworkInterface iface) //snmpmonitor.java
      ...
217. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
193. public void initialize(NetworkInterface iface) //snmpmonitor.java
      ...
217. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      ...
257. if (peer == null)
```

[Table of Contents] [Description]

Path 4169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      ...
274. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      ...
274. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      ...
287. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
```

```
...
287. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
...
293. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
...
293. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
...
347. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
...
347. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public boolean meetsCriteria(SnmpSyntax result, String operator, String
operand) //snmpmonitor.java
     365. if (result == null)
```

[Table of Contents] [Description]

Path 4178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public boolean meetsCriteria(SnmpSyntax result, String operator, String
operand) //snmpmonitor.java
     ...
     368. if (operator == null || operand == null)
```

[Table of Contents] [Description]

Path 4179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public boolean meetsCriteria(SnmpSyntax result, String operator, String
operand) //snmpmonitor.java
     ...
     372. if (EQUALS.equals(operator))
```

[Table of Contents] [Description]

Path 4180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public boolean meetsCriteria(SnmpSyntax result, String operator, String
operand) //snmpmonitor.java
     ...
     374. else if (NOT_EQUAL.equals(operator))
```

[Table of Contents] [Description]

Path 4181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
364. public boolean meetsCriteria(SnmpSyntax result, String operator, String
operand) //snmpmonitor.java
     ...
     376. else if (MATCHES.equals(operator))
```

[Table of Contents] [Description]

Path 4182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |...
     120. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |...
     150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |...
     150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |...
     188. if (response == null)
```

[Table of Contents] [Description]

Path 4186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |...
```

```
221. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
221. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
230. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
230. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
235. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |
     ...
235. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |
     ...
240. if (socket != null)
```

[Table of Contents] [Description]

Path 4193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |
     ...
240. if (socket != null)
```

[Table of Contents] [Description]

Path 4194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |
     ...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     |
     ...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     |...
     |126. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     |...
     |156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     |...
     |156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     |...
     |203. if (response == null)
```

[Table of Contents] [Description]

Path 4200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     |...
```

```
224. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     ...
224. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     ...
233. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     ...
233. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
     ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
243. if (socket != null)
```

[Table of Contents] [Description]

Path 4207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
243. if (socket != null)
```

[Table of Contents] [Description]

Path 4208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
247. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. public static String convertTypeToString(short type) //nsclientmanager.java
     ...
209. if (val == type)
```

[Table of Contents] [Description]

Path 4211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. private NsclientPacket sendCheckRequest(String request) //nsclientmanager.java
     ...
444. if (read > 0)
```

[Table of Contents] [Description]

Path 4212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
434. private NsclientPacket sendCheckRequest(String request) //nsclientmanager.java
     ...
444. if (read > 0)
```

[Table of Contents] [Description]

Path 4213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
720. private NsclientPacket checkProcState(NsclientCheckParams
param) //nsclientmanager.java
     ...
739. if (services[i].split(":\\s+")[1].matches("not running\\s*"))
```

[Table of Contents] [Description]

Path 4214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
720. private NsclientPacket checkProcState(NsclientCheckParams
param) //nsclientmanager.java
     ...
739. if (services[i].split(":\\s+")[1].matches("not running\\s*"))
```

[Table of Contents] [Description]

Path 4215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
929. private NsclientPacket checkFileAge(NsclientCheckParams
param) //nsclientmanager.java
      ...
953. if (minutes < param.getWarningPercent())
```

[Table of Contents] [Description]

Path 4216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
929. private NsclientPacket checkFileAge(NsclientCheckParams
param) //nsclientmanager.java
      ...
953. if (minutes < param.getWarningPercent())
```

[Table of Contents] [Description]

Path 4217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
929. private NsclientPacket checkFileAge(NsclientCheckParams
param) //nsclientmanager.java
      ...
957. if (minutes < param.getCriticalPercent())
```

[Table of Contents] [Description]

Path 4218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
929. private NsclientPacket checkFileAge(NsclientCheckParams
param) //nsclientmanager.java
      ...
957. if (minutes < param.getCriticalPercent())
```

[Table of Contents] [Description]

Path 4219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
71. public static String convertStateToString(short type) //nsclientpacket.java
      |...
      |76. if (val == type)
```

[Table of Contents] [Description]

Path 4220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public boolean equals(PollEvent e) //dbpollevent.java
      |63. if (e == null) return false;
```

[Table of Contents] [Description]

Path 4221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
67. public boolean equals(Object o) //dbpollevent.java
      |68. if (o instanceof PollEvent)
```

[Table of Contents] [Description]

Path 4222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public void addPending(Runnable r) //pendingpollevent.java
      |73. if (m_pending)
```

[Table of Contents] [Description]

Path 4223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public void addPending(Runnable r) //pendingpollevent.java
      |73. if (m_pending)
```

[Table of Contents] [Description]

Path 4224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public void addPending(Runnable r) //pendingpollevent.java
      |73. if (m_pending)
```

[Table of Contents] [Description]

Path 4225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
92. public void deleteMember(PollableElement member) //pollablecontainer.java
      ...
94. if (m_members.size() == 0)
```

[Table of Contents] [Description]

Path 4226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
92. public void deleteMember(PollableElement member) //pollablecontainer.java
      ...
94. if (m_members.size() == 0)
```

[Table of Contents] [Description]

Path 4227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public void forEachElement(PollableElement elem) //pollablecontainer.java
      ...
159. if (elem.getStatus().isUp())
```

[Table of Contents] [Description]

Path 4228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public void forEachElement(PollableElement elem) //pollablecontainer.java
      ...
159. if (elem.getStatus().isUp())
```

[Table of Contents] [Description]

Path 4229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
216. public void forEachElement(PollableElement elem) //pollablecontainer.java
      ...
218. if (elem.poll().isUp())
```

[Table of Contents] [Description]

Path 4230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
216. public void forEachElement(PollableElement elem) //pollablecontainer.java
     ...
218. if (elem.poll().isUp())
```

[Table of Contents] [Description]

Path 4231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public void forEachElement(PollableElement elem) //pollablecontainer.java
     230. if (elem.getStatus().isUp())
```

[Table of Contents] [Description]

Path 4232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
229. public void forEachElement(PollableElement elem) //pollablecontainer.java
     230. if (elem.getStatus().isUp())
```

[Table of Contents] [Description]

Path 4233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
239. public PollStatus poll() //pollablecontainer.java
     ...
241. if (leaf == null) return PollStatus.STATUS_UP;
```

[Table of Contents] [Description]

Path 4234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
248. public PollableElement selectPollElement() //pollablecontainer.java
     249. if (getMemberCount() == 0)
```

[Table of Contents] [Description]

Path 4235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
314. public void forEachElement(PollableElement elem) //pollablecontainer.java
     |315. if (!hasOpenOutage())
```

[Table of Contents] [Description]

Path 4236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
314. public void forEachElement(PollableElement elem) //pollablecontainer.java
     |315. if (!hasOpenOutage())
```

[Table of Contents] [Description]

Path 4237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
153. protected PollStatus poll(PollableElement elem) //pollableelement.java
     |154. if (elem != this)
```

[Table of Contents] [Description]

Path 4238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
232. protected void processCause(PollEvent cause) //pollableelement.java
     |233. if (!hasOpenOutage())
```

[Table of Contents] [Description]

Path 4239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
232. protected void processCause(PollEvent cause) //pollableelement.java
     |233. if (!hasOpenOutage())
```

[Table of Contents] [Description]

Path 4240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
237. protected void resolveAllOutages(PollEvent resolvedCause, PollEvent
resolution) //pollableelement.java
     '
```

```
238. if (resolvedCause.equals(getCause()))
```

[Table of Contents] [Description]

Path 4241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
237. protected void resolveAllOutages(PollEvent resolvedCause, PollEvent
resolution) //pollableelement.java
238. if (resolvedCause.equals(getCause()))
```

[Table of Contents] [Description]

Path 4242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
134. protected PollStatus poll(PollableElement elem) //pollableinterface.java
...
136. if (getStatus().isUp() || critSvc == null || elem == critSvc)
```

[Table of Contents] [Description]

Path 4243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public PollStatus pollRemainingMembers(PollableElement
member) //pollableinterface.java
...
147. if (member != critSvc)
```

[Table of Contents] [Description]

Path 4244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public PollStatus pollRemainingMembers(PollableElement
member) //pollableinterface.java
...
147. if (member != critSvc)
```

[Table of Contents] [Description]

Path 4245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
142. public PollStatus pollRemainingMembers(PollableElement
member) //pollableinterface.java
      ...
153. if (getContext().isPollingAllIfCritServiceUndefined())
```

[Table of Contents] [Description]

Path 4246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
171. public void reparentTo(final PollableNode newNode) //pollableinterface.java
      ...
174. if (oldNode.equals(newNode)) return;
```

[Table of Contents] [Description]

Path 4247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public PollableInterface createInterface(int nodeId, InetAddress
addr) //pollablenetwork.java
      ...
78. if (node == null)
```

[Table of Contents] [Description]

Path 4248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
88. public PollableService createService(int nodeId, InetAddress addr, String
svcName) //pollablenetwork.java
      ...
92. if (node == null)
```

[Table of Contents] [Description]

Path 4249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public synchronized void obtain(long timeout) //pollablenode.java
      ...
82. if (now >= endTime)
```

[Table of Contents] [Description]

Path 4250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
161. public void run() //pollablenode.java
      |...
      |163. if (iface == null)
```

[Table of Contents] [Description]

Path 4251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void obtainTreeLock(long timeout) //pollablenode.java
      |207. if (timeout == 0)
```

[Table of Contents] [Description]

Path 4252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void obtainTreeLock(long timeout) //pollablenode.java
      |207. if (timeout == 0)
```

[Table of Contents] [Description]

Path 4253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void obtainTreeLock(long timeout) //pollablenode.java
      |207. if (timeout == 0)
```

[Table of Contents] [Description]

Path 4254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void processStatusChange(Date date) //pollableservice.java
      |...
      |207. if (m_oldStatus == PollStatus.STATUS_UP)
```

[Table of Contents] [Description]

Path 4255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void processStatusChange(Date date) //pollableservice.java
```

```
...
207. if (m_oldStatus == PollStatus.STATUS_UP)
```

[Table of Contents] [Description]

Path 4256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void processStatusChange(Date date) //pollableservice.java
...
212. if (getStatus() == PollStatus.STATUS_UP)
```

[Table of Contents] [Description]

Path 4257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void processStatusChange(Date date) //pollableservice.java
...
212. if (getStatus() == PollStatus.STATUS_UP)
```

[Table of Contents] [Description]

Path 4258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
219. public void updateStatus(PollStatus newStatus) //pollableservice.java
...
222. if (newStatus == PollStatus.STATUS_UNRESPONSIVE)
```

[Table of Contents] [Description]

Path 4259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public void run() //pollableservice.java
...
295. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public void run() //pollableservice.java
```

```
      ...
295. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
313. public void schedule() //pollableservice.java
      314. if (m_schedule == null)
```

[Table of Contents] [Description]

Path 4262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
146. public long getInterval() //pollableserviceconfig.java
      ...
      148. if (m_service.isDeleted())
```

[Table of Contents] [Description]

Path 4263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public boolean scheduledSuspension() //pollableserviceconfig.java
      ...
      201. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 4264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
191. public boolean scheduledSuspension() //pollableserviceconfig.java
      ...
      201. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 4265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public void visitElement(PollableElement element) //pollablestest.java
```

```
264. if (!element.hasOpenOutage())
```

[Table of Contents] [Description]

Path 4266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public void visitElement(PollableElement element) //pollablestest.java
     264. if (!element.hasOpenOutage())
```

[Table of Contents] [Description]

Path 4267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
     ...
     311. if (svcLostEventId == null)
```

[Table of Contents] [Description]

Path 4268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
287. public void processRow(ResultSet rs) throws SQLException //pollablestest.java
     ...
     311. if (svcLostEventId == null)
```

[Table of Contents] [Description]

Path 4269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
337. private Package findPackageForService(String ipAddr, String
serviceName) //pollablestest.java
     ...
     342. if (pollableServiceInPackage(ipAddr, serviceName, pkg))
```

[Table of Contents] [Description]

Path 4270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
726. public void visitNode(PollableNode node) //pollablestest.java
     727. if (node == pDot1Icmp.getNode())
```

[Table of Contents] [Description]

Path 4271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
726. public void visitNode(PollableNode node) //pollablestest.java
     |727. if (node == pDot1Icmp.getNode())
```

[Table of Contents] [Description]

Path 4272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
726. public void visitNode(PollableNode node) //pollablestest.java
     |727. if (node == pDot1Icmp.getNode())
```

[Table of Contents] [Description]

Path 4273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
732. public void visitInterface(PollableInterface iface) //pollablestest.java
     |733. if (iface == pDot1Icmp.getInterface())
```

[Table of Contents] [Description]

Path 4274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
732. public void visitInterface(PollableInterface iface) //pollablestest.java
     |733. if (iface == pDot1Icmp.getInterface())
```

[Table of Contents] [Description]

Path 4275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
732. public void visitInterface(PollableInterface iface) //pollablestest.java
     |733. if (iface == pDot1Icmp.getInterface())
```

[Table of Contents] [Description]

Path 4276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
738. public void visitService(PollableService s) //pollablestest.java
     |739. if (s == pDot1Icmp)
```

[Table of Contents] [Description]

Path 4277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
738. public void visitService(PollableService s) //pollablestest.java
     |739. if (s == pDot1Icmp)
```

[Table of Contents] [Description]

Path 4278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
738. public void visitService(PollableService s) //pollablestest.java
     |739. if (s == pDot1Icmp)
```

[Table of Contents] [Description]

Path 4279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. public static synchronized Interface getInstance() //interface.java
     |164. if (!m_loaded)
```

[Table of Contents] [Description]

Path 4280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
95. private void checkState(String methodName) //jnirrdstrategy.java
     |96. if (!initialized)
```

[Table of Contents] [Description]

Path 4281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
     |...
```

```
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
     ...
218. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
     ...
261. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
     ...
261. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
     ...
267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
188. public Double fetchLastValue(String rrdFile, int interval) throws
NumberFormatException, RrdException //jnirrdstrategy.java
      ...
      267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
      ...
      308. if (log.isEnabledFor(Priority.DEBUG))
```

[Table of Contents] [Description]

Path 4288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
      ...
      308. if (log.isEnabledFor(Priority.DEBUG))
```

[Table of Contents] [Description]

Path 4289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
      ...
      341. if (log.isEnabledFor(Priority.DEBUG))
```

[Table of Contents] [Description]

Path 4290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
      ...
      341. if (log.isEnabledFor(Priority.DEBUG))
```

[Table of Contents] [Description]

Path 4291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
    ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
    ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
    ...
364. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public Double fetchLastValueInRange(String rrdFile, int interval, int range)
throws NumberFormatException, RrdException //jnirrdstrategy.java
    ...
364. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
379. public InputStream createGraph(String command, File workDir) throws
IOException, RrdException //jnirrdstrategy.java
    ...
```

```
402. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
379. public InputStream createGraph(String command, File workDir) throws
IOException, RrdException //jnirrdstrategy.java
...
402. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
247. if (debugTokens)
```

[Table of Contents] [Description]

Path 4298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
247. if (debugTokens)
```

[Table of Contents] [Description]

Path 4299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
251. if (debugTokens)
```

[Table of Contents] [Description]

Path 4300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      251. if (debugTokens)
```

[Table of Contents] [Description]

Path 4301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      264. if (debugTokens)
```

[Table of Contents] [Description]

Path 4302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      264. if (debugTokens)
```

[Table of Contents] [Description]

Path 4303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      267. if (debugTokens)
```

[Table of Contents] [Description]

Path 4304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      267. if (debugTokens)
```

[Table of Contents] [Description]

Path 4305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
269. if (quoting && !processQuoted)
```

[Table of Contents] [Description]

Path 4306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
269. if (quoting && !processQuoted)
```

[Table of Contents] [Description]

Path 4307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
274. if (!processQuoted)
```

[Table of Contents] [Description]

Path 4308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
274. if (!processQuoted)
```

[Table of Contents] [Description]

Path 4309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
```

```
277. if (debugTokens)
```

[Table of Contents] [Description]

Path 4310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
277. if (debugTokens)
```

[Table of Contents] [Description]

Path 4311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
281. if (debugTokens)
```

[Table of Contents] [Description]

Path 4312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
281. if (debugTokens)
```

[Table of Contents] [Description]

Path 4313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
...
286. if (debugTokens)
```

[Table of Contents] [Description]

Path 4314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      286. if (debugTokens)
```

[Table of Contents] [Description]

Path 4315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      291. if (debugTokens)
```

[Table of Contents] [Description]

Path 4316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      291. if (debugTokens)
```

[Table of Contents] [Description]

Path 4317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      299. if (debugTokens)
```

[Table of Contents] [Description]

Path 4318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |
      ...
      299. if (debugTokens)
```

[Table of Contents] [Description]

Path 4319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |...
      |304. if (debugTokens)
```

[Table of Contents] [Description]

Path 4320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      |...
      |304. if (debugTokens)
```

[Table of Contents] [Description]

Path 4321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
535. private void parseGraphColor(RrdGraphDef graphDef, String colorArg) throws
IllegalArgumentException //jrobinrrdstrategy.java
      |...
      |539. if (colorArgParts.length != 2)
```

[Table of Contents] [Description]

Path 4322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
535. private void parseGraphColor(RrdGraphDef graphDef, String colorArg) throws
IllegalArgumentException //jrobinrrdstrategy.java
      |...
      |548. if (colorHex.length() != 6)
```

[Table of Contents] [Description]

Path 4323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
289. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
      |...
```

```
291. if (rrd == null)
```

[Table of Contents] [Description]

Path 4324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
     ...
336. if (rrd == null)
```

[Table of Contents] [Description]

Path 4325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public Operation makeUpdateOperation(String fileName, String
update) //queuingrrdstrategy.java
     ...
427. if (initialTimeStamp == 0)
```

[Table of Contents] [Description]

Path 4326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
422. public Operation makeUpdateOperation(String fileName, String
update) //queuingrrdstrategy.java
     ...
427. if (initialTimeStamp == 0)
```

[Table of Contents] [Description]

Path 4327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
447. public void addOperation(Operation op) //queuingrrdstrategy.java
     ...
468. if (op.isSignificant())
```

[Table of Contents] [Description]

Path 4328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
480. private boolean queueIsFull() //queuingrrdstrategy.java
     |481. if (QUEUE_HIGH_WATER_MARK <= 0)
```

[Table of Contents] [Description]

Path 4329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
487. private boolean sigQueueIsFull() //queuingrrdstrategy.java
     |488. if (SIG_HIGH_WATER_MARK <= 0)
```

[Table of Contents] [Description]

Path 4330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
494. private boolean inSigQueueIsFull() //queuingrrdstrategy.java
     |495. if (INSIG_HIGH_WATER_MARK <= 0)
```

[Table of Contents] [Description]

Path 4331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
518. public LinkedList getNext() //queuingrrdstrategy.java
     |...
     |535. if (updateStart == 0)
```

[Table of Contents] [Description]

Path 4332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. private synchronized void storeAssignment(Operation
op) //queuingrrdstrategy.java
     |...
     |577. if (!PRIORITIZE_SIGS || op.isSignificant())
```

[Table of Contents] [Description]

Path 4333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. private synchronized void storeAssignment(Operation
op) //queuingrrdstrategy.java
```

```
    ...
577. if (!PRIORITIZE_SIGS || op.isSignificant())
```

[Table of Contents] [Description]

Path 4334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
564. private synchronized void storeAssignment(Operation
op) //queuingrrdstrategy.java
    ...
577. if (!PRIORITIZE_SIGS || op.isSignificant())
```

[Table of Contents] [Description]

Path 4335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
599. private synchronized void promoteAgedFiles() //queuingrrdstrategy.java
    ...
602. if (!PRIORITIZE_SIGS) return;
```

[Table of Contents] [Description]

Path 4336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
599. private synchronized void promoteAgedFiles() //queuingrrdstrategy.java
    ...
605. if (MAX_INSIG_UPDATE_SECONDS == 0 || filesWithInsignificantWork.isEmpty
())
```

[Table of Contents] [Description]

Path 4337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
687. private synchronized void completeAssignment() //queuingrrdstrategy.java
    ...
690. if (previousAssignment != null)
```

[Table of Contents] [Description]

Path 4338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
687. private synchronized void completeAssignment() //queuingrrdstrategy.java
       ...
690. if (previousAssignment != null)
```

[Table of Contents] [Description]

Path 4339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
728. public void createFile(Object op) throws Exception //queuingrrdstrategy.java
       729. if (QUEUE_CREATES)
```

[Table of Contents] [Description]

Path 4340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
728. public void createFile(Object op) throws Exception //queuingrrdstrategy.java
       729. if (QUEUE_CREATES)
```

[Table of Contents] [Description]

Path 4341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
829. private void processPendingOperations() //queuingrrdstrategy.java
       ...
835. if (ops == null)
```

[Table of Contents] [Description]

Path 4342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public static String mapType(String objectType) //rrddatasource.java
       ...
158. if (objectType.startsWith(supportedObjectTypes[index]))
```

[Table of Contents] [Description]

Path 4343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. public RRDDataSource(MibObject obj, String collectionName) //rrddatasource.java
```

```
      ...
208. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. public RRDDataSource(MibObject obj, String collectionName) //rrddatasource.java
      ...
208. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. private static String makeZeroUpdate(int fileNum, String
update) //rrdstresser.java
      ...
162. if (initialTimeStamp == 0)
```

[Table of Contents] [Description]

Path 4346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. private static String makeZeroUpdate(int fileNum, String
update) //rrdstresser.java
      ...
162. if (initialTimeStamp == 0)
```

[Table of Contents] [Description]

Path 4347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
79. private static RrdStrategy getStrategy() throws RrdException //rrdutils.java
      80. if (m_rrdStrategy == null)
```

[Table of Contents] [Description]

Path 4348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public static boolean createRRD(String creator, String directory, String
dsName, int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List
rraList) throws RrdException //rrdutils.java
      |...
      157. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public static boolean createRRD(String creator, String directory, String
dsName, int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List
rraList) throws RrdException //rrdutils.java
      |...
      157. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
      |...
      211. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
      |...
      211. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
      |...
      223. if (rrd != null)
```

[Table of Contents] [Description]

Path 4353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
     ...
223. if (rrd != null)
```

[Table of Contents] [Description]

Path 4354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
     ...
230. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
     ...
230. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
     142. if (event == null)
```

[Table of Contents] [Description]

Path 4357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
     ...
147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      |147. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      |155. if (uei == null)
```

[Table of Contents] [Description]

Path 4360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      |160. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
141. public void onEvent(Event event) //broadcasteventprocessor.java
      |...
      |160. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
      |...
      |160. if (statusRS != null)
```

[Table of Contents] [Description]

Path 4363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
      ...
160. if (statusRS != null)
```

[Table of Contents] [Description]

Path 4364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
      ...
164. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
      ...
169. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
```

```
      ...
169. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
212. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
212. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
254. if (regainedTimeTS != null)
```

[Table of Contents] [Description]

Path 4371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
      266. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
      270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
      270. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
      ...
      276. if (outRS != null)
```

[Table of Contents] [Description]

Path 4376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
    ...
276. if (outRS != null)
```

[Table of Contents] [Description]

Path 4377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
    ...
279. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
    ...
279. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
    ...
303. if (!rtcS.contains(rtcN))
```

[Table of Contents] [Description]

Path 4380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
    ...
```

```
303. if (!rtcS.contains(rtcN))
```

[Table of Contents] [Description]

Path 4381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
...
308. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
...
308. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. private void addIpToCategory(List ipRTCs, RTCCategory cat) //datamanager.java
371. if (ipRTCs == null)
```

[Table of Contents] [Description]

Path 4384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. private void addIpToCategory(List ipRTCs, RTCCategory cat) //datamanager.java
...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. private void addIpToCategory(List ipRTCs, RTCCategory cat) //datamanager.java
     ...
383. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. private void addIpToCategory(List ipRTCs, RTCCategory cat) //datamanager.java
     ...
395. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
370. private void addIpToCategory(List ipRTCs, RTCCategory cat) //datamanager.java
     ...
395. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
489. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
506. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
506. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |
      ...
      526. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |
      ...
      544. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |
      ...
      544. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |
      ...
      551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      |
```

```
...
551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
561. if (svcRS != null)
```

[Table of Contents] [Description]

Path 4401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
561. if (svcRS != null)
```

[Table of Contents] [Description]

Path 4402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
564. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
564. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
571. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
571. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
577. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
577. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
583. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
583. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
590. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
590. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
593. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |
```

```
...
593. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
603. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 4415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
603. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 4416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
606. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 4417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
...
606. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 4418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
609. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 4419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
609. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 4420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
612. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
612. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
      ...
664. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
664. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
690. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
632. public DataManager() throws SAXException, IOException, SQLException,
FilterParseException, RTCException //datamanager.java
     |
     ...
690. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
710. public synchronized void nodeGainedService(long nodeid, String ip, String
svcName) //datamanager.java
     |
     ...
722. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
710. public synchronized void nodeGainedService(long nodeid, String ip, String
svcName) //datamanager.java
     |
```

```
...
722. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
710. public synchronized void nodeGainedService(long nodeid, String ip, String
svcName) //datamanager.java
     ...
725. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
710. public synchronized void nodeGainedService(long nodeid, String ip, String
svcName) //datamanager.java
     ...
725. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
1040. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
     ...
1040. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
1043. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
1043. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1106. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1106. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1141. if (catip.equals(ip))
```

[Table of Contents] [Description]

Path 4439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1145. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1145. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
```

```
...
1148. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1148. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1151. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1151. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1163. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1163. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1166. if (nodeid != catnodeid)
```

[Table of Contents] [Description]

Path 4448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1184. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1184. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1198. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1198. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
```

```
...
1203. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1216. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1216. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1222. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1222. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1228. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1228. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1235. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1235. if (ipRS != null)
```

[Table of Contents] [Description]

Path 4466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1238. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1248. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 4469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
```

```
...
1248. if (servicesGetStmt != null)
```

[Table of Contents] [Description]

Path 4470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1251. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 4471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1251. if (ipInfoGetStmt != null)
```

[Table of Contents] [Description]

Path 4472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1254. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 4473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
...
1254. if (outagesGetStmt != null)
```

[Table of Contents] [Description]

Path 4474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1257. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1468. public synchronized long getServiceID(String svcname) throws
SQLException //datamanager.java
      ...
1510. if (id != -1)
```

[Table of Contents] [Description]

Path 4477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1468. public synchronized long getServiceID(String svcname) throws
SQLException //datamanager.java
      ...
1510. if (id != -1)
```

[Table of Contents] [Description]

Path 4478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      ...
266. if (!urlList.contains(postInfo))
```

[Table of Contents] [Description]

Path 4479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      |
      ...
266. if (!urlList.contains(postInfo))
```

[Table of Contents] [Description]

Path 4480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      |
      ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      |
      ...
291. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      |
      ...
300. if (bytesRead > 0)
```

[Table of Contents] [Description]

Path 4483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
      |
```

```
...
300. if (bytesRead > 0)
```

[Table of Contents] [Description]

Path 4484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
...
311. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public synchronized void subscribe(String url, String catlabel, String user,
String passwd) //datasender.java
...
311. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized void unsubscribe(String urlStr) //datasender.java
...
352. if (value == null)
```

[Table of Contents] [Description]

Path 4487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized void unsubscribe(String urlStr) //datasender.java
...
365. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public synchronized void unsubscribe(String urlStr) //datasender.java
     ...
365. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
     ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
     ...
386. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
     ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
     ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
      |...
      399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
      |...
      399. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
      |...
      427. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
      |...
      427. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
      |...
      432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
       ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
       ...
439. if (bytesRead > 0)
```

[Table of Contents] [Description]

Path 4500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
373. public synchronized void sendData() //datasender.java
       ...
439. if (bytesRead > 0)
```

[Table of Contents] [Description]

Path 4501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
73. private void handleNodeGainedService(long nodeid, String ip, String
svcName) //dataupdater.java
       ...
83. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
73. private void handleNodeGainedService(long nodeid, String ip, String
svcName) //dataupdater.java
       ...
83. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. private void handleNodeLostService(long nodeid, String ip, String svcName, long
eventTime) //dataupdater.java
      ...
102. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. private void handleNodeLostService(long nodeid, String ip, String svcName, long
eventTime) //dataupdater.java
      ...
102. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. private void handleInterfaceDown(long nodeid, String ip, long
eventTime) //dataupdater.java
      ...
120. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. private void handleInterfaceDown(long nodeid, String ip, long
eventTime) //dataupdater.java
      ...
120. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private void handleNodeDown(long nodeid, long eventTime) //dataupdater.java
      ...
138. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private void handleNodeDown(long nodeid, long eventTime) //dataupdater.java
       ...
138. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
145. private void handleNodeUp(long nodeid, long eventTime) //dataupdater.java
       ...
156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
145. private void handleNodeUp(long nodeid, long eventTime) //dataupdater.java
       ...
156. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void handleInterfaceUp(long nodeid, String ip, long
eventTime) //dataupdater.java
       ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void handleInterfaceUp(long nodeid, String ip, long
eventTime) //dataupdater.java
       ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
181. private void handleNodeRegainedService(long nodeid, String ip, String svcName,
long eventTime) //dataupdater.java
      |
      |...
      |192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
181. private void handleNodeRegainedService(long nodeid, String ip, String svcName,
long eventTime) //dataupdater.java
      |
      |...
      |192. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void handleServiceDeleted(long nodeid, String ip, String
svcName) //dataupdater.java
      |
      |...
      |210. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
199. private void handleServiceDeleted(long nodeid, String ip, String
svcName) //dataupdater.java
      |
      |...
      |210. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
218. private void handleInterfaceReparented(String ip, Parms
eventParms) //dataupdater.java
      |
```

```
...
241. if (parmValue == null)
```

[Table of Contents] [Description]

Path 4518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
218. private void handleInterfaceReparented(String ip, Parms
eventParms) //dataupdater.java
...
275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
218. private void handleInterfaceReparented(String ip, Parms
eventParms) //dataupdater.java
...
275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. private void handleRtcSubscribe(Parms eventParms) //dataupdater.java
...
307. if (parmValue == null)
```

[Table of Contents] [Description]

Path 4521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. private void handleRtcSubscribe(Parms eventParms) //dataupdater.java
...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
285. private void handleRtcSubscribe(Parms eventParms) //dataupdater.java
        ...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
344. private void handleRtcUnsubscribe(Parms eventParms) //dataupdater.java
        ...
363. if (parmValue == null)
```

[Table of Contents] [Description]

Path 4524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
344. private void handleRtcUnsubscribe(Parms eventParms) //dataupdater.java
        ...
378. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
344. private void handleRtcUnsubscribe(Parms eventParms) //dataupdater.java
        ...
378. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
386. private void handleAssetInfoChangedEvent(long nodeid) throws SQLException,
FilterParseException, RTCException //dataupdater.java
        ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
386. private void handleAssetInfoChangedEvent(long nodeid) throws SQLException,
FilterParseException, RTCException //dataupdater.java
      ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
407. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
407. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
416. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
416. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
445. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
445. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
491. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. private void processEvent() throws SQLException, FilterParseException,
RTCException //dataupdater.java
      ...
491. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
      ...
255. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
255. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
261. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
261. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
267. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
283. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
288. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
288. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
294. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
294. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
310. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private synchronized void timerTaskComplete(RTCTimerTask tt) //rtcmanager.java
     ...
310. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
     ...
338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
     ...
338. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
343. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
343. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
359. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
359. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
370. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
370. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
385. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
385. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
393. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
398. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
       ...
403. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
      ...
403. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
      ...
413. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
      ...
413. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
      ...
420. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public synchronized void checkTimerTasksOnEventReceipt() //rtcmanager.java
      ...
420. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
432. public synchronized void resetUserTimer() //rtcmanager.java
      |
      ...
      |
434. if (m_userTask != null)
```

[Table of Contents] [Description]

Path 4573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
432. public synchronized void resetUserTimer() //rtcmanager.java
      |
      ...
      |
441. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
432. public synchronized void resetUserTimer() //rtcmanager.java
      |
      ...
      |
441. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. public void init() //rtcmanager.java
      |
      ...
      |
542. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. public void init() //rtcmanager.java
      |
      ...
      |
542. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. public void init() //rtcmanager.java
       ...
546. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
463. public void init() //rtcmanager.java
       ...
546. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
611. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
611. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
615. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
615. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
619. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
573. public void start() //rtcmanager.java
       ...
619. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
635. public void pause() //rtcmanager.java
     |636. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 4588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
635. public void pause() //rtcmanager.java
     |...
     |645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
635. public void pause() //rtcmanager.java
     |...
     |645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
652. public void resume() //rtcmanager.java
     |653. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 4591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
652. public void resume() //rtcmanager.java
     |...
     |662. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
652. public void resume() //rtcmanager.java
       ...
662. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
675. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
675. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
684. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
684. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
690. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
690. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
693. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
693. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
698. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
698. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
702. if (m_lowTtask != null)
```

[Table of Contents] [Description]

Path 4604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
702. if (m_lowTtask != null)
```

[Table of Contents] [Description]

Path 4605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
705. if (m_highTtask != null)
```

[Table of Contents] [Description]

Path 4606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
705. if (m_highTtask != null)
```

[Table of Contents] [Description]

Path 4607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
708. if (m_userTask != null)
```

[Table of Contents] [Description]

Path 4608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
708. if (m_userTask != null)
```

[Table of Contents] [Description]

Path 4609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
711. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
711. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
716. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
       ...
716. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public void clearErrors() //httppostinfo.java
110. if (m_errors != 0)
```

[Table of Contents] [Description]

Path 4614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
83. public void addNode(RTCNode node) //rtccategory.java
      ...
86. if (!m_nodes.contains(longnodeid))
```

[Table of Contents] [Description]

Path 4615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
83. public void addNode(RTCNode node) //rtccategory.java
      ...
86. if (!m_nodes.contains(longnodeid))
```

[Table of Contents] [Description]

Path 4616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void addNode(long nodeid) //rtccategory.java
      ...
99. if (!m_nodes.contains(longnodeid))
```

[Table of Contents] [Description]

Path 4617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
96. public void addNode(long nodeid) //rtccategory.java
      ...
99. if (!m_nodes.contains(longnodeid))
```

[Table of Contents] [Description]

Path 4618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public double getValue(String catLabel, long curTime, long
rollingWindow) //rtchashmap.java
      ...
229. if (valList == null || valList.size() == 0)
```

[Table of Contents] [Description]

Path 4619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
339. public int getServiceCount(long nodeid, String catLabel) //rtchashmap.java
      ...
349. if (node.belongsTo(catLabel))
```

[Table of Contents] [Description]

Path 4620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. public long getDownTime(String cat, long curTime, long
rollingWindow) //rtcnode.java
      ...
332. if (!m_categories.contains(cat))
```

[Table of Contents] [Description]

Path 4621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
353. public double getValue(String cat, long curTime, long
rollingWindow) //rtcnode.java
      ...
358. if (outageTime < 0)
```

[Table of Contents] [Description]

Path 4622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public int compareTo(Object o) //rtcnodekey.java
      ...
174. if (rc != 0)
```

[Table of Contents] [Description]

Path 4623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. public RTCNodeSvcTime(long lostt, long regainedt) //rtcnodesvctime.java
      ...
85. if (regainedt <= 0)
```

[Table of Contents] [Description]

Path 4624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
107. public void setRegainedTime(long t) //rtcnodesvctime.java
108. if (t <= 0)
```

[Table of Contents] [Description]

Path 4625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
157. public long getDownTime(long curTime, long rollingWindow) //rtcnodesvctime.java
      ...
161. if (curTime < m_svcLostTime)
```

[Table of Contents] [Description]

Path 4626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. public void run() //schedule.java
      ...
98. if (interval >= 0 && m_scheduled)
```

[Table of Contents] [Description]

Path 4627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
74. public void run() //schedule.java
      |...
      |98. if (interval >= 0 && m_scheduled)
```

[Table of Contents] [Description]

Path 4628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private void schedule(long interval) //schedule.java
      |128. if (interval >= 0 && m_scheduled)
```

[Table of Contents] [Description]

Path 4629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
127. private void schedule(long interval) //schedule.java
      |128. if (interval >= 0 && m_scheduled)
```

[Table of Contents] [Description]

Path 4630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
131. public synchronized Object peek() throws InterruptedException,
FifoQueueException //scheduler.java
      |132. if (m_hold == null)
```

[Table of Contents] [Description]

Path 4631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      |263. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      263. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      271. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      278. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      278. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      |282. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4638:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public synchronized void schedule(ReadyRunnable runnable, long
interval) //scheduler.java
      |...
      |282. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4639:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. public synchronized void start() //scheduler.java
      |329. if (m_worker != null)
```

[Table of Contents] [Description]

Path 4640:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. public synchronized void start() //scheduler.java
      |...
      |339. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4641:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
328. public synchronized void start() //scheduler.java
      |...
      |339. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4642:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
350. public synchronized void stop() //scheduler.java
     |351. if (m_worker == null)
```

[Table of Contents] [Description]

Path 4643:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
350. public synchronized void stop() //scheduler.java
     |...
     |360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4644:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
350. public synchronized void stop() //scheduler.java
     |...
     |360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4645:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public synchronized void pause() //scheduler.java
     |373. if (m_worker == null)
```

[Table of Contents] [Description]

Path 4646:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public synchronized void pause() //scheduler.java
     |...
     |376. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 4647:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public synchronized void pause() //scheduler.java
       ...
379. if (m_status == PAUSED)
```

[Table of Contents] [Description]

Path 4648:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. public synchronized void resume() //scheduler.java
395. if (m_worker == null)
```

[Table of Contents] [Description]

Path 4649:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. public synchronized void resume() //scheduler.java
       ...
398. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 4650:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. public synchronized void resume() //scheduler.java
       ...
401. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 4651:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
413. public synchronized int getStatus() //scheduler.java
414. if (m_worker != null && m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 4652:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
440. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4653:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
440. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4654:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4655:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4656:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4657:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4658:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
536. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4659:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
433. public void run() //scheduler.java
       ...
536. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4660:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
64. public void run() //scheduletest.java
      ...
66. if (isCallingAdjustSchedule())
```

[Table of Contents] [Description]

Path 4661:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
64. public void run() //scheduletest.java
      ...
66. if (isCallingAdjustSchedule())
```

[Table of Contents] [Description]

Path 4662:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. private void assertRunAndScheduled(long currentTime, long interval, int count,
int entryCount) //scheduletest.java
     ...
228. if (entryCount > 0)
```

[Table of Contents] [Description]

Path 4663:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. private void assertRunAndScheduled(long currentTime, long interval, int count,
int entryCount) //scheduletest.java
     ...
228. if (entryCount > 0)
```

[Table of Contents] [Description]

Path 4664:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
     ...
322. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4665:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
291. private void reinitializePrimarySnmpInterfaceHandler(Event
event) //broadcasteventprocessor.java
     ...
322. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4666:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
341. private void nodeGainedServiceHandler(Event
event) //broadcasteventprocessor.java
     ...
346. if (!event.getService().equals("SNMP") && !event.getService().equals
("SNMPv1") && !event.getService().equals("SNMPv2"))
```

[Table of Contents] [Description]

Path 4667:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      |...
      |373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4668:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      |...
      |373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4669:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      |...
      |392. if (parmValue == null)
```

[Table of Contents] [Description]

Path 4670:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      |...
      |435. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4671:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
```

```
      ...
435.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4672:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
451.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4673:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
371. private void primarySnmpInterfaceChangedHandler(Event
event) //broadcasteventprocessor.java
      ...
451.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4674:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
472.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4675:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      ...
472.  if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4676:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      476. if (event.getInterface() == null)
```

[Table of Contents] [Description]

Path 4677:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      493. if (parmValue == null)
```

[Table of Contents] [Description]

Path 4678:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      539. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4679:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      539. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4680:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      548. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4681:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      548. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4682:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      555. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4683:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
470. private void interfaceReparentedHandler(Event
event) //broadcasteventprocessor.java
      |...
      555. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4684:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
566. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
      |...
      580. if (!(tSvc.getNodeId() == nodeId))
```

[Table of Contents] [Description]

Path 4685:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
566. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
      |...
      601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4686:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
566. private void nodeDeletedHandler(Event event) //broadcasteventprocessor.java
       ...
601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4687:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
612. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
629. if (!(tSvc.getNodeId() == nodeId && addr.getHostName().equals(ipAddr)))
```

[Table of Contents] [Description]

Path 4688:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
612. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
650. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4689:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
612. private void interfaceDeletedHandler(Event
event) //broadcasteventprocessor.java
       ...
650. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4690:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
661. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
       |
```

```
...
666. if (!event.getService().equals("SNMP") && !event.getService().equals
("SNMPv1") && !event.getService().equals("SNMPv2"))
```

[Table of Contents] [Description]

Path 4691:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
661. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
...
685. if (!(tSvc.getNodeId() == nodeId && addr.getHostName().equals(ipAddr)) &&
tSvc.getServiceName().equals(svcName))
```

[Table of Contents] [Description]

Path 4692:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
661. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
...
706. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4693:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
661. private void serviceDeletedHandler(Event event) //broadcasteventprocessor.java
...
706. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4694:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
...
160. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4695:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
160. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4696:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
168. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4697:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
168. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4698:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
176. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4699:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
176. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4700:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
181. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4701:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. public void initialize(Map parameters) //latencythresholder.java
     ...
181. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4702:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     ...
210. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4703:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     ...
240. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4704:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     ...
240. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4705:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     |...
     |264. if (rs.wasNull())
```

[Table of Contents] [Description]

Path 4706:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     |...
     |269. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4707:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     |...
     |269. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4708:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     |...
     |280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4709:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
     |...
     |280. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4710:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |...
      |283. if (nodeId == -1)
```

[Table of Contents] [Description]

Path 4711:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |...
      |290. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4712:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |...
      |290. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4713:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |...
      |346. if (thresh.getType().equals(ThresholdEntity.HIGH_THRESHOLD))
```

[Table of Contents] [Description]

Path 4714:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |
```

```
...
346. if (thresh.getType().equals(ThresholdEntity.HIGH_THRESHOLD))
```

[Table of Contents] [Description]

Path 4715:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
348. else if (thresh.getType().equals(ThresholdEntity.LOW_THRESHOLD))
```

[Table of Contents] [Description]

Path 4716:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
348. else if (thresh.getType().equals(ThresholdEntity.LOW_THRESHOLD))
```

[Table of Contents] [Description]

Path 4717:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4718:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4719:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
     ...
418. if (tmp != null)
```

[Table of Contents] [Description]

Path 4720:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
     ...
425. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4721:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
     ...
425. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4722:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
     ...
431. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4723:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
403. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //latencythresholder.java
     ...
431. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4724:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
      ...
518. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4725:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
      ...
518. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4726:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
      ...
527. if (files == null)
```

[Table of Contents] [Description]

Path 4727:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
      ...
547. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4728:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
    ...
547. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4729:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
    ...
551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4730:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. private void checkRrdDir(File directory, int nodeId, InetAddress ipAddr, int
interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //latencythresholder.java
    ...
551. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4731:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
609. private Event createEvent(int nodeId, InetAddress ipAddr, double dsValue,
Threshold threshold, String uei, java.util.Date date) //latencythresholder.java
    ...
612. if (threshold == null)
```

[Table of Contents] [Description]

Path 4732:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
    ...
186. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4733:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     |...
186. if (log.isEnabledFor(Priority.WARN))
```

[Table of Contents] [Description]

Path 4734:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     |...
194. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4735:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     |...
194. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4736:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     |...
199. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4737:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
176. public void initialize(Map parameters) //snmpthresholder.java
     |...
199. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4738:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
228. if (iface.getType() != NetworkInterface.TYPE_IPV4)
```

[Table of Contents] [Description]

Path 4739:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
289. if (thresh.getType().equals(ThresholdEntity.HIGH_THRESHOLD))
```

[Table of Contents] [Description]

Path 4740:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
289. if (thresh.getType().equals(ThresholdEntity.HIGH_THRESHOLD))
```

[Table of Contents] [Description]

Path 4741:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
291. else if (thresh.getType().equals(ThresholdEntity.LOW_THRESHOLD))
```

[Table of Contents] [Description]

Path 4742:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |
```

```
...
291. else if (thresh.getType().equals(ThresholdEntity.LOW_THRESHOLD))
```

[Table of Contents] [Description]

Path 4743:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
299. if (thresh.getDsType().equals("node"))
```

[Table of Contents] [Description]

Path 4744:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
299. if (thresh.getDsType().equals("node"))
```

[Table of Contents] [Description]

Path 4745:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
301. else if (thresh.getDsType().equals("if"))
```

[Table of Contents] [Description]

Path 4746:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
301. else if (thresh.getDsType().equals("if"))
```

[Table of Contents] [Description]

Path 4747:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
331. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4748:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
331. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4749:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
355. if (rs.wasNull())
```

[Table of Contents] [Description]

Path 4750:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
358. if (rs.wasNull())
```

[Table of Contents] [Description]

Path 4751:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     ...
401. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4752:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |
      ...
      366. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4753:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |
      ...
      366. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4754:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |
      ...
      377. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4755:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |
      ...
      377. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4756:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |
```

```
...
385. if (nodeId == -1)
```

[Table of Contents] [Description]

Path 4757:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
388. if (primaryIfIndex == -1)
```

[Table of Contents] [Description]

Path 4758:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
394. if (isSnmpPrimary != DbIpInterfaceEntry.SNMP_PRIMARY)
```

[Table of Contents] [Description]

Path 4759:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
401. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4760:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
401. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4761:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
     427. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4762:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
     |...
     427. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4763:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
     |...
     467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4764:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
     |...
     467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4765:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
     |...
     473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4766:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
454. public int check(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpthresholder.java
      ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4767:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
      ...
586. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4768:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
      ...
586. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4769:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
      ...
595. if (files == null)
```

[Table of Contents] [Description]

Path 4770:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
633. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4771:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
633. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4772:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
641. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4773:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
641. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4774:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
     ...
```

```
645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4775:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
...
645. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4776:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
...
655. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4777:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void checkNodeDir(File directory, Integer nodeId, InetAddress primary,
int interval, int range, Date date, Map thresholdMap, Events events) throws
IllegalArgumentException //snmpthresholder.java
...
655. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4778:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
...
720. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4779:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     |...
     |720. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4780:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     |...
     |724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4781:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     |...
     |724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4782:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
     |...
     |769. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4783:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
    ...
769. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4784:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
    ...
784. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4785:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
    ...
784. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4786:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
    ...
832. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4787:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
    ...
```

```
832. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4788:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
...
858. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4789:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
...
858. if (dbConn != null)
```

[Table of Contents] [Description]

Path 4790:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
...
861. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4791:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
712. private void checkIfDir(File directory, Integer nodeId, InetAddress primary,
int interval, Date date, Map baseIfThresholdMap, Map allIfThresholdMap, Events
events) throws IllegalArgumentException //snmpthresholder.java
...
861. if (log.isEnabledFor(Priority.INFO))
```

[Table of Contents] [Description]

Path 4792:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
922. private Event createEvent(Integer nodeId, InetAddress primary, Map ifDataMap,
double dsValue, Threshold threshold, String uei, java.util.Date date, String
label) //snmpthresholder.java
      ...
925. if (nodeId == null || primary == null || threshold == null)
```

[Table of Contents] [Description]

Path 4793:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
922. private Event createEvent(Integer nodeId, InetAddress primary, Map ifDataMap,
double dsValue, Threshold threshold, String uei, java.util.Date date, String
label) //snmpthresholder.java
      ...
950. if (primary != null)
```

[Table of Contents] [Description]

Path 4794:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
922. private Event createEvent(Integer nodeId, InetAddress primary, Map ifDataMap,
double dsValue, Threshold threshold, String uei, java.util.Date date, String
label) //snmpthresholder.java
      ...
950. if (primary != null)
```

[Table of Contents] [Description]

Path 4795:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4796:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4797:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
175. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4798:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
175. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4799:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
179. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4800:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
179. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4801:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
183. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4802:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
183. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4803:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
194. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4804:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
194. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4805:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
198. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4806:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
198. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4807:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
202. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4808:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
202. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4809:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
212. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4810:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
212. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4811:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
      ...
216. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4812:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
216. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4813:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
220. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4814:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
220. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4815:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
229. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4816:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
229. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4817:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
250. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4818:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
250. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4819:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
254. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4820:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
254. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4821:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
258. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4822:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
258. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4823:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
262. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4824:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
262. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4825:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
266. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4826:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
      ...
266. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4827:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
274. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4828:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
274. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4829:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
282. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4830:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
      ...
282. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 4831:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
288. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4832:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
288. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4833:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4834:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4835:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
335. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4836:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
335. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4837:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
340. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4838:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
340. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4839:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
352. public void run() //threshd.java
...
357. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4840:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
352. public void run() //threshd.java
...
357. if (log.isEnabledFor(Priority.ERROR))
```

[Table of Contents] [Description]

Path 4841:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4842:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
373. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4843:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
378. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4844:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
378. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4845:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4846:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
```

```
...
388. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4847:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
...
404. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4848:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
...
404. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4849:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
...
410. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4850:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
...
410. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4851:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
```

```
      ...
415. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4852:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
      ...
415. if (log.isEnabledFor(Priority.FATAL))
```

[Table of Contents] [Description]

Path 4853:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
      ...
424. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4854:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
395. public synchronized void start() //threshd.java
      ...
424. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4855:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public synchronized void stop() //threshd.java
      ...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4856:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public synchronized void stop() //threshd.java
```

```
...
439. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4857:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
446. public synchronized void pause() //threshd.java
     447. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 4858:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
446. public synchronized void pause() //threshd.java
     ...
     455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4859:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
446. public synchronized void pause() //threshd.java
     ...
     455. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4860:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public synchronized void resume() //threshd.java
     463. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 4861:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public synchronized void resume() //threshd.java
     ...
```

```
471. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4862:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
462. public synchronized void resume() //threshd.java
     ...
471. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4863:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
     ...
552. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4864:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
     ...
552. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4865:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
     ...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4866:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
```

```
...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4867:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
...
588. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4868:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
...
588. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4869:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
...
597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4870:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
...
597. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4871:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
```

```
existing) //threshd.java
    ...
636. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4872:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
    ...
636. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4873:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
    ...
653. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4874:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
    ...
653. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4875:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
    ...
707. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4876:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
618. void scheduleInterface(int nodeId, String ipAddress, String svcName, boolean
existing) //threshd.java
     ...
707. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4877:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. ThresholdableService(int dbNodeId, InetAddress address, String svcName,
org.opennms.netmgt.config.threshd.Package pkg) //thresholdableservice.java
     ...
179. if (svc == null)
```

[Table of Contents] [Description]

Path 4878:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. public boolean isReady() //thresholdableservice.java
     ...
264. if (!Threshd.getInstance().isSchedulingCompleted())
```

[Table of Contents] [Description]

Path 4879:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
288. private void sendEvent(String uei) //thresholdableservice.java
     ...
312. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4880:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
288. private void sendEvent(String uei) //thresholdableservice.java
     ...
312. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4881:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
       ...
329. if (processUpdates() == ABORT_THRESHOLD_CHECK)
```

[Table of Contents] [Description]

Path 4882:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
       ...
346. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4883:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
       ...
346. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4884:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
       ...
364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4885:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
324. public void run() //thresholdableservice.java
       ...
364. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4886:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. private boolean scheduledOutage() //thresholdableservice.java
       ...
427. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 4887:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
407. private boolean scheduledOutage() //thresholdableservice.java
       ...
427. if (ThreadCategory.getInstance(getClass()).isDebugEnabled())
```

[Table of Contents] [Description]

Path 4888:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
451. if (!m_updates.hasUpdates())
```

[Table of Contents] [Description]

Path 4889:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4890:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4891:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
472. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4892:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
472. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4893:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4894:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4895:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
490. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4896:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
490. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4897:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
510. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4898:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
510. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4899:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
514. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4900:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
444. private boolean processUpdates() //thresholdableservice.java
       ...
514. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4901:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
103. void setHighThreshold(Threshold threshold) //thresholdentity.java
     |104. if (m_highThreshold != null)
```

[Table of Contents] [Description]

Path 4902:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
112. void setLowThreshold(Threshold threshold) //thresholdentity.java
     |113. if (m_lowThreshold != null)
```

[Table of Contents] [Description]

Path 4903:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. String getDatasourceName() //thresholdentity.java
     |141. if (this.hasHighThreshold())
```

[Table of Contents] [Description]

Path 4904:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
140. String getDatasourceName() //thresholdentity.java
     |...
     |143. else if (this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4905:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. String getDatasourceType() //thresholdentity.java
     |153. if (this.hasHighThreshold())
```

[Table of Contents] [Description]

Path 4906:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. String getDatasourceType() //thresholdentity.java
     |...
     |155. else if (this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4907:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. double getHighValue() //thresholdentity.java
     |165. if (m_highThreshold == null)
```

[Table of Contents] [Description]

Path 4908:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
174. double getLowValue() //thresholdentity.java
     |175. if (m_lowThreshold == null)
```

[Table of Contents] [Description]

Path 4909:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
184. double getHighRearm() //thresholdentity.java
     |185. if (m_highThreshold == null)
```

[Table of Contents] [Description]

Path 4910:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
194. double getLowRearm() //thresholdentity.java
     |195. if (m_lowThreshold == null)
```

[Table of Contents] [Description]

Path 4911:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. int getHighTrigger() //thresholdentity.java
     |205. if (m_highThreshold == null)
```

[Table of Contents] [Description]

Path 4912:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
214. int getLowTrigger() //thresholdentity.java
     |215. if (m_lowThreshold == null)
```

[Table of Contents] [Description]

Path 4913:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. String getDatasourceLabel() //thresholdentity.java
     |225. if (this.hasHighThreshold())
```

[Table of Contents] [Description]

Path 4914:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
224. String getDatasourceLabel() //thresholdentity.java
     |...
     |227. else if (this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4915:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public Object clone() //thresholdentity.java
     |...
     |244. if (this.hasHighThreshold())
```

[Table of Contents] [Description]

Path 4916:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public Object clone() //thresholdentity.java
     |...
     |244. if (this.hasHighThreshold())
```

[Table of Contents] [Description]

Path 4917:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public Object clone() //thresholdentity.java
       ...
246. if (this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4918:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
242. public Object clone() //thresholdentity.java
       ...
246. if (this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4919:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
259. public String toString() //thresholdentity.java
       ...
264. if (!this.hasHighThreshold() && !this.hasLowThreshold())
```

[Table of Contents] [Description]

Path 4920:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
       ...
306. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4921:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
       ...
306. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4922:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
319. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4923:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
319. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4924:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4925:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
324. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4926:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4927:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4928:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4929:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
344. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4930:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4931:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
360. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4932:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
365. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4933:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
365. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4934:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4935:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
376. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4936:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
385. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4937:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
385. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4938:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
393. if (lowTriggered && highTriggered)
```

[Table of Contents] [Description]

Path 4939:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
395. else if (lowTriggered)
```

[Table of Contents] [Description]

Path 4940:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
397. else if (highTriggered)
```

[Table of Contents] [Description]

Path 4941:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
     ...
399. else if (lowRearmed && highRearmed)
```

[Table of Contents] [Description]

Path 4942:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
       ...
401. else if (lowRearmed)
```

[Table of Contents] [Description]

Path 4943:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. int evaluate(double dsValue) //thresholdentity.java
       ...
403. else if (highRearmed)
```

[Table of Contents] [Description]

Path 4944:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
116. void setAttribute(String property, Object value) //thresholderupdates.java
       117. if (m_properties == null)
```

[Table of Contents] [Description]

Path 4945:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
128. Object getAttribute(String property) //thresholderupdates.java
       ...
130. if (m_properties != null)
```

[Table of Contents] [Description]

Path 4946:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
95. if (eventUei == null)
```

[Table of Contents] [Description]

Path 4947:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
98. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4948:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
98. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4949:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
106. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4950:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
106. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4951:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4952:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
      113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4953:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
      120. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4954:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
      120. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4955:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      ...
      98. if (value instanceof String)
```

[Table of Contents] [Description]

Path 4956:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      ...
      100. else if (value instanceof Number)
```

[Table of Contents] [Description]

Path 4957:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      |...
      |102. else if (value instanceof SnmpInt32)
```

[Table of Contents] [Description]

Path 4958:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      |...
      |104. else if (value instanceof SnmpUInt32)
```

[Table of Contents] [Description]

Path 4959:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      |...
      |106. else if (value instanceof SnmpCounter64)
```

[Table of Contents] [Description]

Path 4960:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      |...
      |108. else if (value instanceof SnmpIPAddress)
```

[Table of Contents] [Description]

Path 4961:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
      |...
```

```
110. else if (value instanceof SnmpOctetString)
```

[Table of Contents] [Description]

Path 4962:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
    ...
112. else if (value instanceof SnmpObjectId)
```

[Table of Contents] [Description]

Path 4963:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
90. public static String toString(String encoding, Object
value) //eventconstants.java
    ...
115. if (value instanceof String)
```

[Table of Contents] [Description]

Path 4964:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
    ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4965:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
    ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4966:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     ...
359. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4967:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     ...
359. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4968:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     ...
372. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4969:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     ...
372. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4970:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
     ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4971:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
       ...
379. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4972:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public synchronized void start() //trapd.java
       ...
425. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4973:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public synchronized void start() //trapd.java
       ...
425. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4974:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public synchronized void start() //trapd.java
       ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4975:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
418. public synchronized void start() //trapd.java
       ...
432. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4976:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
     441. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 4977:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
     ...
     448. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4978:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
     ...
     448. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4979:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
     ...
     453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4980:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
     ...
     453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4981:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
```

```
    ...
458. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4982:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
440. public void pause() //trapd.java
    ...
458. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4983:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
466. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 4984:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
    ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4985:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
    ...
473. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4986:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
```

```
...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4987:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
...
478. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4988:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
...
483. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4989:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
465. public void resume() //trapd.java
...
483. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4990:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4991:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
```

```
...
497. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4992:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4993:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
501. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4994:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
506. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4995:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
506. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4996:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
```

```
...
509. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4997:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
509. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4998:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
513. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 4999:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
513. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5000:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
...
521. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5001:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
491. public synchronized void stop() //trapd.java
```

```
      ...
      521. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5002:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. static synchronized void dataSourceSync() throws SQLException //trapdipmgr.java
      ...
      119. if (c != null)
```

[Table of Contents] [Description]

Path 5003:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
93. static synchronized void dataSourceSync() throws SQLException //trapdipmgr.java
      ...
      119. if (c != null)
```

[Table of Contents] [Description]

Path 5004:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. static synchronized String getNodeId(String addr) //trapdipmgr.java
      136. if (addr == null)
```

[Table of Contents] [Description]

Path 5005:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
152. static synchronized String setNodeId(String addr, String
nodeid) //trapdipmgr.java
      153. if (addr == null || nodeid == null)
```

[Table of Contents] [Description]

Path 5006:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. static String removeNodeId(String addr) //trapdipmgr.java
      169. if (addr == null)
```

[Table of Contents] [Description]

Path 5007:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
290. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5008:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
290. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5009:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
297. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5010:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
297. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5011:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
322. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5012:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
322. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5013:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5014:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
330. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5015:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5016:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
335. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5017:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5018:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5019:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     ...
402. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5020:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     ...
402. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5021:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
     ...
462. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5022:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
462. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5023:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5024:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
560. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5025:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5026:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
566. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5027:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
       ...
601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5028:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
       ...
601. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5029:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
       ...
724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5030:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
       ...
724. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5031:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
       ...
730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5032:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
578. private void process(Trapd.V1TrapInformation info) //trapqueueprocessor.java
      ...
730. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5033:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
828. public synchronized void start() //trapqueueprocessor.java
      ...
831. if (m_worker != null)
```

[Table of Contents] [Description]

Path 5034:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
828. public synchronized void start() //trapqueueprocessor.java
      ...
839. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5035:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
828. public synchronized void start() //trapqueueprocessor.java
      ...
839. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5036:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
846. public synchronized void pause() //trapqueueprocessor.java
      847. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 5037:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
857. public synchronized void resume() //trapqueueprocessor.java
     |858. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 5038:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
874. public synchronized void stop() //trapqueueprocessor.java
     |875. if (m_worker == null)
```

[Table of Contents] [Description]

Path 5039:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
897. public synchronized int getStatus() //trapqueueprocessor.java
     |898. if (m_worker != null && !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 5040:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java
     |...
     |82. if (replacedChar)
```

[Table of Contents] [Description]

Path 5041:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java
     |...
     |82. if (replacedChar)
```

[Table of Contents] [Description]

Path 5042:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
117. if (Pattern.matches(".*-cef.*", rs.getString("snmpifdescr")))
```

[Table of Contents] [Description]

Path 5043:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
57. static long byteToLong(byte b) //ipsorter.java
    ...
59. if (r < 0)
```

[Table of Contents] [Description]

Path 5044:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. public static long convertToLong(byte[] addr) //ipsorter.java
83. if (addr == null)
```

[Table of Contents] [Description]

Path 5045:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
    ...
49. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5046:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
    ...
49. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5047:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
      ...
59. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5048:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
      ...
59. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5049:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
      ...
65. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5050:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
45. public static boolean isIpValid(String ipAddr) //ipvalidator.java
      ...
65. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5051:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
89. private String argsToString(Object[] values) //jdbctemplate.java
      ...
92. if (i != 0)
```

[Table of Contents] [Description]

Path 5052:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. private void doExecute(Object values[]) throws SQLException //jdbctemplate.java
     ...
113. if (stmt != null)
```

[Table of Contents] [Description]

Path 5053:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. private void doExecute(Object values[]) throws SQLException //jdbctemplate.java
     ...
113. if (stmt != null)
```

[Table of Contents] [Description]

Path 5054:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
272. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5055:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
272. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5056:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
     ...
341. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5057:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
     ...
341. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5058:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
     ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5059:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
     ...
353. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5060:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
     ...
358. if (label.length() > MAX_NODE_LABEL_LENGTH)
```

[Table of Contents] [Description]

Path 5061:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
```

```
...
436. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5062:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
436. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5063:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
457. if (netbiosName.length() > MAX_NODE_LABEL_LENGTH)
```

[Table of Contents] [Description]

Path 5064:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5065:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
460. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5066:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |...
      |485. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5067:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |...
      |485. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5068:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |...
      |494. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5069:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |...
      |494. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5070:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |...
      |519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5071:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |
      ...
      519. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5072:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |
      ...
      528. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5073:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |
      ...
      528. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5074:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |
      ...
      558. if (primaryHostName.length() > MAX_NODE_LABEL_LENGTH)
```

[Table of Contents] [Description]

Path 5075:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      |
```

```
...
561. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5076:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
561. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5077:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
576. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5078:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
576. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5079:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
597. if (primarySysName.length() > MAX_NODE_LABEL_LENGTH)
```

[Table of Contents] [Description]

Path 5080:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
600. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5081:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
600. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5082:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
610. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5083:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
610. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5084:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
      ...
646. if (hostName == null || hostName.equals(ipv4Addr.toString()))
```

[Table of Contents] [Description]

Path 5085:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
     ...
646. if (hostName == null || hostName.equals(ipv4Addr.toString()))
```

[Table of Contents] [Description]

Path 5086:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
     ...
646. if (hostName == null || hostName.equals(ipv4Addr.toString()))
```

[Table of Contents] [Description]

Path 5087:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
     ...
651. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5088:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
631. private static void loadAddressList(ResultSet rs, List ipv4AddrList, List
ipHostNameList) throws SQLException //nodelabel.java
     ...
651. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5089:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
```

```
...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5090:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
...
682. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5091:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
...
695. if (current < 0)
```

[Table of Contents] [Description]

Path 5092:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
...
698. if (primary < 0)
```

[Table of Contents] [Description]

Path 5093:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
...
701. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5094:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
     |...
     |701. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5095:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
     |...
     |708. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5096:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
     |...
     |708. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5097:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
     |...
     |715. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5098:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
669. private static IPv4Address selectPrimaryAddress(List ipv4AddrList, String
method) //nodelabel.java
     |...
     |715. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5099:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
44. public Querier(DbConnectionFactory db, String sql, RowProcessor
rowProcessor) //querier.java
      |
      ...
      46. if (rowProcessor == null)
```

[Table of Contents] [Description]

Path 5100:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
61. protected void executeStmt(PreparedStatement stmt) throws
SQLException //querier.java
      |
      ...
      69. if (rs != null)
```

[Table of Contents] [Description]

Path 5101:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
61. protected void executeStmt(PreparedStatement stmt) throws
SQLException //querier.java
      |
      ...
      69. if (rs != null)
```

[Table of Contents] [Description]

Path 5102:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
68. public void snmpReceivedPdu(SnmpSession sess, int command, SnmpPduPacket
pkt) //snmpresponsehandler.java
      |
      ...
      70. if (((SnmpPduRequest) pkt).getErrorStatus() == SnmpPduPacket.ErrNoError)
```

[Table of Contents] [Description]

Path 5103:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
      |
```

```
...
89. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5104:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
...
89. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5105:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
...
102. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5106:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
...
102. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5107:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
...
110. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5108:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
70. public static SocketChannel getConnectedSocketChannel(InetAddress host, int
port, int timeout) throws IOException, InterruptedException //socketchannelutil.java
    ...
110. if (sChannel.socket() != null)
```

[Table of Contents] [Description]

Path 5109:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
    ...
64. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5110:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
    ...
64. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5111:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
    ...
125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5112:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
    ...
125. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5113:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      ...
146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5114:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      ...
146. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5115:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public void pause() //vacuumd.java
      151. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 5116:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public void resume() //vacuumd.java
      169. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 5117:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
250. public void runUpdate(String sql) //vacuumd.java
      ...
265. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5118:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
250. public void runUpdate(String sql) //vacuumd.java
       ...
265. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5119:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
142. if (isTracing)
```

[Table of Contents] [Description]

Path 5120:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
142. if (isTracing)
```

[Table of Contents] [Description]

Path 5121:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
150. if (isTracing)
```

[Table of Contents] [Description]

Path 5122:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
150. if (isTracing)
```

[Table of Contents] [Description]

Path 5123:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
157. if (isTracing)
```

[Table of Contents] [Description]

Path 5124:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
157. if (isTracing)
```

[Table of Contents] [Description]

Path 5125:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
166. if (isTracing)
```

[Table of Contents] [Description]

Path 5126:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
166. if (isTracing)
```

[Table of Contents] [Description]

Path 5127:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
184. if (isTracing)
```

[Table of Contents] [Description]

Path 5128:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
184. if (isTracing)
```

[Table of Contents] [Description]

Path 5129:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
207. if (isTracing)
```

[Table of Contents] [Description]

Path 5130:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
207. if (isTracing)
```

[Table of Contents] [Description]

Path 5131:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
225. if (isTracing)
```

[Table of Contents] [Description]

Path 5132:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
       ...
225. if (isTracing)
```

[Table of Contents] [Description]

Path 5133:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
     ...
261. if (isTracing)
```

[Table of Contents] [Description]

Path 5134:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
     ...
261. if (isTracing)
```

[Table of Contents] [Description]

Path 5135:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
     ...
278. if (isTracing)
```

[Table of Contents] [Description]

Path 5136:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
     ...
278. if (isTracing)
```

[Table of Contents] [Description]

Path 5137:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
     ...
301. if (isTracing)
```

[Table of Contents] [Description]

Path 5138:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
       ...
301. if (isTracing)
```

[Table of Contents] [Description]

Path 5139:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
       ...
319. if (isTracing)
```

[Table of Contents] [Description]

Path 5140:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
       ...
319. if (isTracing)
```

[Table of Contents] [Description]

Path 5141:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
       ...
357. if (isTracing)
```

[Table of Contents] [Description]

Path 5142:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
       ...
357. if (isTracing)
```

[Table of Contents] [Description]

Path 5143:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     ...
375. if (isTracing)
```

[Table of Contents] [Description]

Path 5144:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     ...
375. if (isTracing)
```

[Table of Contents] [Description]

Path 5145:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     ...
398. if (isTracing)
```

[Table of Contents] [Description]

Path 5146:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     ...
398. if (isTracing)
```

[Table of Contents] [Description]

Path 5147:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
     ...
418. if (isTracing)
```

[Table of Contents] [Description]

Path 5148:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
       ...
418. if (isTracing)
```

[Table of Contents] [Description]

Path 5149:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
147. if (eventUei == null)
```

[Table of Contents] [Description]

Path 5150:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5151:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
150. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5152:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
59. public boolean isValid() //descrvalues.java
      60. if ((descr != null) && (severity > 0))
```

[Table of Contents] [Description]

Path 5153:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
178. public PortValues parsePort(String portString) throws
IllegalArgumentException //nessusparser.java
     ...
200. if (retval.isValid())
```

[Table of Contents] [Description]

Path 5154:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public DescrValues parseDescr(String descr) throws
IllegalArgumentException //nessusparser.java
     ...
333. if (retval.isValid())
```

[Table of Contents] [Description]

Path 5155:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
202. public NessusScan(NessusScanConfiguration newConfig) throws
IllegalArgumentException //nessusscan.java
203. if (newConfig.isValid())
```

[Table of Contents] [Description]

Path 5156:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
302. if (!found)
```

[Table of Contents] [Description]

Path 5157:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
     ...
322. if (!found)
```

[Table of Contents] [Description]

Path 5158:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
345. if (!found)
```

[Table of Contents] [Description]

Path 5159:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
363. if (!found)
```

[Table of Contents] [Description]

Path 5160:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
381. if (!found)
```

[Table of Contents] [Description]

Path 5161:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
406. if (!found)
```

[Table of Contents] [Description]

Path 5162:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
504. if (nessusSocket != null)
```

[Table of Contents] [Description]

Path 5163:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
225. public void run() //nessusscan.java
       ...
504. if (nessusSocket != null)
```

[Table of Contents] [Description]

Path 5164:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
525. private String buildPreferencesString() //nessusscan.java
       ...
528. if (safeChecks[config.scanLevel])
```

[Table of Contents] [Description]

Path 5165:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
632. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5166:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
632. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5167:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
632. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5168:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
638. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5169:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
638. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5170:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
638. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5171:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
684. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5172:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
684. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5173:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
      ...
684. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5174:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
      ...
714. if (!pluginLogmsg.equals(""))
```

[Table of Contents] [Description]

Path 5175:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
      ...
729. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5176:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
      ...
739. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5177:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
      ...
739. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5178:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
739. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5179:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
746. if (descrvals.cveEntry != null)
```

[Table of Contents] [Description]

Path 5180:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
746. if (descrvals.cveEntry != null)
```

[Table of Contents] [Description]

Path 5181:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
746. if (descrvals.cveEntry != null)
```

[Table of Contents] [Description]

Path 5182:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
805. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5183:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
805. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5184:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
805. if (portvals.port > 0)
```

[Table of Contents] [Description]

Path 5185:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
811. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5186:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
811. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5187:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
       ...
811. if (portvals.protocol != null)
```

[Table of Contents] [Description]

Path 5188:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
857. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5189:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
857. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5190:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
857. if (nodeId > 0)
```

[Table of Contents] [Description]

Path 5191:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
882. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5192:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
882. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5193:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
882. if (portvals.port >= 0)
```

[Table of Contents] [Description]

Path 5194:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
557. private int processScanMessage(String message) //nessusscan.java
     ...
917. if (lastPlugin == totalPlugins)
```

[Table of Contents] [Description]

Path 5195:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. public FifoQueue readLines(InputStream in) //nessusscan.java
     ...
1008. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5196:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. public FifoQueue readLines(InputStream in) //nessusscan.java
     ...
1008. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5197:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. public FifoQueue readLines(InputStream in) //nessusscan.java
     ...
1016. if (bytesInThisRead < 0)
```

[Table of Contents] [Description]

Path 5198:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
990. public FifoQueue readLines(InputStream in) //nessusscan.java
       ...
1025. if (alreadyRecdData != null)
```

[Table of Contents] [Description]

Path 5199:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. boolean isTimeForRescan() //nessusscanconfiguration.java
       152. if (System.currentTimeMillis() >= (lastScan.getTime() + interval))
```

[Table of Contents] [Description]

Path 5200:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
53. public boolean isValid() //portvalues.java
      54. if ((protocol != null))
```

[Table of Contents] [Description]

Path 5201:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
       ...
143. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5202:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
       ...
143. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5203:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
      |...
      |149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5204:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
129. Scheduler(FifoQueue rescanQ) throws SQLException //scheduler.java
      |...
      |149. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5205:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
347. public synchronized void start() //scheduler.java
      |348. if (m_worker != null)
```

[Table of Contents] [Description]

Path 5206:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
367. public synchronized void stop() //scheduler.java
      |368. if (m_worker == null)
```

[Table of Contents] [Description]

Path 5207:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
387. public synchronized void pause() //scheduler.java
      |388. if (m_worker == null)
```

[Table of Contents] [Description]

Path 5208:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
387. public synchronized void pause() //scheduler.java
```

```
...
391. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 5209:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
387. public synchronized void pause() //scheduler.java
...
394. if (m_status == PAUSED)
```

[Table of Contents] [Description]

Path 5210:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
409. public synchronized void resume() //scheduler.java
410. if (m_worker == null)
```

[Table of Contents] [Description]

Path 5211:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
409. public synchronized void resume() //scheduler.java
...
413. if (m_status == STOPPED || m_status == STOP_PENDING)
```

[Table of Contents] [Description]

Path 5212:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
409. public synchronized void resume() //scheduler.java
...
416. if (m_status == RUNNING)
```

[Table of Contents] [Description]

Path 5213:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
428. public synchronized int getStatus() //scheduler.java
```

```
429. if (m_worker != null && m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 5214:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
448. public void run() //scheduler.java
     ...
502. if (addressInfo.isScheduled())
```

[Table of Contents] [Description]

Path 5215:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
448. public void run() //scheduler.java
     ...
506. if (!addressInfo.isTimeForRescan())
```

[Table of Contents] [Description]

Path 5216:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
     ...
219. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5217:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
     ...
219. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5218:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
```

```
...
226. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5219:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
...
226. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5220:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5221:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
173. public void start() //vulnscand.java
...
244. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5222:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void pause() //vulnscand.java
...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5223:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void pause() //vulnscand.java
```

```
...
259. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5224:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public void resume() //vulnscand.java
     264. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 5225:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public void resume() //vulnscand.java
       ...
       275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5226:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. public void resume() //vulnscand.java
       ...
       275. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5227:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
       110. if (eventUei == null)
```

[Table of Contents] [Description]

Path 5228:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
       113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5229:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
113. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5230:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
121. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5231:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
121. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5232:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
127. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5233:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
127. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5234:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
104. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5235:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
104. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5236:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
109. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5237:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
109. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5238:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
114. if (event.hasDbid())
```

[Table of Contents] [Description]

Path 5239:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
119. if (event.hasNodeid())
```

[Table of Contents] [Description]

Path 5240:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
126. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5241:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
98. private void processEvent(Event event) //eventqueueprocessor.java
      ...
126. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5242:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      ...
182. if (parmValue == null)
```

[Table of Contents] [Description]

Path 5243:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      ...
190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5244:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5245:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      201. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5246:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      201. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5247:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
      ...
      205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5248:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |
```

```
      ...
205. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5249:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      ...
209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5250:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      ...
209. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5251:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private void pushBackEvent(Event event) //eventqueueprocessor.java
      ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5252:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. private void pushBackEvent(Event event) //eventqueueprocessor.java
      ...
260. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5253:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
331. public synchronized void start() //eventqueueprocessor.java
      |...
      |334. if (m_worker != null)
```

[Table of Contents] [Description]

Path 5254:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
331. public synchronized void start() //eventqueueprocessor.java
      |...
      |342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5255:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
331. public synchronized void start() //eventqueueprocessor.java
      |...
      |342. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5256:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
349. public synchronized void pause() //eventqueueprocessor.java
      |350. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 5257:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. public synchronized void resume() //eventqueueprocessor.java
      |361. if (m_worker == null || m_worker.isAlive() == false)
```

[Table of Contents] [Description]

Path 5258:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
377. public synchronized void stop() //eventqueueprocessor.java
      |
```

```
378. if (m_worker == null)
```

[Table of Contents] [Description]

Path 5259:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
400. public synchronized int getStatus() //eventqueueprocessor.java
     401. if (m_worker != null && !m_worker.isAlive())
```

[Table of Contents] [Description]

Path 5260:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
     ...
     115. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5261:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
     ...
     115. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5262:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
     ...
     122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5263:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
     ...
```

```
122. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5264:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
110. public synchronized void init() //xmlrpcd.java
       ...
132. if (verifyServer)
```

[Table of Contents] [Description]

Path 5265:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void start() //xmlrpcd.java
       ...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5266:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void start() //xmlrpcd.java
       ...
167. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5267:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void start() //xmlrpcd.java
       ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5268:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
160. public synchronized void start() //xmlrpcd.java
```

```
...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5269:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
     183. if (m_status != RUNNING)
```

[Table of Contents] [Description]

Path 5270:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
     ...
     190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5271:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
     ...
     190. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5272:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
     ...
     195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5273:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
```

```
...
195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5274:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
...
200. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5275:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
182. public void pause() //xmlrpcd.java
...
200. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5276:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
208. if (m_status != PAUSED)
```

[Table of Contents] [Description]

Path 5277:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
...
215. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5278:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
```

```
...
215. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5279:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
...
220. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5280:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
...
220. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5281:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
...
225. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5282:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
207. public void resume() //xmlrpcd.java
...
225. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5283:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
```

```
...
239. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5284:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
...
239. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5285:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
...
242. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5286:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
...
242. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5287:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
...
250. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5288:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
233. public synchronized void stop() //xmlrpcd.java
```

```
...
250. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5289:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
151. public XmlRpcNotifier(XmlrpcServer[] rpcServers, int retries, int elapseTime,
boolean verifyServer, String localServer) //xmlrpcnotifier.java
...
158. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5290:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public boolean sendServiceDownEvent(Event event) //xmlrpcnotifier.java
...
239. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5291:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public boolean sendServiceDownEvent(Event event) //xmlrpcnotifier.java
...
239. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5292:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
231. public boolean sendServiceDownEvent(Event event) //xmlrpcnotifier.java
...
239. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5293:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean sendServiceUpEvent(Event event) //xmlrpcnotifier.java
     ...
260. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5294:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean sendServiceUpEvent(Event event) //xmlrpcnotifier.java
     ...
260. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5295:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
252. public boolean sendServiceUpEvent(Event event) //xmlrpcnotifier.java
     ...
260. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5296:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public boolean sendInterfaceDownEvent(Event event) //xmlrpcnotifier.java
     ...
279. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5297:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public boolean sendInterfaceDownEvent(Event event) //xmlrpcnotifier.java
     ...
279. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5298:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public boolean sendInterfaceDownEvent(Event event) //xmlrpcnotifier.java
     ...
279. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5299:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
292. public boolean sendInterfaceUpEvent(Event event) //xmlrpcnotifier.java
     ...
299. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5300:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
292. public boolean sendInterfaceUpEvent(Event event) //xmlrpcnotifier.java
     ...
299. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5301:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
292. public boolean sendInterfaceUpEvent(Event event) //xmlrpcnotifier.java
     ...
299. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5302:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
310. public boolean sendNodeDownEvent(Event event) //xmlrpcnotifier.java
     ...
315. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5303:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
310. public boolean sendNodeDownEvent(Event event) //xmlrpcnotifier.java
      ...
315. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5304:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
310. public boolean sendNodeDownEvent(Event event) //xmlrpcnotifier.java
      ...
315. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5305:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
327. public boolean sendNodeUpEvent(Event event) //xmlrpcnotifier.java
      ...
332. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5306:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
327. public boolean sendNodeUpEvent(Event event) //xmlrpcnotifier.java
      ...
332. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5307:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
327. public boolean sendNodeUpEvent(Event event) //xmlrpcnotifier.java
      ...
332. if (m_verifyServer)
```

[Table of Contents] [Description]

Path 5308:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
     ...
357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5309:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
     ...
357. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5310:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
     ...
373. if (dbConn != null)
```

[Table of Contents] [Description]

Path 5311:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
348. private String getNodeLabel(long nodeId) //xmlrpcnotifier.java
     ...
373. if (dbConn != null)
```

[Table of Contents] [Description]

Path 5312:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
     ...
397. if (m_xmlrpcClient == null)
```

[Table of Contents] [Description]

Path 5313:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5314:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      405. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5315:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      414. if (success)
```

[Table of Contents] [Description]

Path 5316:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      418. if (!success)
```

[Table of Contents] [Description]

Path 5317:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      418. if (!success)
```

[Table of Contents] [Description]

Path 5318:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5319:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
453. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5320:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5321:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
467. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5322:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
477. if (success)
```

[Table of Contents] [Description]

Path 5323:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
       ...
488. if (success)
```

[Table of Contents] [Description]

Path 5324:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
201. private void decodeFlags(int flags) throws IOException //dnsaddressrequest.java
       ...
206. if (!isResponse)
```

[Table of Contents] [Description]

Path 5325:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
201. private void decodeFlags(int flags) throws IOException //dnsaddressrequest.java
       ...
215. if (code == 2) // Throw exception if error code indicates 'Server
```

[Table of Contents] [Description]

Path 5326:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
240. public DNSAddressRequest(String host) //dnsaddressrequest.java
       ...
251. if (labels.nextToken().length() > 63)
```

[Table of Contents] [Description]

Path 5327:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
       ...
325. if (id != m_reqID)
```

[Table of Contents] [Description]

Path 5328:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
319. public void receiveResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |
      ...
      359. if (!m_truncated)
```

[Table of Contents] [Description]

Path 5329:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
384. public void verifyResponse(byte[] data, int length) throws
IOException //dnsaddressrequest.java
      |
      ...
      390. if (id != m_reqID)
```

[Table of Contents] [Description]

Path 5330:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
89. public int readByte() throws IOException //dnsinputstream.java
      |
      ...
      91. if (rc == -1)
```

[Table of Contents] [Description]

Path 5331:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
138. public String readString() throws IOException //dnsinputstream.java
      |
      ...
      146. if (rc == -1 || rc != len)
```

[Table of Contents] [Description]

Path 5332:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public String readDomainName() throws IOException //dnsinputstream.java
      |
      ...
      172. if (pos >= count)
```

[Table of Contents] [Description]

Path 5333:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public String readDomainName() throws IOException //dnsinputstream.java
      ...
191. if (tail.length() > 0)
```

[Table of Contents] [Description]

Path 5334:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
166. public String readDomainName() throws IOException //dnsinputstream.java
      ...
212. if ((buf[pos] & 0xc0) != 0xc0)
```

[Table of Contents] [Description]

Path 5335:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
113. public final int storeToBuffer(byte[] buf, int offset) //addressmaskreply.java
114. if (buf.length < (offset + 12))
```

[Table of Contents] [Description]

Path 5336:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public final int loadFromBuffer(byte[] buf, int offset) //addressmaskreply.java
151. if (buf.length < (offset + 12))
```

[Table of Contents] [Description]

Path 5337:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
150. public final int loadFromBuffer(byte[] buf, int offset) //addressmaskreply.java
      ...
156. if (getType() != TYPE_ADDRESS_MASK_REPLY)
```

[Table of Contents] [Description]

Path 5338:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
109. public final int storeToBuffer(byte[] buf, int
offset) //addressmaskrequest.java
     |110. if (buf.length < (offset + 12))
```

[Table of Contents] [Description]

Path 5339:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public final int loadFromBuffer(byte[] buf, int
offset) //addressmaskrequest.java
     |145. if (buf.length < (offset + 12))
```

[Table of Contents] [Description]

Path 5340:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
144. public final int loadFromBuffer(byte[] buf, int
offset) //addressmaskrequest.java
     |...
     |150. if (getType() != TYPE_ADDRESS_MASK_REQUEST)
```

[Table of Contents] [Description]

Path 5341:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. public final int loadFromBuffer(byte[] buf, int offset) //errormessage.java
     |...
     |112. if (buf.length < (offset + 36))
```

[Table of Contents] [Description]

Path 5342:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
108. public final int loadFromBuffer(byte[] buf, int offset) //errormessage.java
     |...
     |125. if (buf.length < (offset + 16 + m_iphdr.getHeaderLength()))
```

[Table of Contents] [Description]

Path 5343:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
184. protected void setProtocolData(byte[] pd) //errormessage.java
     185. if (pd.length < 8)
```

[Table of Contents] [Description]

Path 5344:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
184. protected void setProtocolData(byte[] pd) //errormessage.java
     ...
     188. if (pd.length > 8)
```

[Table of Contents] [Description]

Path 5345:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
180. protected static short byteToShort(byte b) //icmpheader.java
     ...
     182. if (s < 0)
```

[Table of Contents] [Description]

Path 5346:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
196. protected static int byteToInt(byte b) //icmpheader.java
     ...
     198. if (i < 0)
```

[Table of Contents] [Description]

Path 5347:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
302. public ICMPHeader(byte[] data, int offset) //icmpheader.java
     303. if ((data.length - offset) < getNetworkSize())
```

[Table of Contents] [Description]

Path 5348:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
472. protected int storeToBuffer(byte[] buf, int offset) //icmpheader.java
     473. if (buf.length < (offset + 8))
```

[Table of Contents] [Description]

Path 5349:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
505. protected int loadFromBuffer(byte[] buf, int offset) //icmpheader.java
     506. if (buf.length < (offset + 8))
```

[Table of Contents] [Description]

Path 5350:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
121. public final int storeToBuffer(byte[] buf, int offset) //timestampreply.java
     122. if (buf.length < (offset + 20))
```

[Table of Contents] [Description]

Path 5351:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public final int loadFromBuffer(byte[] buf, int offset) //timestampreply.java
     169. if (buf.length < (offset + 20))
```

[Table of Contents] [Description]

Path 5352:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
168. public final int loadFromBuffer(byte[] buf, int offset) //timestampreply.java
     ...
     174. if (getType() != TYPE_TIMESTAMP_REPLY)
```

[Table of Contents] [Description]

Path 5353:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
115. public final int storeToBuffer(byte[] buf, int offset) //timestamprequest.java
     116. if (buf.length < (offset + 20))
```

[Table of Contents] [Description]

Path 5354:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
158. public final int loadFromBuffer(byte[] buf, int offset) //timestamprequest.java
     159. if (buf.length < (offset + 20))
```

[Table of Contents] [Description]

Path 5355:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
158. public final int loadFromBuffer(byte[] buf, int offset) //timestamprequest.java
     ...
     164. if (getType() != TYPE_TIMESTAMP_REQUEST)
```

[Table of Contents] [Description]

Path 5356:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
439. void add(IPv4Address addr) //ipheader.java
     440. if (m_addrs.size() == 9)
```

[Table of Contents] [Description]

Path 5357:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
503. RouteOption(byte code, IPv4Address[] addrs) //ipheader.java
     ...
     505. if (addrs.length > 9)
```

[Table of Contents] [Description]

Path 5358:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
525. RouteOption(byte code, List addrs) //ipheader.java
     ...
     527. if (addrs.size() > 9)
```

[Table of Contents] [Description]

Path 5359:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
746. private static short byteToShort(byte b) //ipheader.java
       ...
748. if (r < 0)
```

[Table of Contents] [Description]

Path 5360:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
762. private static int byteToInt(byte b) //ipheader.java
       ...
764. if (r < 0)
```

[Table of Contents] [Description]

Path 5361:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
778. private static int shortToInt(short s) //ipheader.java
       ...
780. if (r < 0)
```

[Table of Contents] [Description]

Path 5362:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. public IPHeader(byte[] header, int offset) //ipheader.java
       ...
850. if ((length - offset) < 20)
```

[Table of Contents] [Description]

Path 5363:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. public IPHeader(byte[] header, int offset) //ipheader.java
       ...
872. if (m_version != 4)
```

[Table of Contents] [Description]

Path 5364:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
847. public IPHeader(byte[] header, int offset) //ipheader.java
       ...
879. if ((length - offset) < (m_hdrlen * 4))
```

[Table of Contents] [Description]

Path 5365:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1013. public boolean getTypeOfService(int bit) //ipheader.java
1014. if (bit >= 0 && bit < 8)
```

[Table of Contents] [Description]

Path 5366:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1094. public boolean getFlag(int bit) //ipheader.java
1095. if (bit >= 0 && bit < 3)
```

[Table of Contents] [Description]

Path 5367:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1242. public List getOptions() throws InstantiationException //ipheader.java
       ...
1246. if (m_options == null)
```

[Table of Contents] [Description]

Path 5368:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1324. public void addOption(Option opt) //ipheader.java
       ...
1328. if ((len % 4) != 0)
```

[Table of Contents] [Description]

Path 5369:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1324. public void addOption(Option opt) //ipheader.java
      |...
      |1337. if (origLen + opt.bytesRequired() > 40)
```

[Table of Contents] [Description]

Path 5370:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1324. public void addOption(Option opt) //ipheader.java
      |...
      |1341. if ((len % 4) != 0)
```

[Table of Contents] [Description]

Path 5371:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1431. public static String addressToString(byte[] buf) //ipheader.java
      |1432. if (buf.length != 4)
```

[Table of Contents] [Description]

Path 5372:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
91. private static int byteToInt(byte b) //ipv4address.java
      |...
      |93. if (r < 0)
```

[Table of Contents] [Description]

Path 5373:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
135. public static String addressToString(byte[] buf) //ipv4address.java
      |136. if (buf.length != 4)
```

[Table of Contents] [Description]

Path 5374:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
186. public IPv4Address(byte[] addr) //ipv4address.java
     187. if (addr.length != 4)
```

[Table of Contents] [Description]

Path 5375:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public IPv4Address(String ipv4Addr) //ipv4address.java
     ...
     225. if (tok.countTokens() != 4)
```

[Table of Contents] [Description]

Path 5376:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
223. public IPv4Address(String ipv4Addr) //ipv4address.java
     ...
     232. if (converted < 0 || converted > 255)
```

[Table of Contents] [Description]

Path 5377:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
298. public void setAddress(byte[] addr) //ipv4address.java
     299. if (addr.length != 4)
```

[Table of Contents] [Description]

Path 5378:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void setAddress(String ipv4Addr) //ipv4address.java
     ...
     337. if (tok.countTokens() != 4)
```

[Table of Contents] [Description]

Path 5379:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
335. public void setAddress(String ipv4Addr) //ipv4address.java
```

```
...
344. if (converted < 0 || converted > 255)
```

[Table of Contents] [Description]

Path 5380:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
57. private static int byteToInt(byte b) //oc16checksumproducer.java
    58. if (b < 0)
```

[Table of Contents] [Description]

Path 5381:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. private static int shortToInt(short s) //oc16checksumproducer.java
    73. if (s < 0)
```

[Table of Contents] [Description]

Path 5382:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public short getChecksum() //oc16checksumproducer.java
     196. if (m_finalized == false)
```

[Table of Contents] [Description]

Path 5383:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public short getChecksum() //oc16checksumproducer.java
     196. if (m_finalized == false)
```

[Table of Contents] [Description]

Path 5384:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
297. public static short unsignedByteToShort(byte b) //ntpmessage.java
     298. if ((b & 0x80) == 0x80)
```

[Table of Contents] [Description]

Path 5385:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
345. public static String timestampToString(double timestamp) //ntpmessage.java
     |346. if (timestamp == 0)
```

[Table of Contents] [Description]

Path 5386:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void init(FilterConfig config) //addrefreshheaderfilter.java
     |...
     |82. if (seconds != null)
```

[Table of Contents] [Description]

Path 5387:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
     |...
     |134. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5388:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
     |...
     |134. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5389:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
     |...
     |166. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5390:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
166. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5391:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
171. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5392:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
171. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5393:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5394:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
174. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5395:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
178. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5396:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
178. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5397:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
192. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5398:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
192. if (log.isInfoEnabled())
```

[Table of Contents] [Description]

Path 5399:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5400:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
164. public void run() //reportmailer.java
       ...
195. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5401:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. protected void generateFile(String shellScript) throws IOException,
InterruptedException //reportmailer.java
       ...
221. if (!commandParms.equals(""))
```

[Table of Contents] [Description]

Path 5402:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. protected void generateFile(String shellScript) throws IOException,
InterruptedException //reportmailer.java
       ...
226. if (!format.equals(""))
```

[Table of Contents] [Description]

Path 5403:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. protected void mailFileToUser(String mailScript, String filename, String
emailAddr) throws IOException //reportmailer.java
       ...
281. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5404:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
263. protected void mailFileToUser(String mailScript, String filename, String
emailAddr) throws IOException //reportmailer.java
```

```
...
281. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5405:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
317. public void setUseScript(boolean useScript) //reportmailer.java
     318. if (useScript)
```

[Table of Contents] [Description]

Path 5406:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
317. public void setUseScript(boolean useScript) //reportmailer.java
     318. if (useScript)
```

[Table of Contents] [Description]

Path 5407:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public static String getHumanReadableIfSpeed(long ifSpeed) //util.java
     196. if (ifSpeed == 10000000L)
```

[Table of Contents] [Description]

Path 5408:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public static String getHumanReadableIfSpeed(long ifSpeed) //util.java
     ...
     198. else if (ifSpeed == 100000000L)
```

[Table of Contents] [Description]

Path 5409:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public static String getHumanReadableIfSpeed(long ifSpeed) //util.java
     ...
     200. else if (ifSpeed == 1000000000L)
```

[Table of Contents] [Description]

Path 5410:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
195. public static String getHumanReadableIfSpeed(long ifSpeed) //util.java
     ...
202. else if (ifSpeed == 10000000000L)
```

[Table of Contents] [Description]

Path 5411:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
222. if (errorflag)
```

[Table of Contents] [Description]

Path 5412:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
251. if (!errorflag)
```

[Table of Contents] [Description]

Path 5413:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
251. if (!errorflag)
```

[Table of Contents] [Description]

Path 5414:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
```

```
      ...
301.  if (props.get("banner") != null)
```

[Table of Contents] [Description]

Path 5415:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255.  public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
310.  if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5416:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255.  public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
310.  if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5417:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255.  public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
310.  if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5418:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255.  public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
328.  if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5419:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
328. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5420:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
328. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5421:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
338. if (props.get("timeout") != null)
```

[Table of Contents] [Description]

Path 5422:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
346. if (props.get("retry") != null)
```

[Table of Contents] [Description]

Path 5423:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
382. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5424:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
382. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5425:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
382. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5426:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
395. if (monitorColl != null)
```

[Table of Contents] [Description]

Path 5427:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
```

```
395. if (monitorColl != null)
```

[Table of Contents] [Description]

Path 5428:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
403. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5429:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
403. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5430:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
403. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5431:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
419. if (props.get("banner") != null)
```

[Table of Contents] [Description]

Path 5432:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
427. if (props.get("retry") != null)
```

[Table of Contents] [Description]

Path 5433:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
445. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5434:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
445. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5435:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
445. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5436:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
204. public boolean isOnDuty(Calendar aTime) //groupinfo.java
      ...
220. if (result)
```

[Table of Contents] [Description]

Path 5437:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
125. if (deleteDir(nodeDir))
```

[Table of Contents] [Description]

Path 5438:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
125. if (deleteDir(nodeDir))
```

[Table of Contents] [Description]

Path 5439:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
147. if (deleteDir(intfDir))
```

[Table of Contents] [Description]

Path 5440:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
147. if (deleteDir(intfDir))
```

[Table of Contents] [Description]

Path 5441:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getinterfacesservlet.java
    ...
80. if (nodeIdString == null)
```

[Table of Contents] [Description]

Path 5442:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getinterfacesservlet.java
    ...
91. if (nodeId < 0)
```

[Table of Contents] [Description]

Path 5443:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
    ...
214. if (manageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5444:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
    ...
214. if (manageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5445:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
    ...
216. if (unmanageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5446:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
      ...
216. if (unmanageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5447:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
      ...
248. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5448:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
      ...
248. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5449:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
      ...
268. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5450:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodeservlet.java
```

```
...
268. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5451:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
...
214. if (manageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5452:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
...
214. if (manageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5453:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
...
216. if (unmanageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5454:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
...
216. if (unmanageInterfacesList.size() > 0)
```

[Table of Contents] [Description]

Path 5455:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
     ...
248. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5456:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
241. private void manageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
     ...
248. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5457:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
     ...
268. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5458:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
261. private void unmanageInterfaces(List interfaces, Connection connection) throws
SQLException //managenodesservlet.java
     ...
268. if (i < interfaces.size() - 1)
```

[Table of Contents] [Description]

Path 5459:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
     ...
80. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5460:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
    ...
80. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5461:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
    ...
109. if (testInterface.getNodeid() == currNodeId && "P".equals
(testInterface.getStatus()))
```

[Table of Contents] [Description]

Path 5462:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
    ...
150. if (primeInt != null)
```

[Table of Contents] [Description]

Path 5463:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
    ...
150. if (primeInt != null)
```

[Table of Contents] [Description]

Path 5464:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
```

```
...
198. if (index == -1)
```

[Table of Contents] [Description]

Path 5465:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
...
198. if (index == -1)
```

[Table of Contents] [Description]

Path 5466:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
...
198. if (index == -1)
```

[Table of Contents] [Description]

Path 5467:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
...
218. if (index == -1)
```

[Table of Contents] [Description]

Path 5468:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
...
218. if (index == -1)
```

[Table of Contents] [Description]

Path 5469:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
218. if (index == -1)
```

[Table of Contents] [Description]

Path 5470:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
239. if (index == -1)
```

[Table of Contents] [Description]

Path 5471:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
239. if (index == -1)
```

[Table of Contents] [Description]

Path 5472:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
239. if (index == -1)
```

[Table of Contents] [Description]

Path 5473:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      ...
230. if (subject != null && !subject.trim().equals(""))
```

[Table of Contents] [Description]

Path 5474:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      |...
      |230. if (subject != null && !subject.trim().equals(""))
```

[Table of Contents] [Description]

Path 5475:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      |...
      |163. if (i < notServices.length - 1)
```

[Table of Contents] [Description]

Path 5476:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      |...
      |163. if (i < notServices.length - 1)
```

[Table of Contents] [Description]

Path 5477:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      |...
      |206. if (services != null)
```

[Table of Contents] [Description]

Path 5478:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      |
```

```
...
206. if (services != null)
```

[Table of Contents] [Description]

Path 5479:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
...
222. if (description != null && !description.trim().equals(""))
```

[Table of Contents] [Description]

Path 5480:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
...
222. if (description != null && !description.trim().equals(""))
```

[Table of Contents] [Description]

Path 5481:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
...
222. if (description != null && !description.trim().equals(""))
```

[Table of Contents] [Description]

Path 5482:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
...
230. if (subject != null && !subject.trim().equals(""))
```

[Table of Contents] [Description]

Path 5483:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
230. if (subject != null && !subject.trim().equals(""))
```

[Table of Contents] [Description]

Path 5484:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
230. if (subject != null && !subject.trim().equals(""))
```

[Table of Contents] [Description]

Path 5485:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
236. if (numMessage != null && !numMessage.trim().equals(""))
```

[Table of Contents] [Description]

Path 5486:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
236. if (numMessage != null && !numMessage.trim().equals(""))
```

[Table of Contents] [Description]

Path 5487:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
236. if (numMessage != null && !numMessage.trim().equals(""))
```

[Table of Contents] [Description]

Path 5488:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
254. if (redirectString.toString().equals(""))
```

[Table of Contents] [Description]

Path 5489:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
314. private static StringBuffer toSingleQuote(StringBuffer
buffer) //notificationwizardservlet.java
    ...
316. if ((i < buffer.length() - 5) && (buffer.substring(i, i + 6).equals
("&quot;"))) {
```

[Table of Contents] [Description]

Path 5490:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
226. if (errorflag)
```

[Table of Contents] [Description]

Path 5491:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
252. if (!errorflag)
```

[Table of Contents] [Description]

Path 5492:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
```

```
...
252. if (!errorflag)
```

[Table of Contents] [Description]

Path 5493:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
299. if (props.get("banner") != null)
```

[Table of Contents] [Description]

Path 5494:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
308. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5495:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
308. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5496:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
308. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5497:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
     ...
322. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5498:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
     ...
322. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5499:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
     ...
322. if (port.indexOf(":") == -1)
```

[Table of Contents] [Description]

Path 5500:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
     ...
331. if (props.get("timeout") != null)
```

[Table of Contents] [Description]

Path 5501:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
      ...
335. if (pluginAdd != null)
```

[Table of Contents] [Description]

Path 5502:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
      ...
335. if (pluginAdd != null)
```

[Table of Contents] [Description]

Path 5503:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
      ...
340. if (props.get("retry") != null)
```

[Table of Contents] [Description]

Path 5504:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
377. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5505:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
```

```
377. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5506:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
377. if (bPolled != null)
```

[Table of Contents] [Description]

Path 5507:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
396. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5508:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
396. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5509:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
396. if (props.get("interval") != null)
```

[Table of Contents] [Description]

Path 5510:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
412. if (props.get("banner") != null)
```

[Table of Contents] [Description]

Path 5511:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
420. if (props.get("retry") != null)
```

[Table of Contents] [Description]

Path 5512:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
440. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5513:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
440. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5514:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
440. if (port.indexOf(":") != -1)
```

[Table of Contents] [Description]

Path 5515:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
445. if (monitorColl == null)
```

[Table of Contents] [Description]

Path 5516:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
445. if (monitorColl == null)
```

[Table of Contents] [Description]

Path 5517:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
311. public boolean isOnDuty(Calendar aTime) //notificationinfo.java
      ...
327. if (result)
```

[Table of Contents] [Description]

Path 5518:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
262. private static String hexToString(byte data[]) //user.java
      ...
266. if ((data.length % 2) != 0)
```

[Table of Contents] [Description]

Path 5519:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
245. private static String hexToString(byte data[]) //userinfo.java
       ...
249. if ((data.length % 2) != 0)
```

[Table of Contents] [Description]

Path 5520:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateviewservlet.java
       ...
83. if (membership.getMemberCount() > 0)
```

[Table of Contents] [Description]

Path 5521:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateviewservlet.java
       ...
83. if (membership.getMemberCount() > 0)
```

[Table of Contents] [Description]

Path 5522:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private ArrayList splitfields(String string, String sep, int
maxsplit) //importassetsservlet.java
       ...
217. if (maxsplit < 0)
```

[Table of Contents] [Description]

Path 5523:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. private ArrayList splitfields(String string, String sep, int
maxsplit) //importassetsservlet.java
       ...
225. if (index == -1)
```

[Table of Contents] [Description]

Path 5524:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      ...
97. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5525:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      ...
97. if (log.isDebugEnabled())
```

[Table of Contents] [Description]

Path 5526:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
618. if (element != null)
```

[Table of Contents] [Description]

Path 5527:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
655. protected static Interface[] rs2Interfaces(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
676. if (element != null)
```

[Table of Contents] [Description]

Path 5528:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
```

```
SQLException //networkelementfactory.java
    ...
751. if (element != null)
```

[Table of Contents] [Description]

Path 5529:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
758. if (element != null)
```

[Table of Contents] [Description]

Path 5530:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
189. public boolean eventHasNotice(String uei, String
notice) //eventconffactory.java
    ...
202. if (index != -1)
```

[Table of Contents] [Description]

Path 5531:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
211. public Event getEvent(String uei) //eventconffactory.java
    ...
223. if (index == null)
```

[Table of Contents] [Description]

Path 5532:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
268. public synchronized void removeEvent(String uei) throws
XMLWriteException //eventconffactory.java
    ...
272. if (index == null)
```

[Table of Contents] [Description]

Path 5533:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public void init() throws ServletException //rrdgraphservlet.java
       |...
       |142. if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 5534:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
117. public void init() throws ServletException //rrdgraphservlet.java
       |...
       |142. if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 5535:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
       |...
       |258. if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 5536:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
       |...
       |258. if (fileInputStream != null) fileInputStream.close();
```

[Table of Contents] [Description]

Path 5537:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
       |...
       |157. if ("FATAL".equalsIgnoreCase(verbosity))
```

[Table of Contents] [Description]

Path 5538:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
     ...
159. else if ("ERROR".equalsIgnoreCase(verbosity))
```

[Table of Contents] [Description]

Path 5539:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
     ...
161. else if ("WARNING".equalsIgnoreCase(verbosity))
```

[Table of Contents] [Description]

Path 5540:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
     ...
163. else if ("INFORMATION".equalsIgnoreCase(verbosity))
```

[Table of Contents] [Description]

Path 5541:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
155. public void setVerbosityLevel(String verbosity) //log4jlogger.java
     ...
165. else if ("DEBUG".equalsIgnoreCase(verbosity))
```

[Table of Contents] [Description]

Path 5542:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public void log(String message, int verbosity) //log4jlogger.java
     ...
256. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 5543:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
254. public void log(String message, int verbosity) //log4jlogger.java
      |...
      256. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 5544:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public void log(String message, Throwable throwable, int
verbosity) //log4jlogger.java
      |...
      275. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 5545:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
273. public void log(String message, Throwable throwable, int
verbosity) //log4jlogger.java
      |...
      275. if (this.verbosity >= verbosity)
```

[Table of Contents] [Description]

Path 5546:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
123. public Date getTimeReplied() //notification.java
      |...
      125. if (this.m_timeReply == 0)
```

[Table of Contents] [Description]

Path 5547:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |...
      123. if (ts != null)
```

[Table of Contents] [Description]

Path 5548:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |
      ...
      123. if (ts != null)
```

[Table of Contents] [Description]

Path 5549:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |
      ...
      123. if (ts != null)
```

[Table of Contents] [Description]

Path 5550:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      |
      ...
      154. if (element != null)
```

[Table of Contents] [Description]

Path 5551:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      |
      ...
      253. if (element != null)
```

[Table of Contents] [Description]

Path 5552:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private String appendZero(String values) //notificationmodel.java
      |
```

```
...
509. } else if (len > 2)
```

[Table of Contents] [Description]

Path 5553:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
501. private String appendZero(String values) //notificationmodel.java
...
509. } else if (len > 2)
```

[Table of Contents] [Description]

Path 5554:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. protected String processParmValue(Node parmValueNode) //bbparser.java
...
239. if (numChildren < 1)
```

[Table of Contents] [Description]

Path 5555:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. protected String processParmValue(Node parmValueNode) //bbparser.java
...
244. if (temp.getNodeType() == Node.TEXT_NODE)
```

[Table of Contents] [Description]

Path 5556:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. protected String processParmValue(Node parmValueNode) //bbparser.java
...
246. else if (temp.getNodeType() == Node.CDATA_SECTION_NODE)
```

[Table of Contents] [Description]

Path 5557:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
235. protected String processParmValue(Node parmValueNode) //bbparser.java
```

```
...
249. if (value != null)
```

[Table of Contents] [Description]

Path 5558:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. public String getErrorMessage() //bbparser.java
     361. if (m_errNum == ATTRIB_VALUE_PAIR_ERR)
```

[Table of Contents] [Description]

Path 5559:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. public String getErrorMessage() //bbparser.java
     ...
     364. else if (m_errNum == NULL_VALUE_ERR)
```

[Table of Contents] [Description]

Path 5560:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
360. public String getErrorMessage() //bbparser.java
     ...
     367. else if (m_errNum == EXCEPTION)
```

[Table of Contents] [Description]

Path 5561:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
148. public String getDTD(String dtdFileName) throws FileNotFoundException,
IOException //xmlwriter.java
     ...
     156. if (m_dtd != null)
```

[Table of Contents] [Description]

Path 5562:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. public QueryableNode[] getQueryableNodes() throws
```

```
SQLException //performancemodel.java
      |...
      |407. if (rs != null)
```

[Table of Contents] [Description]

Path 5563:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      |...
      |407. if (rs != null)
```

[Table of Contents] [Description]

Path 5564:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      |...
      |409. if (stmt != null)
```

[Table of Contents] [Description]

Path 5565:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      |...
      |409. if (stmt != null)
```

[Table of Contents] [Description]

Path 5566:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
      |...
      |412. if (rs != null)
```

[Table of Contents] [Description]

Path 5567:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
     ...
412. if (rs != null)
```

[Table of Contents] [Description]

Path 5568:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
     ...
414. if (stmt != null)
```

[Table of Contents] [Description]

Path 5569:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
     ...
414. if (stmt != null)
```

[Table of Contents] [Description]

Path 5570:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //deletenode.jsp
     ...
57. if (nodeId < 0 )
```

[Table of Contents] [Description]

Path 5571:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //index.jsp
     ...
58. if (nodeId < -1)
```

[Table of Contents] [Description]

Path 5572:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //managenode.jsp
      ...
185. if (midInterfaceIndex < 1)
```

[Table of Contents] [Description]

Path 5573:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
143. public String displayTargets(String users[], String groups[]) //escalations.jsp
      ...
145. if (users==null)
```

[Table of Contents] [Description]

Path 5574:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //escalations.jsp
      ...
66. if (path == null)
```

[Table of Contents] [Description]

Path 5575:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
179. public String getGroupInterval(Path path, String group, int
index) //groupintervals.jsp
      ...
194. if (group.equals(targets[i].getName()))
```

[Table of Contents] [Description]

Path 5576:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //validaterule.jsp
      ...
59. if (services==null)
```

[Table of Contents] [Description]

Path 5577:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //validaterule.jsp
       ...
62. if (notServices==null)
```

[Table of Contents] [Description]

Path 5578:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
       ...
172. if ("weekly".equals(pollFactory.getOutageType(outageName)))
```

[Table of Contents] [Description]

Path 5579:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
       ...
174. if ("specific".equals(pollFactory.getOutageType(outageName)))
```

[Table of Contents] [Description]

Path 5580:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //viewdetail.jsp
       ...
138. if(member.getType().equals("user"))
```

[Table of Contents] [Description]

Path 5581:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //viewdetail.jsp
       ...
138. if(member.getType().equals("user"))
```

[Table of Contents] [Description]

Path 5582:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //viewdetail.jsp
       ...
163. if(member.getType().equals("group"))
```

[Table of Contents] [Description]

Path 5583:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //viewdetail.jsp
       ...
163. if(member.getType().equals("group"))
```

[Table of Contents] [Description]

Path 5584:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //modify.jsp
       ...
172. if(os == null || os.equals(""))
```

[Table of Contents] [Description]

Path 5585:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //modify.jsp
       ...
174. if (os == null)
```

[Table of Contents] [Description]

Path 5586:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //index.jsp
       ...
130. if(checkCategory)
```

[Table of Contents] [Description]

Path 5587:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //index.jsp
     ...
146. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 5588:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
     ...
621. if (intf_db.getSnmpIpAdEntNetMask() != null)
```

[Table of Contents] [Description]

Path 5589:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
     ...
624. if (intf_db.getSnmpIfType() > 0)
```

[Table of Contents] [Description]

Path 5590:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
     ...
627. if (intf_db.getSnmpIfSpeed() > 0)
```

[Table of Contents] [Description]

Path 5591:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
     ...
630. if (intf_db.getSnmpIfDescription() != null)
```

[Table of Contents] [Description]

Path 5592:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
       ...
633. if (intf_db.getSnmpIfAdminStatus() > 0)
```

[Table of Contents] [Description]

Path 5593:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
       ...
636. if (intf_db.getSnmpIfOperStatus() > 0)
```

[Table of Contents] [Description]

Path 5594:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
619. private boolean hasSNMPData(Interface intf_db) //interface.jsp
       ...
639. if (intf_db.getSnmpIfAlias() != null)
```

[Table of Contents] [Description]

Path 5595:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //interface.jsp
       ...
109. if (ifindexString != null)
```

[Table of Contents] [Description]

Path 5596:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //node.jsp
       ...
190. if( snmpServices != null && snmpServices.length > 0 )
```

[Table of Contents] [Description]

Path 5597:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //custom_graph3.jsp
      ...
      246. if (graph_index == -1)
```

[Table of Contents] [Description]

Path 5598:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //list.jsp
      ...
      142. <tr <% if( i%2 == 0 ) out.print( "BGCOLOR=\"#cccccc\""); %> valign="top">
```

[Table of Contents] [Description]

Path 5599:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //current.jsp
      ...
      166. <tr valign="top" <% if( nodeIndex%2 == 0 ) out.print
      ( "BGCOLOR=\"#cccccc\""); %>>
```

[Table of Contents] [Description]

Path 5600:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
      ...
      108. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
      out.print("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 5601:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
      ...
      120. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
```

```
" ); %>>
```

[Table of Contents] [Description]

Path 5602:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
...
135. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 5603:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
...
147. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
>>
```

[Table of Contents] [Description]

Path 5604:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
...
174. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
out.print("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 5605:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
...
186. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
" ); %>>
```

[Table of Contents] [Description]

Path 5606:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      |
      ...
      201. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
      ("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 5607:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      |
      ...
      213. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
      >>
```

[Table of Contents] [Description]

Path 5608:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
      |
      ...
      110. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
      out.print("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 5609:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
      |
      ...
      122. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
      " ); %>>
```

[Table of Contents] [Description]

Path 5610:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
      |
      ...
      137. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
      ("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 5611:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       ...
149. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
     >>
```

[Table of Contents] [Description]

Path 5612:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
       ...
174. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
     out.print("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 5613:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
       ...
186. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
     " ); %>>
```

[Table of Contents] [Description]

Path 5614:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
       ...
201. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
     ("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 5615:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
       ...
213. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
```

```
>>
```

[Table of Contents] [Description]

Path 5616:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
282. public void run() //sockettest.java
...
314. if (len == 0)
```

[Table of Contents] [Description]

Path 5617:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
397. if(walker.m_port != -1)
```

[Table of Contents] [Description]

Path 5618:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
397. if(walker.m_port != -1)
```

[Table of Contents] [Description]

Path 5619:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
400. if(walker.m_timeout != -1)
```

[Table of Contents] [Description]

Path 5620:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
```

```
...
400. if(walker.m_timeout != -1)
```

[Table of Contents] [Description]

Path 5621:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
403. if(walker.m_retries != -1)
```

[Table of Contents] [Description]

Path 5622:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
403. if(walker.m_retries != -1)
```

[Table of Contents] [Description]

Path 5623:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
408. if(walker.m_community != null)
```

[Table of Contents] [Description]

Path 5624:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
...
408. if(walker.m_community != null)
```

[Table of Contents] [Description]

Path 5625:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
```

```
...
535. if(walker.m_port != -1)
```

[Table of Contents] [Description]

Path 5626:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
535. if(walker.m_port != -1)
```

[Table of Contents] [Description]

Path 5627:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
538. if(walker.m_timeout != -1)
```

[Table of Contents] [Description]

Path 5628:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
538. if(walker.m_timeout != -1)
```

[Table of Contents] [Description]

Path 5629:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
541. if(walker.m_retries != -1)
```

[Table of Contents] [Description]

Path 5630:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
```

```
...
541. if(walker.m_retries != -1)
```

[Table of Contents] [Description]

Path 5631:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
546. if(walker.m_community != null)
```

[Table of Contents] [Description]

Path 5632:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
...
546. if(walker.m_community != null)
```

[Table of Contents] [Description]

Path 5633:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
25. public String getPossibilityAsString(int i) throws
IndexOutOfBoundsException //chinesemenu.java
...
27. if (i > this.numberOfPossibilities())
```

[Table of Contents] [Description]

Path 5634:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
43. public boolean[] getPossibilityAt(int i) throws
IndexOutOfBoundsException //chinesemenu.java
...
45. if (i > this.numberOfPossibilities())
```

[Table of Contents] [Description]

Path 5635:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
43. public boolean[] getPossibilityAt(int i) throws
IndexOutOfBoundsException //chinesemenu.java
      |...
      |57. if (bstr.charAt(j) == '0')
```

[Table of Contents] [Description]

Path 5636:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
43. public boolean[] getPossibilityAt(int i) throws
IndexOutOfBoundsException //chinesemenu.java
      |...
      |59. else if (bstr.charAt(j) == '1')
```

[Table of Contents] [Description]

Path 5637:

Query Name - **Single_Line_If_Statement**

Severity - **Information**

```
82. private static void createState(boolean[] on, SimulatedService[]
service) //tester.java
      |...
      |84. if (!(on.length == service.length))
```

[Table of Contents] [Description]

Improper Error Handling

CWE ID	390
Description	Sometimes an error is detected, and bad or no action is taken.
Likelihood of Exploit	Medium
Potential Mitigations	Implementation: Properly handle each exception. This is the recommended solution. Ensure that all exceptions are handled in such a way that you can be sure of the state of your system at any given moment. Subject the software to extensive testing to discover some of the possible instances of where/how errors or return values are not handled. Consider testing techniques such as ad hoc, equivalence partitioning, robustness and fault tolerance, mutation, and fuzzing.

Demonstrative Examples	**C Example:**
	
	foo=malloc(sizeof(char); //the next line checks to see if malloc failed
	if (foo==0) {
	  //We do nothing so we just ignore the error.

```
}
```

C++ Example:

```
while (DoSomething()) {
  try {
    /* perform main loop here */
  }
  catch (Exception e) {
    /* do nothing, but catch so it'll compile... */
  }
}
```

Java Example:

```
while (DoSomething()) {
  try {
    /* perform main loop here */
  }
  catch (Exception e) {
    /* do nothing, but catch so it'll compile... */
  }
}
```

Context Notes	If a function returns an error, it is important to either fix the problem and try again, alert the user that an error has happened and let the program continue, or alert the user and close and cleanup the program.
Node Relationships	ChildOf - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	CLASP - Improper error handling
Applicable Platforms	All

Path 1:

Query Name - **Empty_Catch**

Severity - **Information**

```
59. public static String localTime (String cvsDate) //cvs.java
      ...
82. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 2:

Query Name - **Empty_Catch**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
      ...
50. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 3:

Query Name - **Empty_Catch**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
      ...
103. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 4:

Query Name - **Empty_Catch**

Severity - **Information**

```
126. private void parseBox(String line) //imageintrinsics.java
      ...
134. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Empty_Catch**

Severity - **Information**

```
224. private static void setupColumnWidths(Context context) //table.java
      ...
243. } catch (TransformerException e) {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Empty_Catch**

Severity - **Information**

```
442. protected static String[] correctRoundingError(String widths[]) //table.java
      ...
449. } catch (NumberFormatException nfe) {
```

[Table of Contents] [Description]

Path 7:

Query Name - **Empty_Catch**

Severity - **Information**

```
442. protected static String[] correctRoundingError(String widths[]) //table.java
      ...
470. } catch (NumberFormatException nfe) {
```

[Table of Contents] [Description]

Path 8:

Query Name - **Empty_Catch**

Severity - **Information**

```
59. public String localTime (String cvsDate) //cvs.java
      |
      ...
      82. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 9:

Query Name - **Empty_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      |
      ...
      52. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 10:

Query Name - **Empty_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      |
      ...
      96. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 11:

Query Name - **Empty_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      |
      ...
      103. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 12:

Query Name - **Empty_Catch**

Severity - **Information**

```
126. private void parseBox(String line) //imageintrinsics.java
      |
```

```
      ...
134.  } catch (Exception e) {
```

[Table of Contents] [Description]

Path 13:

Query Name - **Empty_Catch**

Severity - **Information**

```
496. protected String[] correctRoundingError(String widths[]) //table.java
      ...
503.  } catch (NumberFormatException nfe) {
```

[Table of Contents] [Description]

Path 14:

Query Name - **Empty_Catch**

Severity - **Information**

```
496. protected String[] correctRoundingError(String widths[]) //table.java
      ...
524.  } catch (NumberFormatException nfe) {
```

[Table of Contents] [Description]

Path 15:

Query Name - **Empty_Catch**

Severity - **Information**

```
201. public void run() //vmtaskfiber.java
      ...
215.  } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 16:

Query Name - **Empty_Catch**

Severity - **Information**

```
237. public static boolean supplementSystemPropertiesFromKey(String
key) //vault.java
      ...
256.  } catch (IOException ioE) { /* do nothing */
```

[Table of Contents] [Description]

Path 17:

Query Name - **Empty_Catch**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      ...
135. } catch (IllegalThreadStateException e) {
```

[Table of Contents] [Description]

Path 18:

Query Name - **Empty_Catch**

Severity - **Information**

```
114. public static int getProperty(String name, int
defaultVal) //javamailerconfig.java
      ...
119. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 19:

Query Name - **Empty_Catch**

Severity - **Information**

```
135. public static long getProperty(String name, long
defaultVal) //javamailerconfig.java
      ...
140. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 20:

Query Name - **Empty_Catch**

Severity - **Information**

```
1107. public Map getTypesFromDB() throws SQLException //installer.java
      ...
1122. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 21:

Query Name - **Empty_Catch**

Severity - **Information**

```
83. public void run() //processexec.java
      ...
97. } catch (IOException e2) {
```

[Table of Contents] [Description]

Path 22:

Query Name - **Empty_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
       ...
173. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 23:

Query Name - **Empty_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
       ...
180. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 24:

Query Name - **Empty_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
       ...
187. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 25:

Query Name - **Empty_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
       ...
195. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 26:

Query Name - **Empty_Catch**

Severity - **Information**

```
119. public void run() //snmptimer.java
       ...
211. } catch (NoSuchElementException err) {
```

[Table of Contents] [Description]

Path 27:

Query Name - **Empty_Catch**

Severity - **Information**

```
119. public void run() //snmptimer.java
       ...
213. } catch (Throwable err) {
```

[Table of Contents] [Description]

Path 28:

Query Name - **Empty_Catch**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmptrapsession.java
       ...
198. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 29:

Query Name - **Empty_Catch**

Severity - **Information**

```
220. public void processSnmpTrap(InetAddress agent, int port, SnmpOctetString
community, SnmpPduTrap pdu) throws SnmpPduEncodingException //snmptrapsession.java
       ...
223. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 30:

Query Name - **Empty_Catch**

Severity - **Information**

```
249. public void processException(Exception e) //snmptrapsession.java
       ...
252. } catch (Exception e1) {
```

[Table of Contents] [Description]

Path 31:

Query Name - **Empty_Catch**

Severity - **Information**

```
170. public synchronized void stop() //actiond.java
       ...
177. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 32:

Query Name - **Empty_Catch**

Severity - **Information**

```
195. public void run() //executor.java
       ...
228. } catch (IllegalThreadStateException ex) {
```

[Table of Contents] [Description]

Path 33:

Query Name - **Empty_Catch**

Severity - **Information**

```
382. public void run() //executor.java
       ...
469. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 34:

Query Name - **Empty_Catch**

Severity - **Information**

```
171. protected void closeSocket(Socket socket, ConnectionConfig
config) //abstracttcpplugin.java
       ...
175. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 35:

Query Name - **Empty_Catch**

Severity - **Information**

```
176. public void init() //capsd.java
       ...
299. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 36:

Query Name - **Empty_Catch**

Severity - **Information**

```
103. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout, String lookup) //dnsplugin.java
       ...
148. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 37:

Query Name - **Empty_Catch**

Severity - **Information**

```
109. private boolean isServer( String hostname, Map qualifiers ) //jdbcplugin.java
       |...
       |184. catch (SQLException ignore)
```

[Table of Contents] [Description]

Path 38:

Query Name - **Empty_Catch**

Severity - **Information**

```
193. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcplugin.java
       |...
       |225. catch (SQLException ignore)
```

[Table of Contents] [Description]

Path 39:

Query Name - **Empty_Catch**

Severity - **Information**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
       |...
       |113. catch( SQLException ignore)
```

[Table of Contents] [Description]

Path 40:

Query Name - **Empty_Catch**

Severity - **Information**

```
294. static synchronized void dataSourceSync() throws SQLException //knownipmgr.java
       |...
       |343. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 41:

Query Name - **Empty_Catch**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
       |...
```

```
175. } catch (LDAPException e) {
```

[Table of Contents] [Description]

Path 42:

Query Name - **Empty_Catch**

Severity - **Information**

```
137. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //ldapplugin.java
     ...
198. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 43:

Query Name - **Empty_Catch**

Severity - **Information**

```
129. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //msexchangeplugin.java
     ...
177. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 44:

Query Name - **Empty_Catch**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, String command, int
padding, int retries, int timeout, RE regex, StringBuffer
bannerResult) //nrpeplugin.java
     ...
206. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 45:

Query Name - **Empty_Catch**

Severity - **Information**

```
97. private boolean isServer(InetAddress nserver, int port, int retries, int
timeout) //ntpplugin.java
     ...
145. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 46:

Query Name - **Empty_Catch**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
      ...
182. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 47:

Query Name - **Empty_Catch**

Severity - **Information**

```
770. private void updateInterface(Connection dbc, Date now, DbNodeEntry node,
InetAddress target, InetAddress ifaddr, List protocols, IfSnmpCollector snmpc,
boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
907. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 48:

Query Name - **Empty_Catch**

Severity - **Information**

```
963. private void deleteDuplicateNode(Connection dbc, DbNodeEntry duplicateNode)
throws SQLException //rescanprocessor.java
      ...
990. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 49:

Query Name - **Empty_Catch**

Severity - **Information**

```
1009. private boolean isDuplicateInterface(Connection dbc, InetAddress ifaddr, int
nodeId) throws SQLException //rescanprocessor.java
      ...
1031. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 50:

Query Name - **Empty_Catch**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1229. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 51:

Query Name - **Empty_Catch**

Severity - **Information**

```
1255. private void updateServiceInfo(Connection dbc, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, boolean isNewIpEntry, List protocols) throws
SQLException //rescanprocessor.java
      ...
1335. } catch (ClassCastException ccE) {
```

[Table of Contents] [Description]

Path 52:

Query Name - **Empty_Catch**

Severity - **Information**

```
1899. private void addSupportedProtocols(DbNodeEntry node, DbIpInterfaceEntry
ipIfEntry, List protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //rescanprocessor.java
      ...
1944. } catch (ClassCastException ccE) {
```

[Table of Contents] [Description]

Path 53:

Query Name - **Empty_Catch**

Severity - **Information**

```
1975. private void reparentInterface(Connection dbc, InetAddress ifAddr, int
ifIndex, int newNodeId, int oldNodeId) throws SQLException //rescanprocessor.java
      ...
2133. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 54:

Query Name - **Empty_Catch**

Severity - **Information**

```
2530. private boolean isNodeDeleted(Connection dbc, int nodeId) throws
SQLException //rescanprocessor.java
      ...
2552. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 55:

Query Name - **Empty_Catch**

Severity - **Information**

```
3009. private InetAddress updatePrimarySnmpInterface(Connection dbc, DbNodeEntry
dbNodeEntry, Map collectorMap, InetAddress oldPriIf) throws
SQLException //rescanprocessor.java
      ...
3049. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 56:

Query Name - **Empty_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
288. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 57:

Query Name - **Empty_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
294. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 58:

Query Name - **Empty_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
300. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 59:

Query Name - **Empty_Catch**

Severity - **Information**

```
316. void scheduleNode(int nodeId) throws SQLException //scheduler.java
      ...
340. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 60:

Query Name - **Empty_Catch**

Severity - **Information**

```
492. public void run() //scheduler.java
      ...
587. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 61:

Query Name - **Empty_Catch**

Severity - **Information**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      ...
288. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 62:

Query Name - **Empty_Catch**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      ...
162. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 63:

Query Name - **Empty_Catch**

Severity - **Information**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      ...
200. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 64:

Query Name - **Empty_Catch**

Severity - **Information**

```
885. private void addSupportedProtocols(DbNodeEntry node, InetAddress ifaddr, List
protocols, boolean addrUnmanaged, int ifIndex,
org.opennms.netmgt.config.poller.Package ipPkg) throws
SQLException //suspecteventprocessor.java
      ...
931. } catch (ClassCastException ccE) {
```

[Table of Contents] [Description]

Path 65:

Query Name - **Empty_Catch**

Severity - **Information**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
      |...
      |188. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 66:

Query Name - **Empty_Catch**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      |514. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 67:

Query Name - **Empty_Catch**

Severity - **Information**

```
165. protected void sendEvent(String uei) //notifdconfigmanager.java
      |...
      |174. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 68:

Query Name - **Empty_Catch**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
      |...
      |238. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 69:

Query Name - **Empty_Catch**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
```

```
...
243. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 70:

Query Name - **Empty_Catch**

Severity - **Information**

```
173. private boolean nodeInterfaceServiceValid(Notification notif, Event
event) //notificationmanager.java
...
257. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 71:

Query Name - **Empty_Catch**

Severity - **Information**

```
292. public synchronized int getNoticeId() throws SQLException, IOException,
MarshalException, ValidationException //notificationmanager.java
...
312. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 72:

Query Name - **Empty_Catch**

Severity - **Information**

```
322. public boolean noticeOutstanding(int noticeId) throws IOException,
MarshalException, ValidationException //notificationmanager.java
...
354. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 73:

Query Name - **Empty_Catch**

Severity - **Information**

```
364. public Collection acknowledgeNotice(Event event, String uei, String[]
matchList) throws SQLException, IOException, MarshalException,
ValidationException //notificationmanager.java
...
452. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 74:

Query Name - **Empty_Catch**

Severity - **Information**

```
460. public List getActiveNodes() throws SQLException //notificationmanager.java
     ...
485. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 75:

Query Name - **Empty_Catch**

Severity - **Information**

```
493. public String getServiceNoticeStatus(String nodeID, String ipaddr, String
service) throws SQLException //notificationmanager.java
     ...
519. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 76:

Query Name - **Empty_Catch**

Severity - **Information**

```
527. public void updateNoticeWithUserInfo(String userId, int noticeId, String media,
String contactInfo, String autoNotify) throws
SQLException //notificationmanager.java
     ...
551. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 77:

Query Name - **Empty_Catch**

Severity - **Information**

```
561. public void insertNotice(int notifyId, Map params, String queueID) throws
SQLException //notificationmanager.java
     ...
621. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 78:

Query Name - **Empty_Catch**

Severity - **Information**

```
634. private int getServiceId(String service) throws
SQLException //notificationmanager.java
     ...
657. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 79:

Query Name - **Empty_Catch**

Severity - **Information**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java
       ...
698. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 80:

Query Name - **Empty_Catch**

Severity - **Information**

```
424. public void testInvalidRange() throws MarshalException, ValidationException,
UnknownHostException //snmpeventinfotest.java
       ...
447. } catch (IllegalArgumentException e) {
```

[Table of Contents] [Description]

Path 81:

Query Name - **Empty_Catch**

Severity - **Information**

```
225. public synchronized void stop() //client.java
       ...
230. } catch (IOException ex) {
```

[Table of Contents] [Description]

Path 82:

Query Name - **Empty_Catch**

Severity - **Information**

```
245. public void run() //client.java
       ...
286. } catch (IOException ex) {
```

[Table of Contents] [Description]

Path 83:

Query Name - **Empty_Catch**

Severity - **Information**

```
245. public void run() //client.java
       ...
330. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 84:

Query Name - **Empty_Catch**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
251. } catch (IOException ex1) {
```

[Table of Contents] [Description]

Path 85:

Query Name - **Empty_Catch**

Severity - **Information**

```
179. public synchronized void start() //dhcpd.java
       ...
266. } catch (IOException ex1) {
```

[Table of Contents] [Description]

Path 86:

Query Name - **Empty_Catch**

Severity - **Information**

```
281. public synchronized void stop() //dhcpd.java
       ...
297. } catch (IOException ex) {
```

[Table of Contents] [Description]

Path 87:

Query Name - **Empty_Catch**

Severity - **Information**

```
353. public void run() //dhcpd.java
       ...
383. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 88:

Query Name - **Empty_Catch**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
       ...
256. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 89:

Query Name - **Empty_Catch**

Severity - **Information**

```
233. private Poller(long timeout) throws IOException //poller.java
       ...
267. } catch (Throwable tx) {
```

[Table of Contents] [Description]

Path 90:

Query Name - **Empty_Catch**

Severity - **Information**

```
280. public void close() //poller.java
       ...
288. } catch (Throwable ex) {
```

[Table of Contents] [Description]

Path 91:

Query Name - **Empty_Catch**

Severity - **Information**

```
100. public void run() //receiver.java
       ...
141. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 92:

Query Name - **Empty_Catch**

Severity - **Information**

```
100. public void run() //receiver2.java
       ...
141. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 93:

Query Name - **Empty_Catch**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
121. } catch (IllegalStateException ex) {
```

[Table of Contents] [Description]

Path 94:

Query Name - **Empty_Catch**

Severity - **Information**

```
98. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
126. } catch (IllegalStateException ex) {
```

[Table of Contents] [Description]

Path 95:

Query Name - **Empty_Catch**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
181. } catch (UnknownHostException uhE) {
```

[Table of Contents] [Description]

Path 96:

Query Name - **Empty_Catch**

Severity - **Information**

```
162. static synchronized void dataSourceSync() throws
SQLException //discoveredipmgr.java
      ...
193. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 97:

Query Name - **Empty_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      ...
259. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 98:

Query Name - **Empty_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      ...
386. } catch (Exception exx) {
```

[Table of Contents] [Description]

Path 99:

Query Name - **Empty_Catch**

Severity - **Information**

```
421. public synchronized void start() //discovery.java
       ...
438. } catch (Exception exx) {
```

[Table of Contents] [Description]

Path 100:

Query Name - **Empty_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
       ...
460. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 101:

Query Name - **Empty_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
       ...
467. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 102:

Query Name - **Empty_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
       ...
477. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 103:

Query Name - **Empty_Catch**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
       ...
549. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 104:

Query Name - **Empty_Catch**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
       ...
554. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 105:

Query Name - **Empty_Catch**

Severity - **Information**

```
150. public static void loadConfiguration(Reader rdr) throws IOException,
MarshalException, ValidationException //eventconfigurationmanager.java
       ...
189. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 106:

Query Name - **Empty_Catch**

Severity - **Information**

```
232. public static void mergeConfiguration(Reader rdr) throws MarshalException,
ValidationException //eventconfigurationmanager.java
       ...
247. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 107:

Query Name - **Empty_Catch**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
       ...
443. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 108:

Query Name - **Empty_Catch**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
       ...
```

```
477. } catch (java.net.UnknownHostException e) {
```

[Table of Contents] [Description]

Path 109:

Query Name - **Empty_Catch**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
573. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 110:

Query Name - **Empty_Catch**

Severity - **Information**

```
417. public static String getValueOfParm(String parm, Event event) //eventutil.java
     ...
587. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 111:

Query Name - **Empty_Catch**

Severity - **Information**

```
834. private static String getNodeLabel(long nodeId) throws
SQLException //eventutil.java
     ...
856. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 112:

Query Name - **Empty_Catch**

Severity - **Information**

```
834. private static String getNodeLabel(long nodeId) throws
SQLException //eventutil.java
     ...
865. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 113:

Query Name - **Empty_Catch**

Severity - **Information**

```
888. private static String getIfAlias(long nodeId, String ipaddr) throws
SQLException //eventutil.java
      ...
911. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 114:

Query Name - **Empty_Catch**

Severity - **Information**

```
888. private static String getIfAlias(long nodeId, String ipaddr) throws
SQLException //eventutil.java
      ...
920. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 115:

Query Name - **Empty_Catch**

Severity - **Information**

```
943. private static String getAssetInfo(long nodeId, String assetField) throws
SQLException //eventutil.java
      ...
965. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 116:

Query Name - **Empty_Catch**

Severity - **Information**

```
943. private static String getAssetInfo(long nodeId, String assetField) throws
SQLException //eventutil.java
      ...
974. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 117:

Query Name - **Empty_Catch**

Severity - **Information**

```
66. private void findServer() throws
InstanceNotFoundException //eventhandlermbeanproxy.java
      ...
76. } catch (InstanceNotFoundException e) {
```

[Table of Contents] [Description]

Path 118:

Query Name - **Empty_Catch**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
433. } catch (IOException e2) {
```

[Table of Contents] [Description]

Path 119:

Query Name - **Empty_Catch**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
310. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 120:

Query Name - **Empty_Catch**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
       ...
345. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 121:

Query Name - **Empty_Catch**

Severity - **Information**

```
326. public int compare(Object o1, Object o2) //tcpstreamhandler.java
       ...
351. } catch (Throwable t) {
```

[Table of Contents] [Description]

Path 122:

Query Name - **Empty_Catch**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
204. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 123:

Query Name - **Empty_Catch**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
209. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 124:

Query Name - **Empty_Catch**

Severity - **Information**

```
166. public Map getIPServiceMap(String rule) //filter.java
       ...
235. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 125:

Query Name - **Empty_Catch**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
292. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 126:

Query Name - **Empty_Catch**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
297. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 127:

Query Name - **Empty_Catch**

Severity - **Information**

```
256. public List getIPList(String rule) throws FilterParseException //filter.java
       ...
319. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 128:

Query Name - **Empty_Catch**

Severity - **Information**

```
106. public synchronized Collection waitForAnticipated(long
millis) //eventanticipator.java
     ...
115. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 129:

Query Name - **Empty_Catch**

Severity - **Information**

```
157. public void run() //mockeventipcmanager.java
     ...
159. try { Thread.sleep(2000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 130:

Query Name - **Empty_Catch**

Severity - **Information**

```
189. public synchronized void finishProcessingEvents() //mockeventipcmanager.java
     ...
193. try { wait(); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 131:

Query Name - **Empty_Catch**

Severity - **Information**

```
380. public void testInvalidPoll() throws
UnknownHostException //mocknetworktest.java
     ...
387. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 132:

Query Name - **Empty_Catch**

Severity - **Information**

```
534. public void testScheduledOutages() //mocknetworktest.java
     ...
541. try { Thread.sleep(1000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 133:

Query Name - **Empty_Catch**

Severity - **Information**

```
624. public void run() //mocknetworktest.java
       ...
628. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 134:

Query Name - **Empty_Catch**

Severity - **Information**

```
624. public void run() //mocknetworktest.java
       ...
634. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 135:

Query Name - **Empty_Catch**

Severity - **Information**

```
711. public void run() //mocknetworktest.java
       ...
717. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 136:

Query Name - **Empty_Catch**

Severity - **Information**

```
105. private void waitFor(long millis) //pollanticipator.java
       ...
108. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 137:

Query Name - **Empty_Catch**

Severity - **Information**

```
241. public synchronized void stop() //notifd.java
       ...
250. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 138:

Query Name - **Empty_Catch**

Severity - **Information**

```
810. private void sleep(long millis) //notifdtest.java
       ...
813. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 139:

Query Name - **Empty_Catch**

Severity - **Information**

```
140. public synchronized Collection waitForAnticipated(long
millis) //notificationanticipator.java
       ...
149. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 140:

Query Name - **Empty_Catch**

Severity - **Information**

```
157. public void verifyAnticipated(long lastNotifyTime, long
waitTime, //notificationanticipator.java
       ...
169. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 141:

Query Name - **Empty_Catch**

Severity - **Information**

```
211. public void init() //outagemanager.java
       ...
249. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 142:

Query Name - **Empty_Catch**

Severity - **Information**

```
325. private void sleep(long millis) //outagetest.java
       326. try { Thread.sleep(millis); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 143:

Query Name - **Empty_Catch**

Severity - **Information**

```
379. public Date getServiceLostDate(int nodeId, String ipAddr, String svcName, int
serviceId) //defaultquerymanager.java
      ...
434. } catch (SQLException slqE) {
```

[Table of Contents] [Description]

Path 144:

Query Name - **Empty_Catch**

Severity - **Information**

```
748. private void sleep(long millis) //pollertest.java
      ...
751. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 145:

Query Name - **Empty_Catch**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
      ...
221. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 146:

Query Name - **Empty_Catch**

Severity - **Information**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //dnsmonitor.java
      ...
191. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 147:

Query Name - **Empty_Catch**

Severity - **Information**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
      ...
```

```
312. catch (SQLException ignore)
```

[Table of Contents] [Description]

Path 148:

Query Name - **Empty_Catch**

Severity - **Information**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
...
360. catch (SQLException ignore)
```

[Table of Contents] [Description]

Path 149:

Query Name - **Empty_Catch**

Severity - **Information**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
...
111. catch( SQLException ignore)
```

[Table of Contents] [Description]

Path 150:

Query Name - **Empty_Catch**

Severity - **Information**

```
104. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ntpmonitor.java
...
168. } catch (InterruptedIOException ex) {
```

[Table of Contents] [Description]

Path 151:

Query Name - **Empty_Catch**

Severity - **Information**

```
106. public void testErrorConditions() //snmpmonitortest.java
...
111. } catch (IllegalArgumentException e) {
```

[Table of Contents] [Description]

Path 152:

Query Name - **Empty_Catch**

Severity - **Information**

```
106. public void testErrorConditions() //snmpmonitortest.java
       ...
117. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 153:

Query Name - **Empty_Catch**

Severity - **Information**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
       ...
206. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 154:

Query Name - **Empty_Catch**

Severity - **Information**

```
1998. public void run() //pollablestest.java
       ...
2001. try { Thread.sleep(3000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 155:

Query Name - **Empty_Catch**

Severity - **Information**

```
2027. public void run() //pollablestest.java
       ...
2030. try { Thread.sleep(5000); } catch (InterruptedException e) {}
```

[Table of Contents] [Description]

Path 156:

Query Name - **Empty_Catch**

Severity - **Information**

```
422. public Operation makeUpdateOperation(String fileName, String
update) //queuingrrdstrategy.java
       ...
432. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 157:

Query Name - **Empty_Catch**

Severity - **Information**

```
518. public LinkedList getNext() //queuingrrdstrategy.java
       ...
530. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 158:

Query Name - **Empty_Catch**

Severity - **Information**

```
794. public void run() //queuingrrdstrategy.java
       ...
810. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 159:

Query Name - **Empty_Catch**

Severity - **Information**

```
121. public static int getProperty(String name, int defaultVal) //rrdconfig.java
       ...
126. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 160:

Query Name - **Empty_Catch**

Severity - **Information**

```
142. public static long getProperty(String name, long defaultVal) //rrdconfig.java
       ...
147. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 161:

Query Name - **Empty_Catch**

Severity - **Information**

```
157. private static String makeZeroUpdate(int fileNum, String
update) //rrdstresser.java
       ...
167. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 162:

Query Name - **Empty_Catch**

Severity - **Information**

```
314. public void execute(String[] args) throws Exception //rrdstresser.java
      |...
      329. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 163:

Query Name - **Empty_Catch**

Severity - **Information**

```
174. public void run() //executor.java
      |...
      259. catch (Exception ex) {
```

[Table of Contents] [Description]

Path 164:

Query Name - **Empty_Catch**

Severity - **Information**

```
173. public synchronized void stop() //scriptd.java
      |...
      180. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 165:

Query Name - **Empty_Catch**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
      |...
      275. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 166:

Query Name - **Empty_Catch**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
      |...
      372. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 167:

Query Name - **Empty_Catch**

Severity - **Information**

```
93. static synchronized void dataSourceSync() throws SQLException //trapdipmgr.java
      ...
121. } catch (SQLException sqlE) {
```

[Table of Contents] [Description]

Path 168:

Query Name - **Empty_Catch**

Severity - **Information**

```
203. public void run() //vacuumd.java
      ...
228. } catch (InterruptedException e) {
```

[Table of Contents] [Description]

Path 169:

Query Name - **Empty_Catch**

Severity - **Information**

```
213. private Set getAllManagedInterfaces() //scheduler.java
      ...
250. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 170:

Query Name - **Empty_Catch**

Severity - **Information**

```
213. private Set getAllManagedInterfaces() //scheduler.java
      ...
256. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 171:

Query Name - **Empty_Catch**

Severity - **Information**

```
275. void addToKnownAddresses(InetAddress address, int scanLevel) throws
SQLException //scheduler.java
      ...
303. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 172:

Query Name - **Empty_Catch**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      408. } catch (XmlRpcException e) {
```

[Table of Contents] [Description]

Path 173:

Query Name - **Empty_Catch**

Severity - **Information**

```
394. private boolean sendXmlrpcRequest(String command, Vector
params) //xmlrpcnotifier.java
      |
      ...
      410. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 174:

Query Name - **Empty_Catch**

Severity - **Information**

```
437. public void createConnection() //xmlrpcnotifier.java
      |
      ...
      482. } catch (InterruptedException ie) {
```

[Table of Contents] [Description]

Path 175:

Query Name - **Empty_Catch**

Severity - **Information**

```
127. public static String getHostname(String ipAddress) //util.java
      |
      ...
      132. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 176:

Query Name - **Empty_Catch**

Severity - **Information**

```
172. public static String resolveIpAddress(String ipAddress) //util.java
      |
      ...
      177. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 177:

Query Name - **Empty_Catch**

Severity - **Information**

```
129. private int getNodeId(String ipaddr) throws
SQLException //addnewinterfaceservlet.java
      |...
      |149. } catch (SQLException e1) {
```

[Table of Contents] [Description]

Path 178:

Query Name - **Empty_Catch**

Severity - **Information**

```
129. private int getNodeId(String ipaddr) throws
SQLException //addnewinterfaceservlet.java
      |...
      |155. } catch (SQLException e2) {
```

[Table of Contents] [Description]

Path 179:

Query Name - **Empty_Catch**

Severity - **Information**

```
63. public void init() throws ServletException //deletegetnodesservlet.java
      |...
      |66. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 180:

Query Name - **Empty_Catch**

Severity - **Information**

```
86. private List getAllNodes(HttpSession userSession) throws
SQLException //deletegetnodesservlet.java
      |...
      |114. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 181:

Query Name - **Empty_Catch**

Severity - **Information**

```
117. private List getInterfaces(HttpSession userSession, int nodeId) throws
SQLException //getinterfacesservlet.java
      |...
```

```
158. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 182:

Query Name - **Empty_Catch**

Severity - **Information**

```
73. public void init() throws ServletException //getnodesservlet.java
      ...
76. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 183:

Query Name - **Empty_Catch**

Severity - **Information**

```
96. private List getAllNodes(HttpSession userSession) throws
SQLException //getnodesservlet.java
      ...
161. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 184:

Query Name - **Empty_Catch**

Severity - **Information**

```
69. public void init() throws ServletException //snmpgetinterfacesservlet.java
      ...
72. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 185:

Query Name - **Empty_Catch**

Severity - **Information**

```
98. private List getNodeInterfaces(HttpSession userSession, int nodeid) throws
SQLException //snmpgetinterfacesservlet.java
      ...
132. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 186:

Query Name - **Empty_Catch**

Severity - **Information**

```
70. public void init() throws ServletException //snmpgetnodesservlet.java
      ...
73. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 187:

Query Name - **Empty_Catch**

Severity - **Information**

```
91. private List getAllNodes(HttpSession userSession) throws
SQLException //snmpgetnodesservlet.java
      ...
128. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 188:

Query Name - **Empty_Catch**

Severity - **Information**

```
110. public Category(String aDataString) //category.java
      ...
135. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 189:

Query Name - **Empty_Catch**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
      ...
114. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 190:

Query Name - **Empty_Catch**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
      ...
125. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 191:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 192:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //negativeservicefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 193:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 194:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //servicefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 195:

Query Name - **Empty_Catch**

Severity - **Information**

```
285. public String getTextDescription() //noticefactory.java
      ...
289. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 196:

Query Name - **Empty_Catch**

Severity - **Information**

```
365. public String getTextDescription() //noticefactory.java
       ...
369. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 197:

Query Name - **Empty_Catch**

Severity - **Information**

```
462. public static boolean canDisplayEvent(int eventId) //noticefactory.java
       ...
489. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 198:

Query Name - **Empty_Catch**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noticequeryservlet.java
       ...
107. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 199:

Query Name - **Empty_Catch**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noticequeryservlet.java
       ...
117. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 200:

Query Name - **Empty_Catch**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
       ...
109. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 201:

Query Name - **Empty_Catch**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
      ...
119. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 202:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 203:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
      ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 204:

Query Name - **Empty_Catch**

Severity - **Information**

```
112. public static Properties loadLocal(String aFileName) //propertyloader.java
      ...
121. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 205:

Query Name - **Empty_Catch**

Severity - **Information**

```
139. public static Properties loadLocal(String aFileName, Properties
someProperties) //propertyloader.java
      ...
148. } catch (IOException e) {
```

[Table of Contents] [Description]

Path 206:

Query Name - **Empty_Catch**

Severity - **Information**

```
147. private String getHost() //xmlheader.java
       ...
153. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 207:

Query Name - **Empty_Catch**

Severity - **Information**

```
165. public String getVersion() //xmlwriter.java
       ...
176. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 208:

Query Name - **Empty_Catch**

Severity - **Information**

```
309. protected static double parseDouble(String s) //rrdinfo.java
       ...
314. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 209:

Query Name - **Empty_Catch**

Severity - **Information**

```
313. protected static double parseDouble(String s) //rrdinfo.java
       ...
318. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 210:

Query Name - **Empty_Catch**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
       ...
115. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 211:

Query Name - **Empty_Catch**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
    ...
126. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 212:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //negativenodefilter.java
    ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 213:

Query Name - **Empty_Catch**

Severity - **Information**

```
57. public String getTextDescription() //nodefilter.java
    ...
61. } catch (SQLException e) {
```

[Table of Contents] [Description]

Path 214:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <!-- //index.jsp
    ...
180. } catch (Exception e) { /*if factory can't be initialized, status is
already 'Unknown'*/ }
```

[Table of Contents] [Description]

Path 215:

Query Name - **Empty_Catch**

Severity - **Information**

```
180. private String getEventsList() //rearrangeevents.jsp
    ...
187. } catch (Exception e ) {
```

[Table of Contents] [Description]

Path 216:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
     ...
308. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 217:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
     ...
390. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 218:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
     ...
400. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 219:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
     ...
409. } catch (NumberFormatException e) {
```

[Table of Contents] [Description]

Path 220:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <!-- //eventlist.jsp
     ...
72. catch( NumberFormatException e ) {}
```

[Table of Contents] [Description]

Path 221:

Query Name - **Empty_Catch**

Severity - **Information**

```
19. public void init() throws ServletException //header.jsp
      ...
      23. catch( Exception e ) {/*notice status will be unknown if the factory can't
      be initialized*/}
```

[Table of Contents] [Description]

Path 222:

Query Name - **Empty_Catch**

Severity - **Information**

```
1. <%-- //header.jsp
      ...
      58. } catch (Exception e) { /*if factory can't be initialized, status is
      already 'Unknown'*/ }
```

[Table of Contents] [Description]

Path 223:

Query Name - **Empty_Catch**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
      109. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 224:

Query Name - **Empty_Catch**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
      117. try { connection.close(); } catch(Throwable tx) { }
```

[Table of Contents] [Description]

Path 225:

Query Name - **Empty_Catch**

Severity - **Information**

```
94. public void run() //sockettest.java
      ...
      139. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 226:

Query Name - **Empty_Catch**

Severity - **Information**

```
94. public void run() //sockettest.java
      |
      ...
      |
153. try { connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 227:

Query Name - **Empty_Catch**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
      |
294. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 228:

Query Name - **Empty_Catch**

Severity - **Information**

```
282. public void run() //sockettest.java
      |
      ...
      |
331. try { m_connection.close(); } catch(Throwable t) { }
```

[Table of Contents] [Description]

Path 229:

Query Name - **Empty_Catch**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
      |
      ...
      |
453. catch(InterruptedException e)
```

[Table of Contents] [Description]

Path 230:

Query Name - **Empty_Catch**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
      |
      ...
      |
596. catch(InterruptedException e)
```

[Table of Contents] [Description]

Path 231:

Query Name - **Empty_Catch**

Severity - **Information**

```
40. protected void sendCommand(String command) //simulatorcontroller.java
      |
      ...
      50. } catch (IllegalThreadStateException e) {
```

[Table of Contents] [Description]

Path 232:

Query Name - **Empty_Catch**

Severity - **Information**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
      |
      ...
      98. } catch (IllegalThreadStateException e) {
```

[Table of Contents] [Description]

Bad Practices: Threads

CWE ID	383
Description	Thread management in a Web application is forbidden in some circumstances and is always highly error prone.
Affected Resource	System Process
Potential Mitigations	For EBJ, use framework approaches for parallel execution, instead of using threads.
Context Notes	Thread management in a web application is forbidden by the J2EE standard in some circumstances and is always highly error prone. Managing threads is difficult and is likely to interfere in unpredictable ways with the behavior of the application container. Even without interfering with the container, thread management usually leads to bugs that are hard to detect and diagnose like deadlock, race conditions, and other synchronization errors.
Node Relationships	ChildOf - J2EE Time and State Issues (381) ChildOf - Weaknesses that Affect System Processes (634) ParentOf - Use of Singleton Pattern in a Non-thread-safe Manner (543)
Source Taxonomies	7 Pernicious Kingdoms - J2EE Bad Practices: Threads
Applicable Platforms	Java .Net

Path 1:

Query Name - **Threads_in_WebApp**

Severity - **Information**

```
93. public final class UnicastListener extends Thread //client.java
      |
```

[Table of Contents] [Description]

Path 2:

Query Name - **Threads_in_WebApp**

Severity - **Information**

```
616. class EventSender extends Thread //mocknetworktest.java
      |
```

[Table of Contents] [Description]

Path 3:

Query Name - **Threads_in_WebApp**

Severity - **Information**

```
703. class PollerThread extends Thread //mocknetworktest.java
      |
```

[Table of Contents] [Description]

Path 4:

Query Name - **Threads_in_WebApp**

Severity - **Information**

```
76. public class NotificationTask extends Thread //notificationtask.java
      |
```

[Table of Contents] [Description]

Path 5:

Query Name - **Threads_in_WebApp**

Severity - **Information**

```
59. class StreamGobbler extends Thread //streamgobbler.java
      |
```

[Table of Contents] [Description]

J2EE Bad Practices: Sockets

CWE ID	246
Description	Socket-based communication in web applications is prone to error.
Weakness Ordinality	Resultant (Weakness is typically related to the presence of some other weaknesses)

Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Use framework method calls instead of using sockets directly.
Context Notes	The J2EE standard permits the use of sockets only for the purpose of communication with legacy systems when no higher-level protocol is available. Authoring your own communication protocol requires wrestling with difficult security issues, including: - In-band versus out-of-band signaling - Compatibility between protocol versions - Channel security - Error handling - Network constraints (firewalls) - Session management Without significant scrutiny by a security expert, chances are good that a custom communication protocol will suffer from security problems. Many of the same issues apply to a custom implementation of a standard protocol. While there are usually more resources available that address security concerns related to implementing a standard protocol, these resources are also available to attackers.
Node Relationships	ChildOf - API Abuse (227)
Source Taxonomies	7 Pernicious Kingdoms - J2EE Bad Practices: Sockets
Applicable Platforms	Java

Path 1:

Query Name - **Sockets_in_WebApp**

Severity - **Information**

```
353. public void run() //dhcpd.java
       ...
392. sock = m_server.accept();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Sockets_in_WebApp**

Severity - **Information**

```
238. public void run() //tcpserver.java
       ...
316. Socket newbie = m_tcpSock.accept();
```

[Table of Contents] [Description]

Failure to Account for Default Case in Switch

CWE ID	478
Description	The failure to account for the default case in switch statements may lead to complex logical errors and may aid in other, unexpected security-related conditions.
Common Consequences	Undefined: Depending on the logical circumstances involved, any consequences may result: e.g., issues of confidentiality, authentication, authorization, availability, integrity, accountability, or non-repudiation.

Potential Mitigations	Implementation: Ensure that there are no unaccounted for cases, when adjusting flow or values based on the value of a given variable. In switch statements, this can be accomplished through the use of the default label.

Demonstrative Examples	In general, a safe switch statement has this form:
	
	switch (value) {
	  case 'A':
	  printf("A!\n");
	  break;
	  case 'B':
	  printf("B!\n");
	  break;
	  default:
	  printf("Neither A nor B\n");
	}

Context Notes	This is because the assumption cannot be made that all possible cases are accounted for. A good practice is to reserve the default case for error handling. This flaw represents a common problem in software development, in which not all possible values for a variable are considered or handled by a given process. Because of this, further decisions are made based on poor information, and cascading failure results. This cascading failure may result in any number of security issues, and constitutes a significant failure in the system. In the case of switch style statements, the very simple act of creating a default case can mitigate this situation, if done correctly. Often however, the default cause is used simply to represent an assumed option, as opposed to working as a sanity check. This is poor practice and in some cases is as bad as omitting a default case entirely.
Node Relationships	ChildOf - Code Quality (398)
Source Taxonomies	CLASP - Failure to account for default case in switch
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **No_Default_Case**

Severity - **Information**

```
4266. static final private void jj_rescan_token() //parsemib.java
      ...
4273. switch (i) {
```

[Table of Contents] [Description]

Path 2:

Query Name - **No_Default_Case**

Severity - **Information**

```
139. private void adjust() //runnableconsumerthreadpool.java
     ...
156. switch (m_fibers[x].getStatus()) {
```

[Table of Contents] [Description]

Path 3:

Query Name - **No_Default_Case**

Severity - **Information**

```
129. public String toString() //constraint.java
     ...
135. switch (m_type) {
```

[Table of Contents] [Description]

Path 4:

Query Name - **No_Default_Case**

Severity - **Information**

```
77. private static int[] convert(String idstr) //snmpobjectid.java
     ...
153. switch (idArray[idArrayNdx]) {
```

[Table of Contents] [Description]

Path 5:

Query Name - **No_Default_Case**

Severity - **Information**

```
587. void transmit(SnmpRequest req) throws SnmpPduEncodingException,
AsnEncodingException, java.io.IOException //snmpsession.java
     ...
607. switch (pdu.getCommand()) {
```

[Table of Contents] [Description]

Path 6:

Query Name - **No_Default_Case**

Severity - **Information**

```
114. static SnmpSyntax getSyntaxObject(byte asnType) //snmputil.java
     ...
116. switch (asnType) {
```

[Table of Contents] [Description]

Path 7:

Query Name - **No_Default_Case**

Severity - **Information**

```
374. private static synchronized int getDays(boolean isLeap, int
month) //availabilitydata.java
      |375. switch (month) {
```

[Table of Contents] [Description]

Path 8:

Query Name - **No_Default_Case**

Severity - **Information**

```
1104. private static synchronized int getDays(boolean isLeap, int
month) //availcalculations.java
      |1105. switch (month) {
```

[Table of Contents] [Description]

Path 9:

Query Name - **No_Default_Case**

Severity - **Information**

```
470. public NsclientPacket processCheckCommand(short type, //nsclientmanager.java
      |...
      |473. switch (type) {
```

[Table of Contents] [Description]

Path 10:

Query Name - **No_Default_Case**

Severity - **Information**

```
163. private void xmlrpcNotificationEventHandler(Event
event) //eventqueueprocessor.java
      |...
      |229. switch (status) {
```

[Table of Contents] [Description]

Path 11:

Query Name - **No_Default_Case**

Severity - **Information**

```
1. <!-- //error.jsp
      |...
      |83. switch(errorcode)
```

[Table of Contents] [Description]

Erroneous String Compare

CWE ID	597
Description	Strings should be compared with the equals() method, not == or !=

Demonstrative Examples	The following branch will never be taken.
	
	if (args[0] == STRING_CONSTANT) {
	  logger.info("miracle");
	}

Context Notes	Using == or != to compare two strings for equality actually compares two objects for equality, not their values. Chances are good that the two references will never be equal.
Node Relationships	ChildOf - Incorrect Object Comparison: Syntactic (595) ChildOf - Incorrect Object Comparison: Syntactic (595) ChildOf - String Errors (133)

Path 1:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
881. private void openClosedElements(DOMBuilder rtf) //verbatim.java
      |
      ...
      |
896. if (((ns == null || ns == "http://www.w3.org/1999/xhtml")
```

[Table of Contents] [Description]

Path 2:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
369. public synchronized void renameUser(String oldName, String newName) throws
Exception //groupmanager.java
      |
      ...
      |
371. if (oldName == null || newName == null || oldName == "" || newName == "")
     {
```

[Table of Contents] [Description]

Path 3:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
369. public synchronized void renameUser(String oldName, String newName) throws
Exception //groupmanager.java
      |
      ...
      |
371. if (oldName == null || newName == null || oldName == "" || newName == "")
     {
```

[Table of Contents] [Description]

Path 4:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
215. public synchronized void renameUser(String oldName, String newName) throws
Exception //viewfactory.java
      ...
      217. if (oldName == null || newName == null || oldName == "" || newName == "")
      {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
215. public synchronized void renameUser(String oldName, String newName) throws
Exception //viewfactory.java
      ...
      217. if (oldName == null || newName == null || oldName == "" || newName == "")
      {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
304. public synchronized void renameView(String oldName, String newName) throws
Exception //viewfactory.java
      ...
      308. if (oldName != null || !oldName.equals("") || newName != "" || oldName !=
      "") {
```

[Table of Contents] [Description]

Path 7:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
304. public synchronized void renameView(String oldName, String newName) throws
Exception //viewfactory.java
      ...
      308. if (oldName != null || !oldName.equals("") || newName != "" || oldName !=
      "") {
```

[Table of Contents] [Description]

Path 8:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
161. public boolean isValid() //nessusscanconfiguration.java
     ...
164. boolean retval = ((hostname != null) && (username != null) && (username !
= "") && (password != null) && (password != "") && (scanLevel > 0) &&
(scanLevel < 5) && (targetAddress != null) && (hostport > 0) && (hostport < (1
<< 16)));
```

[Table of Contents] [Description]

Path 9:

Query Name - **Erroneous_String_Compare**

Severity - **Information**

```
161. public boolean isValid() //nessusscanconfiguration.java
     ...
164. boolean retval = ((hostname != null) && (username != null) && (username !
= "") && (password != null) && (password != "") && (scanLevel > 0) &&
(scanLevel < 5) && (targetAddress != null) && (hostport > 0) && (hostport < (1
<< 16)));
```

[Table of Contents] [Description]

Magic Numbers

CWE ID	10017
Description	Using literals in the code makes the code less readable and maintainable.
Likelihood of Exploit	Low
Common Consequences	Code become less readable and maintainable.
Potential Mitigations	Define constants with meaningful names and use them instead.
Applicable Platforms	All

Path 1:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1006. static private final int jjMoveNfa_0(int startState, int
curPos) //parsemibtokenmanager.java
      ...
1013. int j, kind = 0x7fffffff;
```

[Table of Contents] [Description]

Path 2:

Query Name - **Magic_Numbers**

Severity - **Information**

```
47. public class CalloutEmitter extends CopyEmitter //calloutemitter.java
```

```
...
64. protected int defaultColumn = 60;
```

[Table of Contents] [Description]

Path 3:

Query Name - **Magic_Numbers**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
...
62. int lineLimit = 100;
```

[Table of Contents] [Description]

Path 4:

Query Name - **Magic_Numbers**

Severity - **Information**

```
50. public class NumberLinesEmitter extends CopyEmitter //numberlinesemitter.java
...
70. protected int modulus = 5;
```

[Table of Contents] [Description]

Path 5:

Query Name - **Magic_Numbers**

Severity - **Information**

```
50. public class NumberLinesEmitter extends CopyEmitter //numberlinesemitter.java
...
73. protected int width = 3;
```

[Table of Contents] [Description]

Path 6:

Query Name - **Magic_Numbers**

Severity - **Information**

```
56. public class Table //table.java
...
58. private static int pixelsPerInch = 96;
```

[Table of Contents] [Description]

Path 7:

Query Name - **Magic_Numbers**

Severity - **Information**

```
59. public class Verbatim //verbatim.java
```

```
...
74. private static int defaultColumn = 60;
```

[Table of Contents] [Description]

Path 8:

Query Name - **Magic_Numbers**

Severity - **Information**

```
59. public class Verbatim //verbatim.java
...
80. private static int graphicsMax = 10;
```

[Table of Contents] [Description]

Path 9:

Query Name - **Magic_Numbers**

Severity - **Information**

```
311. private static void setupCallouts(Context context) //verbatim.java
...
317. int unicodeStart = 49;
```

[Table of Contents] [Description]

Path 10:

Query Name - **Magic_Numbers**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
...
64. int lineLimit = 100;
```

[Table of Contents] [Description]

Path 11:

Query Name - **Magic_Numbers**

Severity - **Information**

```
63. public class Table //table.java
...
65. private static int pixelsPerInch = 96;
```

[Table of Contents] [Description]

Path 12:

Query Name - **Magic_Numbers**

Severity - **Information**

```
78. public class Verbatim //verbatim.java
      |...
      |104. private int graphicsMax = 10;
```

[Table of Contents] [Description]

Path 13:

Query Name - **Magic_Numbers**

Severity - **Information**

```
444. public DocumentFragment insertCallouts (ExpressionContext
context, //verbatim.java
      |...
      |464. int dMax = 10;
```

[Table of Contents] [Description]

Path 14:

Query Name - **Magic_Numbers**

Severity - **Information**

```
90. public static char[] encodeBase64(byte[] data) //base64.java
      |...
      |117. for (int j = 3; j >= pad; j--) {
```

[Table of Contents] [Description]

Path 15:

Query Name - **Magic_Numbers**

Severity - **Information**

```
63. public int execute(String commandLine, List arguments) //commandexecutor.java
      |...
      |127. long timeout = 30000; // wait for 60 seconds
```

[Table of Contents] [Description]

Path 16:

Query Name - **Magic_Numbers**

Severity - **Information**

```
161. SnmpPortal(SnmpPacketHandler handler, AsnEncoder encoder, int port) throws
SocketException //snmpportal.java
      |...
      |194. int timeout = 3000; // Default socket timeout is 3 seconds
```

[Table of Contents] [Description]

Path 17:

Query Name - **Magic_Numbers**

Severity - **Information**

```
473. public int buildInteger32(byte[] buf, int startOffset, byte asnType, int
asnInt32) throws AsnEncodingException //berencoder.java
      |...
      479. int mask = 0xff800000; // negative :)
```

[Table of Contents] [Description]

Path 18:

Query Name - **Magic_Numbers**

Severity - **Information**

```
473. public int buildInteger32(byte[] buf, int startOffset, byte asnType, int
asnInt32) throws AsnEncodingException //berencoder.java
      |...
      480. int intSz = 4; // int == 32-bits == 4 bytes in java
```

[Table of Contents] [Description]

Path 19:

Query Name - **Magic_Numbers**

Severity - **Information**

```
604. public int buildUInteger32(byte[] buf, int startOffset, byte asnType, long
asnUInt32) throws AsnEncodingException //berencoder.java
      |...
      616. int intSz = 4; // int == 32-bits == 4 bytes in java
```

[Table of Contents] [Description]

Path 20:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1066. public int buildObjectId(byte[] buf, int startOffset, byte asnType, int[]
oids) throws AsnEncodingException //berencoder.java
      |...
      1096. int oidNdx = 2;
```

[Table of Contents] [Description]

Path 21:

Query Name - **Magic_Numbers**

Severity - **Information**

```
235. public void testBuiltCalendarReport () //availabilityreporttest.java
      |...
      238. long oneHundred = 100;
```

[Table of Contents] [Description]

Path 22:

Query Name - **Magic_Numbers**

Severity - **Information**

```
52. public class DBTools //dbtools.java
      |...
      |68. public static int MIN_PORT_VALUE = 1024;
```

[Table of Contents] [Description]

Path 23:

Query Name - **Magic_Numbers**

Severity - **Information**

```
52. public class DBTools //dbtools.java
      |...
      |73. public static int MAX_PORT_VALUE = 65535;
```

[Table of Contents] [Description]

Path 24:

Query Name - **Magic_Numbers**

Severity - **Information**

```
118. private boolean isServer(InetAddress host, int retry, int timeout, String
script, String args, RE regex, StringBuffer bannerResult, String hoption, String
toption) //gpplugin.java
      |...
      |127. int exitStatus = 100;
```

[Table of Contents] [Description]

Path 25:

Query Name - **Magic_Numbers**

Severity - **Information**

```
196. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //httpplugin.java
      |...
      |233. int maxRetCode = 399;
```

[Table of Contents] [Description]

Path 26:

Query Name - **Magic_Numbers**

Severity - **Information**

```
81. public final class IfXTableEntry extends java.util.TreeMap //ifxtableentry.java
      |...
      |150. static int NUM_OIDS = 19;
```

[Table of Contents] [Description]

Path 27:

Query Name - **Magic_Numbers**

Severity - **Information**

```
96. final class SnmpCollector implements ServiceCollector //snmpcollector.java
      ...
161. private static int DEFAULT_MAX_VARS_PER_PDU = 10;
```

[Table of Contents] [Description]

Path 28:

Query Name - **Magic_Numbers**

Severity - **Information**

```
267. public static String toIpAddr (long ip) //snmppeerfactory.java
      ...
269. for ( int shift=24; shift >0; shift-=8 ) {
```

[Table of Contents] [Description]

Path 29:

Query Name - **Magic_Numbers**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      ...
299. int retries = 3;
```

[Table of Contents] [Description]

Path 30:

Query Name - **Magic_Numbers**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      ...
322. int retries = 3;
```

[Table of Contents] [Description]

Path 31:

Query Name - **Magic_Numbers**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      ...
347. int retries = 3;
```

[Table of Contents] [Description]

Path 32:

Query Name - **Magic_Numbers**

Severity - **Information**

```
347. public void run() //tcprecordhandler.java
       ...
389. int level = 8;
```

[Table of Contents] [Description]

Path 33:

Query Name - **Magic_Numbers**

Severity - **Information**

```
50. public class NotificationAnticipator //notificationanticipator.java
       ...
57. long m_expectedDiff = 1000;;
```

[Table of Contents] [Description]

Path 34:

Query Name - **Magic_Numbers**

Severity - **Information**

```
103. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //gpmonitor.java
       ...
162. int exitStatus = 100;
```

[Table of Contents] [Description]

Path 35:

Query Name - **Magic_Numbers**

Severity - **Information**

```
85. final public class SnmpMonitor extends IPv4Monitor //snmpmonitor.java
       ...
96. private static int DEFAULT_PORT = 161;
```

[Table of Contents] [Description]

Path 36:

Query Name - **Magic_Numbers**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
       ...
94. int port = 1248;
```

[Table of Contents] [Description]

Path 37:

Query Name - **Magic_Numbers**

Severity - **Information**

```
74. public class NsclientManager //nsclientmanager.java
      ...
78. public static int DEFAULT_SOCKET_TIMEOUT = 5000;
```

[Table of Contents] [Description]

Path 38:

Query Name - **Magic_Numbers**

Severity - **Information**

```
74. public class NsclientManager //nsclientmanager.java
      ...
83. public static int DEFAULT_PORT = 1248;
```

[Table of Contents] [Description]

Path 39:

Query Name - **Magic_Numbers**

Severity - **Information**

```
372. public void testCreateNode() //pollablestest.java
      373. int nodeId = 99;
```

[Table of Contents] [Description]

Path 40:

Query Name - **Magic_Numbers**

Severity - **Information**

```
383. public void testCreateInterface() throws
UnknownHostException //pollablestest.java
      384. int nodeId = 99;
```

[Table of Contents] [Description]

Path 41:

Query Name - **Magic_Numbers**

Severity - **Information**

```
401. public void testCreateService() throws Exception //pollablestest.java
      402. int nodeId = 99;
```

[Table of Contents] [Description]

Path 42:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1512. public void testSchedule() //pollablestest.java
      ...
1534. for(int downFor = 100; downFor < 500; downFor += 100) {
```

[Table of Contents] [Description]

Path 43:

Query Name - **Magic_Numbers**

Severity - **Information**

```
1512. public void testSchedule() //pollablestest.java
      ...
1540. for(int downFor = 500; downFor < 1500; downFor += 200) {
```

[Table of Contents] [Description]

Path 44:

Query Name - **Magic_Numbers**

Severity - **Information**

```
319. public InputStream createGraph(String command, File workDir) throws
IOException, org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
      ...
328. int height = 100;
```

[Table of Contents] [Description]

Path 45:

Query Name - **Magic_Numbers**

Severity - **Information**

```
319. public InputStream createGraph(String command, File workDir) throws
IOException, org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
      ...
329. int width = 400;
```

[Table of Contents] [Description]

Path 46:

Query Name - **Magic_Numbers**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
367. for (int i = 2; i < pdu.getLength(); i++) {
```

[Table of Contents] [Description]

Path 47:

Query Name - **Magic_Numbers**

Severity - **Information**

```
254. private void process(Trapd.V2TrapInformation info) //trapqueueprocessor.java
      ...
475. for (int i = 2; i < pdu.getLength(); i++) {
```

[Table of Contents] [Description]

Path 48:

Query Name - **Magic_Numbers**

Severity - **Information**

```
62. public static void createAndSendXmlrpcNotificationEvent(long txNo, String
sourceUei, String message, int status, String generator) //xmlrpcutil.java
      ...
130. int failureFlag = 2;
```

[Table of Contents] [Description]

Path 49:

Query Name - **Magic_Numbers**

Severity - **Information**

```
468. int writeBytes(byte[] dest, int offset) //ipheader.java
      ...
476. for (int i = 3; i >= 0; i++)
```

[Table of Contents] [Description]

Path 50:

Query Name - **Magic_Numbers**

Severity - **Information**

```
47. public class MapNode //mapnode.java
      ...
67. public static int defaultNodeWidth = 100;
```

[Table of Contents] [Description]

Path 51:

Query Name - **Magic_Numbers**

Severity - **Information**

```
47. public class MapNode //mapnode.java
      ...
69. public static int defaultNodeHeight = 90;
```

[Table of Contents] [Description]

Path 52:

Query Name - **Magic_Numbers**

Severity - **Information**

```
47. public class MapNode //mapnode.java
      ...
71. public static int widthBuffer = 10;
```

[Table of Contents] [Description]

Path 53:

Query Name - **Magic_Numbers**

Severity - **Information**

```
47. public class MapNode //mapnode.java
      ...
73. public static int heightBuffer = 20;
```

[Table of Contents] [Description]

Path 54:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
      ...
228. for (int i = 5; i <= 5; i++) {
```

[Table of Contents] [Description]

Path 55:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
      ...
241. for (int i = 6; i <= 7; i++) {
```

[Table of Contents] [Description]

Path 56:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
      ...
254. for (int i = 8; i <= 10; i++) {
```

[Table of Contents] [Description]

Path 57:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
       ...
267. for (int i = 11; i <= 11; i++) {
```

[Table of Contents] [Description]

Path 58:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
       ...
280. for (int i = 12; i <= 12; i++) {
```

[Table of Contents] [Description]

Path 59:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
       ...
293. for (int i = 13; i <= 14; i++) {
```

[Table of Contents] [Description]

Path 60:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
       ...
306. for (int i = 15; i <= 16; i++) {
```

[Table of Contents] [Description]

Path 61:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
       ...
319. for (int i = 17; i <= 21; i++) {
```

[Table of Contents] [Description]

Path 62:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
     ...
332. for (int i = 22; i <= 23; i++) {
```

[Table of Contents] [Description]

Path 63:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
     ...
345. for (int i = 24; i <= 27; i++) {
```

[Table of Contents] [Description]

Path 64:

Query Name - **Magic_Numbers**

Severity - **Information**

```
203. public Vector getTestNodes() //mapnodefactory.java
     ...
358. for (int i = 28; i <= 36; i++) {
```

[Table of Contents] [Description]

Path 65:

Query Name - **Magic_Numbers**

Severity - **Information**

```
44. <%! //assignnotices.jsp
     ...
53. private int controlCount = 5; /*there are five buttons before the check
boxes begin*/
```

[Table of Contents] [Description]

Confusing naming

CWE ID	10032
Description	Naming a method and field with the same name, may be confusing to developers
Likelihood of Exploit	Low
Common Consequences	Difficulty to maintain the code
Potential Mitigations	Renaming methods and fields
Applicable Platforms	All

Path 1:

Query Name - **Confusing_Naming**

Severity - **Information**

```
27. class ParseMib/*@bgen(jjtree)*/implements ParseMibTreeConstants,
ParseMibConstants /*@bgen(jjtree)*/ //parsemib.java
    ...
4003. static private int jj_ntk;
```

[Table of Contents] [Description]

Path 2:

Query Name - **Confusing_Naming**

Severity - **Information**

```
27. class ParseMib/*@bgen(jjtree)*/implements ParseMibTreeConstants,
ParseMibConstants /*@bgen(jjtree)*/ //parsemib.java
    ...
4010. static private int[] jj_la1_0;
```

[Table of Contents] [Description]

Path 3:

Query Name - **Confusing_Naming**

Severity - **Information**

```
27. class ParseMib/*@bgen(jjtree)*/implements ParseMibTreeConstants,
ParseMibConstants /*@bgen(jjtree)*/ //parsemib.java
    ...
4011. static private int[] jj_la1_1;
```

[Table of Contents] [Description]

Path 4:

Query Name - **Confusing_Naming**

Severity - **Information**

```
27. class ParseMib/*@bgen(jjtree)*/implements ParseMibTreeConstants,
ParseMibConstants /*@bgen(jjtree)*/ //parsemib.java
    ...
4012. static private int[] jj_la1_2;
```

[Table of Contents] [Description]

Path 5:

Query Name - **Confusing_Naming**

Severity - **Information**

```
44. public class UnwrapLinksEmitter extends CopyEmitter //unwraplinksemitter.java
```

```
...
65. protected boolean tryAgain = false;
```

[Table of Contents] [Description]

Path 6:

Query Name - **Confusing_Naming**

Severity - **Information**

```
59. class StreamGobbler extends Thread //streamgobbler.java
...
68. private boolean quit = false;
```

[Table of Contents] [Description]

Leftover Debug Code

CWE ID	489
Description	Debug code can create unintended entry points in an application.
Common Consequences	The severity of the exposed debug application will depend on the particular instance. At the least, it will give an attacker sensitive information about the settings and mechanics of web applications on the server. At worst, as is often the case, the debug application will allow an attacker complete control over the web application and server, as well as confidential information that either of these access.
Demonstrative Examples	Debug code can be used to bypass authentication. For example, suppose an application has a login script that receives a username and a password. Assume also that a third, optional, parameter, called "debug", is interpreted by the script as requesting a switch to debug mode, and that when this parameter is given the username and password are not checked. In such a case, it is very simple to bypass the authentication process if the special behavior of the application regarding the debug parameter is known. In a case where the form is: <FORM ACTION="/authenticate_login.cgi"> <INPUT TYPE=TEXT name=username> <INPUT TYPE=PASSWORD name=password> <INPUT TYPE=SUBMIT> </FORM> Then a conforming link will look like: http://TARGET/authenticate_login.cgi?username=...&password=...

	An attacker can change this to: http://TARGET/authenticate_login.cgi? username=&password=&debug=1 Which will grant the attacker access to the site, bypassing the authentication process.
Context Notes	A common development practice is to add "back door" code specifically designed for debugging or testing purposes that is not intended to be shipped or deployed with the application. In web-based applications, debug code is used to test and modify web application properties, configuration information, and functions. If a debug application is left on a production server, an attacker may be able to use it to perform these tasks. When this sort of debug code is left in the application, the application is open to unintended modes of interaction. These back door entry points create security risks because they are not considered during design or testing and fall outside of the expected operating conditions of the application. While it is possible to leave debug code in an application in any language, in J2EE a main method may be a good indicator that debug code has been left in the application, although there may not be any direct security impact.
Node Relationships	ChildOf - Encapsulation (485) ChildOf - Weaknesses Examined by SAMATE (630)
Source Taxonomies	7 Pernicious Kingdoms - Leftover Debug Code
Applicable Platforms	All

Path 1:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
      '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
2449. public static void main(String[] argv) throws Exception //installer.java
      '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
274. public static void main(String args[]) //availabilityreport.java
     '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
475. public static void main(String[] args) //eventsarchiver.java
     '
```

[Table of Contents] [Description]

Path 5:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
998. public static void main(String[] args) //dbifserviceentry.java
     '
```

[Table of Contents] [Description]

Path 6:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
1107. public static void main(String[] args) //dbipinterfaceentry.java
      '
```

[Table of Contents] [Description]

Path 7:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
1055. public static void main(String[] args) //dbsnmpinterfaceentry.java
      '
```

[Table of Contents] [Description]

Path 8:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
74. public static void main(String[] args) //pollcontexttest.java
```

```
    '
```

[Table of Contents] [Description]

Path 9:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
885. public static void main(String[] args) //pollertest.java
    '
```

[Table of Contents] [Description]

Path 10:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
51. public static void main(String[] argv) throws Exception //checknrpe.java
    '
```

[Table of Contents] [Description]

Path 11:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
    '
```

[Table of Contents] [Description]

Path 12:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
117. public static void main(String[] args) //pollablestest.java
    '
```

[Table of Contents] [Description]

Path 13:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
173. public static void main(String[] argv) //interface.java
    '
```

[Table of Contents] [Description]

Path 14:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
173. public static void main(final String[] args) throws
Exception //rrdstresser.java
```

[Table of Contents] [Description]

Path 15:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
52. public static void main(String[] args) //scheduletest.java
```

[Table of Contents] [Description]

Path 16:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
59. public static void main(String[] args) //startservice.java
```

[Table of Contents] [Description]

Path 17:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
383. public static void main(String[] args) throws IOException //execrunner.java
```

[Table of Contents] [Description]

Path 18:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
```

[Table of Contents] [Description]

Path 19:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
171. public static void main(String[] argv) //nessusparser.java
```

```
    '
```

[Table of Contents] [Description]

Path 20:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
11. public static void main(String[] args) //mhtest.java
    '
```

[Table of Contents] [Description]

Path 21:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
245. public static void main(String[] args) //inetechoxchange.java
     '
```

[Table of Contents] [Description]

Path 22:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
424. public static void main(String[] args) //sockettest.java
     '
```

[Table of Contents] [Description]

Path 23:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
120. public static void main(String[] args) //snmpagent.java
     '
```

[Table of Contents] [Description]

Path 24:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
372. public static void main(String[] args) //snmpwalk.java
     '
```

[Table of Contents] [Description]

Path 25:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
510. public static void main(String[] args) //snmpwalkmv.java
     '
```

[Table of Contents] [Description]

Path 26:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
53. public static void main (String args[]) //trapd.java
     '
```

[Table of Contents] [Description]

Path 27:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
97. public static void main(String[] args) throws
java.io.IOException //chinesemenu.java
     '
```

[Table of Contents] [Description]

Path 28:

Query Name - **Leftover_Debug_Code**

Severity - **Information**

```
11. public static void main(String[] args) throws UnknownHostException,
java.io.IOException //tester.java
     '
```

[Table of Contents] [Description]

Missing Catch Block

CWE ID	600
Description	If a Servlet fails to catch all exceptions, it may reveal debugging information that will help an adversary form a plan of attack.
Demonstrative Examples	In the following method a DNS lookup failure will cause the Servlet to throw an exception.

```
protected void doPost (HttpServletRequest req,
HttpServletResponse res)
throws IOException {
  String ip = req.getRemoteAddr();
```

```
  InetAddress addr = InetAddress.getByName(ip);
  ...
  out.println("hello " + addr.getHostName());
}
```

Context Notes When a Servlet throws an exception, the default error response the Servlet container sends back to the user typically includes debugging information. This information is of great value to an attacker. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

Node Relationships ChildOf - Error Handling (388)
ResultsIn - Error Message Information Leaks (209)
Peer - Improper Error Handling (390)

Path 1:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameviewservlet.java
     '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
61. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //finishpollerconfigservlet.java
     '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
```

[Table of Contents] [Description]

Path 5:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addgroupdutyschedulesservlet.java
```

[Table of Contents] [Description]

Path 6:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewgroupservlet.java
```

[Table of Contents] [Description]

Path 7:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletegroupservlet.java
```

[Table of Contents] [Description]

Path 8:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifygroupservlet.java
```

[Table of Contents] [Description]

Path 9:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
57. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //removegroupdutyschedulesservlet.java
```

[Table of Contents] [Description]

Path 10:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renamegroupservlet.java
```

[Table of Contents] [Description]

Path 11:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //savegroupservlet.java
```

[Table of Contents] [Description]

Path 12:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
57. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updategroupservlet.java
```

[Table of Contents] [Description]

Path 13:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
85. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewinterfaceservlet.java
```

[Table of Contents] [Description]

Path 14:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
70. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletegetnodesservlet.java
    66. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 15:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
66. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteinterfaceservlet.java
```

[Table of Contents] [Description]

Path 16:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
```

[Table of Contents] [Description]

Path 17:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
69. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteserviceservlet.java
```

[Table of Contents] [Description]

Path 18:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getinterfacesservlet.java
```

[Table of Contents] [Description]

Path 19:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
80. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //getnodesservlet.java
```

[Table of Contents] [Description]

Path 20:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
```

[Table of Contents] [Description]

Path 21:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
```

[Table of Contents] [Description]

Path 22:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
```

[Table of Contents] [Description]

Path 23:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
76. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpgetinterfacesservlet.java
```

[Table of Contents] [Description]

Path 24:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpgetnodesservlet.java
```

[Table of Contents] [Description]

Path 25:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
```

[Table of Contents] [Description]

Path 26:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
```

[Table of Contents] [Description]

Path 27:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
60. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //servicenoticeupdateservlet.java
```

[Table of Contents] [Description]

Path 28:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updatenotifdstatusservlet.java
```

[Table of Contents] [Description]

Path 29:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
```

[Table of Contents] [Description]

Path 30:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
```

[Table of Contents] [Description]

Path 31:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
64. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //finishpollerconfigservlet.java
```

[Table of Contents] [Description]

Path 32:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
```

[Table of Contents] [Description]

Path 33:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //adddutyschedulesservlet.java
```

[Table of Contents] [Description]

Path 34:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewuserservlet.java
```

[Table of Contents] [Description]

Path 35:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteuserservlet.java
```

[Table of Contents] [Description]

Path 36:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyuserservlet.java
```

[Table of Contents] [Description]

Path 37:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //removedutyschedulesservlet.java
```

[Table of Contents] [Description]

Path 38:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameuserservlet.java
```

[Table of Contents] [Description]

Path 39:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveuserservlet.java
```

[Table of Contents] [Description]

Path 40:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
59. public void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //updateuserservlet.java
     '
```

[Table of Contents] [Description]

Path 41:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewviewservlet.java
     '
```

[Table of Contents] [Description]

Path 42:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteviewservlet.java
     '
```

[Table of Contents] [Description]

Path 43:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyviewservlet.java
     '
```

[Table of Contents] [Description]

Path 44:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameviewservlet.java
     '
```

[Table of Contents] [Description]

Path 45:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveviewservlet.java
     '
```

```
'
```

[Table of Contents] [Description]

Path 46:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateviewservlet.java
'
```

[Table of Contents] [Description]

Path 47:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
'
```

[Table of Contents] [Description]

Path 48:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
91. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //importassetsservlet.java
'
```

[Table of Contents] [Description]

Path 49:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyassetservlet.java
'
```

[Table of Contents] [Description]

Path 50:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
'
```

[Table of Contents] [Description]

Path 51:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
```

[Table of Contents] [Description]

Path 52:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
62. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcdebugservlet.java
```

[Table of Contents] [Description]

Path 53:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcpostservlet.java
```

[Table of Contents] [Description]

Path 54:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
65. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noderescanservlet.java
```

[Table of Contents] [Description]

Path 55:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
```

[Table of Contents] [Description]

Path 56:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
```

[Table of Contents] [Description]

Path 57:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
74. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventfilterservlet.java
```

[Table of Contents] [Description]

Path 58:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
94. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventqueryservlet.java
```

[Table of Contents] [Description]

Path 59:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //copyeventservlet.java
```

[Table of Contents] [Description]

Path 60:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
77. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteeventservlet.java
```

[Table of Contents] [Description]

Path 61:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparamservlet.java
     |
```

[Table of Contents] [Description]

Path 62:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //neweventservlet.java
     |
```

[Table of Contents] [Description]

Path 63:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rearrangeeventsservlet.java
     |
```

[Table of Contents] [Description]

Path 64:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameeventservlet.java
     |
```

[Table of Contents] [Description]

Path 65:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
57. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveeventservlet.java
     |
```

[Table of Contents] [Description]

Path 66:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
56. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateeventservlet.java
     |
```

[Table of Contents] [Description]

Path 67:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //autoactioneditingservlet.java
     |
```

[Table of Contents] [Description]

Path 68:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteautoactionservlet.java
     |
```

[Table of Contents] [Description]

Path 69:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //newautoactionservlet.java
     |
```

[Table of Contents] [Description]

Path 70:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveautoactionsservlet.java
     |
```

[Table of Contents] [Description]

Path 71:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateautoactionsservlet.java
     '
```

[Table of Contents] [Description]

Path 72:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteforwardservlet.java
     '
```

[Table of Contents] [Description]

Path 73:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
56. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //forwardseditingservlet.java
     '
```

[Table of Contents] [Description]

Path 74:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //newforwardservlet.java
     '
```

[Table of Contents] [Description]

Path 75:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveforwardsservlet.java
     '
```

[Table of Contents] [Description]

Path 76:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateforwardsservlet.java
     '
```

[Table of Contents] [Description]

Path 77:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteelementservlet.java
     '
```

[Table of Contents] [Description]

Path 78:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
57. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //maskeditingservlet.java
     '
```

[Table of Contents] [Description]

Path 79:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //newmaskelementservlet.java
     '
```

[Table of Contents] [Description]

Path 80:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
56. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //savemaskelementsservlet.java
     '
```

[Table of Contents] [Description]

Path 81:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //updatemaskelementsservlet.java
     '
```

[Table of Contents] [Description]

Path 82:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteoperactionservlet.java
     '
```

[Table of Contents] [Description]

Path 83:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //newoperactionservlet.java
     '
```

[Table of Contents] [Description]

Path 84:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
56. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //operactioneditingservlet.java
     '
```

[Table of Contents] [Description]

Path 85:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveoperactionsservlet.java
     '
```

[Table of Contents] [Description]

Path 86:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateoperactionsservlet.java
     '
```

```
'
```

[Table of Contents] [Description]

Path 87:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
120. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdadhocgraphservlet.java
     '
```

[Table of Contents] [Description]

Path 88:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
     '
```

[Table of Contents] [Description]

Path 89:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
     |82. String sendto = request.getParameter("sendto");
```

[Table of Contents] [Description]

Path 90:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparentservlet.java
     '
```

[Table of Contents] [Description]

Path 91:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
58. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException //svgservlet.java
     '
```

[Table of Contents] [Description]

Path 92:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException //svgtranscoder.java
```

[Table of Contents] [Description]

Path 93:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
74. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //nodelabelchangeservlet.java
```

[Table of Contents] [Description]

Path 94:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
```

[Table of Contents] [Description]

Path 95:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noticequeryservlet.java
```

[Table of Contents] [Description]

Path 96:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagefilterservlet.java
```

[Table of Contents] [Description]

Path 97:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
62. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //outagequeryservlet.java
```

[Table of Contents] [Description]

Path 98:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
```

[Table of Contents] [Description]

Path 99:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
60. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
```

[Table of Contents] [Description]

Path 100:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
95. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
```

[Table of Contents] [Description]

Path 101:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
66. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
```

[Table of Contents] [Description]

Path 102:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
     |
```

[Table of Contents] [Description]

Path 103:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
64. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
     |
```

[Table of Contents] [Description]

Path 104:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
99. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
     |
```

[Table of Contents] [Description]

Path 105:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
     |
```

[Table of Contents] [Description]

Path 106:

Query Name - **Missing_Catch_Block**

Severity - **Information**

```
75. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //vulnerabilityfilterservlet.java
     |
```

[Table of Contents] [Description]

Missing XML Validation

CWE ID	112
Description	Failure to enable validation when parsing XML gives an attacker the opportunity to supply malicious input.
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Always validate XML input against a known XML Schema or DTD. Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Context Notes	Most successful attacks begin with a violation of the programmer's assumptions. By accepting an XML document without validating it against a DTD or XML schema, the programmer leaves a door open for attackers to provide unexpected, unreasonable, or malicious input. It is not possible for an XML parser to validate all aspects of a document's content; a parser cannot understand the complete semantics of the data. However, a parser can do a complete and thorough job of checking the document's structure and therefore guarantee to the code that processes the document that the content is well-formed.
Node Relationships	ChildOf - Insufficient Input Validation (20)
Source Taxonomies	7 Pernicious Kingdoms - Missing XML Validation
Applicable Platforms	All

Path 1:

Query Name - **Missing_XML_Validation**

Severity - **Information**

```
223. public DocumentFragment adjustColumnWidths (ExpressionContext
context, //table.java
      |...
      406. DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Missing_XML_Validation**

Severity - **Information**

```
158. public DocumentFragment numberLines (ExpressionContext context, //verbatim.java
      |...
      168. DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

Path 3:

Query Name - **Missing_XML_Validation**

Severity

- **Information**

```
511. public DocumentFragment insertCallouts (NodeIterator
areaspecNodeSet, //verbatim.java
      ...
      570. DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Missing_XML_Validation**

Severity - **Information**

```
107. public XMLWriter(String fileName) throws XMLWriteException //xmlwriter.java
      ...
      122. DocumentBuilderFactory docFactory = DocumentBuilderFactory.newInstance();
```

[Table of Contents] [Description]

Overly-Broad Catch Block

CWE ID	396
Description	Catching overly broad exceptions promotes complex error handling code that is more likely to contain security vulnerabilities.
Demonstrative Examples	The following code excerpt handles three types of exceptions in an identical fashion.

```
try { doExchange(); }
  catch (IOException e) {
    logger.error("doExchange failed", e);
  }
  catch (InvocationTargetException e) {
    logger.error("doExchange failed", e);
  }
  catch (SQLException e) {
    logger.error("doExchange failed", e);
}
```

At first blush, it may seem preferable to deal with these exceptions in a single catch block, as follows: try { doExchange(); } catch (Exception e) { logger.error ("doExchange failed", e); } However, if doExchange() is modified to throw a new type of exception that should be handled in some different kind of way, the broad catch block will prevent the compiler from pointing out the situation. Further, the new catch block will now also handle exceptions derived from RuntimeException such as ClassCastException, and NullPointerException, which is not the programmer's intent.

Context Notes	Multiple catch blocks can get ugly and repetitive, but "condensing" catch blocks by catching a high-level class like Exception can obscure exceptions that deserve special treatment or that should not be caught at this point in the program. Catching an overly broad exception essentially defeats the purpose of Java's typed exceptions, and can become particularly dangerous if the program grows and begins to throw new types of exceptions. The new exception types will not receive any attention.
Node Relationships	ChildOf - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	7 Pernicious Kingdoms - Overly-Broad Catch Block
Applicable Platforms	C C++ Java .NET

Path 1:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
106. public class SimpleNode //simplenode.java
       ...
276. catch (Exception e) {
```

[Table of Contents] [Description]

Path 2:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
       ...
50. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 3:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
       ...
96. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 4:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
      |...
      103. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
126. private void parseBox(String line) //imageintrinsics.java
      |...
      134. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
      |...
      199. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 7:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      |...
      52. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 8:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      |...
      96. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 9:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
44. public ImageIntrinsics(ExpressionContext context, String
imageFn) //imageintrinsics.java
      ...
      103. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 10:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
126. private void parseBox(String line) //imageintrinsics.java
      ...
      134. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 11:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      ...
      164. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 12:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
58. public int execute(String className, List arguments) //classexecutor.java
      ...
      67. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 13:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1107. public Map getTypesFromDB() throws SQLException //installer.java
      ...
      1122. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 14:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2172. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 15:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
93. public void testBug1006NoOldTables() throws SQLException //installerdbtest.java
      |...
      100. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 16:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
110. public void testBug1006HasOldTables() throws
SQLException //installerdbtest.java
      |...
      125. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 17:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
319. public void testConstraintOnBogusColumn() throws
Exception //installerdbtest.java
      |...
      341. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 18:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
355. public void doTestBogusConstraint(String constraint, String
errorSubstring) //installerdbtest.java
      |...
```

```
363. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 19:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
375. public void doTestBug931(boolean dropForeignTable, int
badRows, //installerdbtest.java
     ...
389. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 20:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
72. public void tearDown() throws Exception //installerwebapptest.java
     ...
85. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 21:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
103. public void testWebappExistingOpennms() throws
Exception //installerwebapptest.java
     ...
110. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 22:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
123. public void testServerXmlContext() throws Exception //installerwebapptest.java
     ...
144. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 23:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     173. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 24:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     180. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 25:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     187. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 26:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
150. public void run() //snmprequest.java
     |...
     195. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 27:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
192. public void processSnmpMessage(InetAddress agent, int port, SnmpInt32 version,
SnmpOctetString community, int pduType, SnmpPduPacket pdu) throws
SnmpPduEncodingException //snmptrapsession.java
     |...
     198. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 28:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
220. public void processSnmpTrap(InetAddress agent, int port, SnmpOctetString
community, SnmpPduTrap pdu) throws SnmpPduEncodingException //snmptrapsession.java
     ...
223. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 29:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
249. public void processException(Exception e) //snmptrapsession.java
     ...
252. } catch (Exception e1) {
```

[Table of Contents] [Description]

Path 30:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
     ...
300. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 31:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
615. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 32:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
     ...
655. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 33:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      |...
      |170. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 34:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
      |...
      |262. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 35:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
274. public static void main(String args[]) //availabilityreport.java
      |...
      |293. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 36:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
204. private void initDriver() throws Exception //pdfwriter.java
      |...
      |219. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 37:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
231. private XMLReader createParser() throws Exception //pdfwriter.java
      |
```

```
...
240. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 38:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
99. public synchronized void init() //actiond.java
...
128. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 39:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
170. public synchronized void stop() //actiond.java
...
177. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 40:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
...
392. } catch (Exception oe) {
```

[Table of Contents] [Description]

Path 41:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
302. private void archiveEvents() //eventsarchiver.java
...
398. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 42:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1618. private void handleNewSuspect(Event event) throws
```

```
InsufficientInformationException //broadcasteventprocessor.java
      ...
1629. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 43:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
176. public void init() //capsd.java
      ...
299. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 44:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
252. public boolean isProtocolSupported(InetAddress address) //jdbcplugin.java
      ...
261. } catch (Exception exp) {
```

[Table of Contents] [Description]

Path 45:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
289. public boolean isProtocolSupported(InetAddress address, Map
qualifiers) //jdbcplugin.java
      ...
305. } catch (Exception exp) {
```

[Table of Contents] [Description]

Path 46:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1061. private void updateInterfaceInfo(Connection dbc, Date now, DbNodeEntry node,
DbIpInterfaceEntry dbIpIfEntry, IfSnmpCollector snmpc, boolean isNewIpEntry, boolean
isReparented, boolean doesSnmp) throws SQLException //rescanprocessor.java
      ...
1229. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 47:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
2530. private boolean isNodeDeleted(Connection dbc, int nodeId) throws
SQLException //rescanprocessor.java
      ...
2552. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 48:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
3009. private InetAddress updatePrimarySnmpInterface(Connection dbc, DbNodeEntry
dbNodeEntry, Map collectorMap, InetAddress oldPriIf) throws
SQLException //rescanprocessor.java
      ...
3049. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 49:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
288. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 50:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
294. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 51:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
243. private void loadKnownNodes() throws SQLException //scheduler.java
      ...
300. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 52:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
316. void scheduleNode(int nodeId) throws SQLException //scheduler.java
       ...
340. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 53:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
492. public void run() //scheduler.java
       ...
587. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 54:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
       ...
281. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 55:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
163. private DbNodeEntry getExistingNodeEntry(java.sql.Connection dbc, IfCollector
collector) throws SQLException //suspecteventprocessor.java
       ...
308. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 56:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1249. public void run() //suspecteventprocessor.java
       ...
1356. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 57:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
195. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
207. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 58:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
728. protected void configureSNMPHandler(Event event) //broadcasteventprocessor.java
       ...
750. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 59:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
288. private void sendEvent(String uei) //collectableservice.java
       ...
308. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 60:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
       ...
283. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 61:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
234. private void buildServiceIdMap(final Category log) //collectd.java
       ...
291. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 62:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
498. private void scheduleExistingInterfaces() throws SQLException //collectd.java
      |...
      550. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 63:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      514. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 64:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      565. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 65:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
      622. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 66:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
397. public void initialize(NetworkInterface iface, Map
parameters) //snmpcollector.java
      |...
```

```
786. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 67:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
940. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 68:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
892. public int collect(NetworkInterface iface, EventProxy eproxy, Map
parameters) //snmpcollector.java
...
1216. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 69:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
...
1615. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 70:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1585. private boolean testSnmpV2Support(InetAddress addr) //snmpcollector.java
...
1657. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 71:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
        ...
1692. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 72:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1676. private int getInterfaceCount(SnmpPeer peer) //snmpcollector.java
        ...
1742. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 73:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
        ...
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 74:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
72. public boolean performUpdate( //stringdatasource.java
        ...
106. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 75:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
773. public void syncManagementState(Connection conn) throws
SQLException //capsdconfigfactory.java
        ...
1020. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 76:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      |259. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 77:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      |376. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 78:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      |383. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 79:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      |386. } catch (Exception exx) {
```

[Table of Contents] [Description]

Path 80:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
226. public synchronized void init() //discovery.java
      |...
      |396. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 81:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
421. public synchronized void start() //discovery.java
      |...
      |428. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 82:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
421. public synchronized void start() //discovery.java
      |...
      |435. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 83:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
421. public synchronized void start() //discovery.java
      |...
      |438. } catch (Exception exx) {
```

[Table of Contents] [Description]

Path 84:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
      |...
      |460. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 85:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
      |...
      |467. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 86:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
453. public synchronized void stop() //discovery.java
       ...
477. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 87:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
       ...
549. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 88:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
543. public synchronized void stop() //pingmanager.java
       ...
554. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 89:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
646. public void run() //pingmanager.java
       ...
689. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 90:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
646. public void run() //pingmanager.java
       ...
694. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 91:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
834. private static String getNodeLabel(long nodeId) throws
SQLException //eventutil.java
      ...
856. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 92:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
888. private static String getIfAlias(long nodeId, String ipaddr) throws
SQLException //eventutil.java
      ...
911. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 93:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
943. private static String getAssetInfo(long nodeId, String assetField) throws
SQLException //eventutil.java
      ...
965. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 94:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
672. public void persistEvent(Header eventHeader, Event event) throws
SQLException //eventwriter.java
      ...
698. } catch (Exception e2) {
```

[Table of Contents] [Description]

Path 95:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
238. public void run() //tcpserver.java
      ...
265. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 96:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
174. public void run() //tcpstreamhandler.java
       ...
461. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 97:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
64. public MockDatabase() //mockdatabase.java
       ...
67. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 98:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
380. public void testInvalidPoll() throws
UnknownHostException //mocknetworktest.java
       ...
387. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 99:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
159. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
166. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 100:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
185. private void automaticAcknowledge(Event event) //broadcasteventprocessor.java
       ...
201. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 101:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
185. private void automaticAcknowledge(Event event) //broadcasteventprocessor.java
       ...
210. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 102:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
308. private boolean continueWithNotice(Event event) //broadcasteventprocessor.java
       ...
332. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 103:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
349. private boolean autoAckExistsForEvent(String
eventUei) //broadcasteventprocessor.java
       ...
360. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 104:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
       ...
375. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 105:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
       ...
388. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 106:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
      |...
      403. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 107:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
      |...
      430. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 108:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
      |...
      446. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 109:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
368. private void scheduleNoticesForEvent(Event
event) //broadcasteventprocessor.java
      |...
      487. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 110:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
822. public String scheduledOutage(long nodeId, String
theInterface) //broadcasteventprocessor.java
```

```
      ...
849. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 111:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
171. public void processQueue() //defaultqueuehandler.java
      ...
196. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 112:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public int send(List arguments) //growlnotificationstrategy.java
      ...
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 113:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
126. public synchronized void init() //notifd.java
      ...
158. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 114:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
241. public synchronized void stop() //notifd.java
      ...
250. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 115:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
200. public void run() //notificationtask.java
```

```
...
206. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 116:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
288. private String getArgumentValue(String aSwitch) //notificationtask.java
...
307. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 117:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public int send(List arguments) //xmppgroupnotificationstrategy.java
...
87. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 118:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
66. protected XMPPNotificationManager() //xmppnotificationmanager.java
...
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 119:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
66. protected XMPPNotificationManager() //xmppnotificationmanager.java
...
120. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 120:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
142. private void login() //xmppnotificationmanager.java
```

```
...
150. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 121:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public int send(List arguments) //xmppnotificationstrategy.java
...
87. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 122:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
...
66. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 123:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
...
72. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 124:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
...
78. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 125:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
```

```
...
84. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 126:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
...
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 127:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() //notifd.java
...
96. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 128:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
211. public void init() //outagemanager.java
...
249. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 129:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
373. public void stop() //outagemanager.java
...
399. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 130:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
311. public final void run() //replyreceiver.java
```

```
...
324. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 131:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
...
176. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 132:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
149. public void buildServiceNameToIdMaps(Map nameToId, Map
idToName) //defaultquerymanager.java
...
184. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 133:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
222. public int getNodeIDForInterface(String ipaddr) throws
SQLException //defaultquerymanager.java
...
246. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 134:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
271. public String getNodeLabel(int nodeId) throws
SQLException //defaultquerymanager.java
...
294. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 135:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
475. public void openOutage(String outageIdSQL, int nodeId, String ipAddr, int
serviceId, int dbId, String time) //defaultquerymanager.java
      ...
498. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 136:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
513. public void resolveOutage(int nodeId, String ipAddr, int serviceId, int dbId,
String time) //defaultquerymanager.java
      ...
533. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 137:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
544. public void reparentOutages(String ipAddr, int oldNodeId, int
newNodeId) //defaultquerymanager.java
      ...
556. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 138:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
113. public synchronized void init() //poller.java
      ...
133. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 139:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
113. public synchronized void init() //poller.java
      ...
144. } catch (Exception sqlE) {
```

[Table of Contents] [Description]

Path 140:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
353. public void scheduleService(final int nodeId, final String ipAddr, final String
svcName) //poller.java
      |...
      |378. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 141:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
625. private void scheduledOutagesChangeHandler(Category
log) //pollereventprocessor.java
      |...
      |629. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 142:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
      |...
      |219. catch (Exception exp)
```

[Table of Contents] [Description]

Path 143:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      |...
      |292. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 144:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
245. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //snmpmonitor.java
      |...
```

```
346. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 145:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
    ...
108. catch (Exception e) {
```

[Table of Contents] [Description]

Path 146:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
    ...
116. catch (Exception e) {
```

[Table of Contents] [Description]

Path 147:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
    ...
123. catch (Exception e) {
```

[Table of Contents] [Description]

Path 148:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
415. public void close() //nsclientmanager.java
    ...
418. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 149:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
434. private NsclientPacket sendCheckRequest(String request) //nsclientmanager.java
```

```
...
448. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 150:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
319. public InputStream createGraph(String command, File workDir) throws
IOException, org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
...
525. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 151:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
289. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
...
299. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 152:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
332. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
...
343. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 153:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
829. private void processPendingOperations() //queuingrrdstrategy.java
...
852. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 154:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
864. private void processClose(Object rrd) //queuingrrdstrategy.java
        ...
868. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 155:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
214. public void run() //rrdstresser.java
        ...
218. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 156:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
314. public void execute(String[] args) throws Exception //rrdstresser.java
        ...
337. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 157:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
90. public static void initialize() throws RrdException //rrdutils.java
        ...
94. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 158:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
152. public static boolean createRRD(String creator, String directory, String
dsName, int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List
rraList) throws RrdException //rrdutils.java
        ...
181. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 159:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
202. public static void updateRRD(String owner, String repositoryDir, String dsName,
String val) throws RrdException //rrdutils.java
     |...
225. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 160:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
127. private char getServiceStatus(long nodeid, String ip, String
svc) //datamanager.java
     |...
162. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 161:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
196. private void addNodeIpSvcToCategory(long nodeid, String ip, String svcname,
RTCCategory cat, HashMap knownIPs, PreparedStatement
outagesGetStmt) //datamanager.java
     |...
278. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 162:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |...
563. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 163:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
462. private void populateNodesFromDB(Connection dbConn) throws SQLException,
FilterParseException, RTCException //datamanager.java
     |...
592. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 164:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1010. public synchronized void rtcNodeRescan(long nodeid) throws SQLException,
FilterParseException, RTCException //datamanager.java
      ...
1042. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 165:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1069. public synchronized void rtcNodeIpRescan(long nodeid, String ip) throws
SQLException, FilterParseException, RTCException //datamanager.java
      ...
1237. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 166:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1468. public synchronized long getServiceID(String svcname) throws
SQLException //datamanager.java
      ...
1498. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 167:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1468. public synchronized long getServiceID(String svcname) throws
SQLException //datamanager.java
      ...
1503. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 168:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
135. private int setCurrentThreadPriority(int priority) //datasender.java
```

```
...
140. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 169:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
180. public synchronized void start() //datasender.java
...
189. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 170:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
200. public synchronized void stop() //datasender.java
...
208. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 171:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
463. public void init() //rtcmanager.java
...
506. } catch (Exception nfE) {
```

[Table of Contents] [Description]

Path 172:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
463. public void init() //rtcmanager.java
...
536. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 173:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
669. public void stop() //rtcmanager.java
```

```
...
722. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 174:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
174. public void run() //executor.java
...
246. catch (Exception ex) {
```

[Table of Contents] [Description]

Path 175:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
174. public void run() //executor.java
...
259. catch (Exception ex) {
```

[Table of Contents] [Description]

Path 176:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
102. public synchronized void init() //scriptd.java
...
132. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 177:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
173. public synchronized void stop() //scriptd.java
...
180. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 178:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
135. public SnmpTrapHelper() //snmptraphelper.java
```

```
...
145. catch (Exception e) {
```

[Table of Contents] [Description]

Path 179:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
187. public void onEvent(Event event) //broadcasteventprocessor.java
...
199. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 180:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
205. public void initialize(NetworkInterface iface, Map
parameters) //latencythresholder.java
...
275. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 181:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
223. public void initialize(NetworkInterface iface, Map
parameters) //snmpthresholder.java
...
372. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 182:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
...
273. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 183:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
160. public synchronized void init() //threshd.java
       ...
281. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 184:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
529. private void scheduleExistingInterfaces() throws SQLException //threshd.java
       ...
587. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 185:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
288. private void sendEvent(String uei) //thresholdableservice.java
       ...
308. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 186:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
347. public synchronized void init() //trapd.java
       ...
401. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 187:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
383. public static void main(String[] args) throws IOException //execrunner.java
       ...
413. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 188:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public static final boolean isValidRRDNodeDir(File file) //rrdfileconstants.java
      ...
85. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 189:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
166. public void run() //streamgobbler.java
      ...
186. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 190:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
132. public void send(Log eventLog) throws EventProxyException //tcpeventproxy.java
      ...
139. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 191:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
203. public void run() //vacuumd.java
      ...
244. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 192:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
117. public static void start(MBeanServer server) //manager.java
      ...
122. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 193:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
236. public static void stop(MBeanServer server) //manager.java
       ...
241. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 194:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
332. public static List status(MBeanServer server) //manager.java
       ...
337. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 195:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
213. private Set getAllManagedInterfaces() //scheduler.java
       ...
250. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 196:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
213. private Set getAllManagedInterfaces() //scheduler.java
       ...
256. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 197:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
275. void addToKnownAddresses(InetAddress address, int scanLevel) throws
SQLException //scheduler.java
       ...
303. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 198:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
234. public DNSAddressRR readRR() throws IOException //dnsinputstream.java
        ...
257. } catch (Exception ex) {
```

[Table of Contents] [Description]

Path 199:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameviewservlet.java
        ...
60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 200:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
122. public void run() //initializerservletcontextlistener.java
        ...
127. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 201:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
122. public void run() //initializerservletcontextlistener.java
        ...
137. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 202:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
        ...
133. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 203:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
     ...
153. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 204:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
164. public void run() //reportmailer.java
     ...
194. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 205:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
211. protected void generateFile(String shellScript) throws IOException,
InterruptedException //reportmailer.java
     ...
240. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 206:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
183. public synchronized static void destroy(ServletContext context) throws
ServletException //servletinitializer.java
     ...
189. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 207:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
127. public static String getHostname(String ipAddress) //util.java
     ...
132. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 208:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
172. public static String resolveIpAddress(String ipAddress) //util.java
       ...
177. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 209:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
100. public void init() throws ServletException //addpollerconfigservlet.java
       ...
126. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 210:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
137. public void reloadFiles() throws ServletException //addpollerconfigservlet.java
       ...
163. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 211:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
61. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //finishpollerconfigservlet.java
       ...
74. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 212:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
100. public void init() throws ServletException //pollerconfigservlet.java
       ...
122. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 213:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
133. public void reloadFiles() throws ServletException //pollerconfigservlet.java
     ...
155. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 214:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletegroupservlet.java
     ...
59. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 215:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifygroupservlet.java
     ...
63. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 216:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renamegroupservlet.java
     ...
60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 217:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //savegroupservlet.java
     ...
65. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 218:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void init() throws ServletException //addnewinterfaceservlet.java
      ...
79. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 219:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
107. private void createAndSendNewSuspectInterfaceEvent(String ipaddr) throws
ServletException //addnewinterfaceservlet.java
      ...
124. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 220:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void init() throws ServletException //deletegetnodesservlet.java
      ...
66. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 221:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
86. private void sendEvent(Event event) throws
ServletException //deleteinterfaceservlet.java
      ...
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 222:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
79. public void init() throws ServletException //deletenodesservlet.java
      ...
```

```
82. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 223:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
203. private void sendEvent(Event event) throws
ServletException //deletenodesservlet.java
...
207. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 224:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
119. private void sendEvent(Event event) throws
ServletException //deleteserviceservlet.java
...
123. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 225:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
68. public void init() throws ServletException //getinterfacesservlet.java
...
71. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 226:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public void init() throws ServletException //getnodesservlet.java
...
76. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 227:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
91. public void init() throws ServletException //managenodeservlet.java
      |
      ...
      94. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 228:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
91. public void init() throws ServletException //managenodeservlet.java
      |
      ...
      100. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 229:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
332. private void sendEvent(Event event) throws
ServletException //managenodeservlet.java
      |
      ...
      336. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 230:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
91. public void init() throws ServletException //managenodesservlet.java
      |
      ...
      94. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 231:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
91. public void init() throws ServletException //managenodesservlet.java
      |
      ...
      100. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 232:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
332. private void sendEvent(Event event) throws
ServletException //managenodesservlet.java
      |
      ...
      336. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 233:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
      |
      ...
      153. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 234:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
69. public void init() throws ServletException //snmpgetinterfacesservlet.java
      |
      ...
      72. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 235:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
70. public void init() throws ServletException //snmpgetnodesservlet.java
      |
      ...
      73. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 236:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public void init() throws ServletException //snmpmanagenodesservlet.java
      |
      ...
      76. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 237:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public void init() throws ServletException //snmpmanagenodesservlet.java
      |
      ...
      82. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 238:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
174. private void sendEvent(Event event) throws
ServletException //snmpmanagenodesservlet.java
      |
      ...
      178. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 239:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      |
      ...
      96. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 240:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      |
      ...
      103. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 241:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      |
      ...
      148. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 242:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
173. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 243:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
263. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 244:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
      ...
284. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 245:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updatenotifdstatusservlet.java
      ...
60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 246:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
      ...
```

```
91. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 247:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
100. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 248:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
123. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 249:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
247. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 250:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
101. public void init() throws ServletException //addpollerconfigservlet.java
    ...
127. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 251:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
138. public void reloadFiles() throws ServletException //addpollerconfigservlet.java
     ...
164. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 252:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
64. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //finishpollerconfigservlet.java
     ...
77. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 253:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
99. public void init() throws ServletException //pollerconfigservlet.java
     ...
121. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 254:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
132. public void reloadFiles() throws ServletException //pollerconfigservlet.java
     ...
154. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 255:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewuserservlet.java
     ...
57. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 256:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addnewuserservlet.java
    ...
68. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 257:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteuserservlet.java
    ...
59. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 258:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyuserservlet.java
    ...
63. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 259:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameuserservlet.java
    ...
59. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 260:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveuserservlet.java
    ...
64. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 261:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
59. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updateuserservlet.java
      ...
67. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 262:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deleteviewservlet.java
      ...
58. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 263:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyviewservlet.java
      ...
63. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 264:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
51. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameviewservlet.java
      ...
60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 265:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //saveviewservlet.java
```

```
      ...
64. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 266:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
155. private void sendEvent(Event event) throws
ServletException //modifyassetservlet.java
      ...
159. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 267:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
      ...
234. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 268:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      ...
106. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 269:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
86. public CategoryList(ServletContext context) throws
ServletException //categorylist.java
      ...
91. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 270:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
86. public CategoryList(ServletContext context) throws
ServletException //categorylist.java
      ...
108. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 271:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
57. public void init() throws ServletException //noderescanservlet.java
      ...
60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 272:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
65. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noderescanservlet.java
      ...
93. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 273:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
153. public List getEvents() //eventconffactory.java
      ...
157. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 274:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
168. public Map getEventsMap() //eventconffactory.java
      ...
172. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 275:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
189. public boolean eventHasNotice(String uei, String
notice) //eventconffactory.java
      ...
193. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 276:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
211. public Event getEvent(String uei) //eventconffactory.java
      ...
215. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 277:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
233. public List getEventUEIs() //eventconffactory.java
      ...
237. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 278:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
254. public synchronized void saveEvents(Collection eventsList) throws
XMLWriteException //eventconffactory.java
      ...
261. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 279:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparamservlet.java
      ...
61. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 280:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rearrangeeventsservlet.java
      ...
64. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 281:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //renameeventservlet.java
      ...
61. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 282:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
      ...
208. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 283:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
      ...
254. } catch (Exception e1) {
```

[Table of Contents] [Description]

Path 284:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
453. private Element createHostElement(MapNode n, boolean
loadIcons) //documentgenerator.java
```

```
...
472. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 285:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
82. public Vector getOpenNMSNodes() //mapnodefactory.java
...
95. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 286:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
82. public Vector getOpenNMSNodes() //mapnodefactory.java
...
134. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 287:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
82. public Vector getOpenNMSNodes() //mapnodefactory.java
...
167. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 288:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
66. public void init() throws ServletException //nodelabelchangeservlet.java
...
69. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 289:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
316. public synchronized void parse(InputSource inpSource) throws
```

```
IOException //bbparser.java
      ...
      326. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 290:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
72. public static void makeBackup(String original, String fileName) throws
IOException //filebackup.java
      ...
      119. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 291:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public static void backup(String original, String fileName) throws
XMLWriteException //xmlbackup.java
      ...
      120. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 292:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
147. private String getHost() //xmlheader.java
      ...
      153. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 293:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
107. public XMLWriter(String fileName) throws XMLWriteException //xmlwriter.java
      ...
      126. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 294:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
165. public String getVersion() //xmlwriter.java
       ...
176. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 295:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
199. protected void serializeToFile() throws XMLWriteException //xmlwriter.java
       ...
219. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 296:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void init() throws ServletException //addinterfacetourlservlet.java
       ...
63. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 297:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //addreportstourlservlet.java
       ...
55. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 298:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
58. public void init() throws ServletException //rrdlistservlet.java
       ...
61. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 299:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
58. public void init() throws ServletException //addinterfacetourlservlet.java
      |
      ...
      |
68. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 300:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
      |
      ...
      |
90. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 301:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
56. public void init() throws ServletException //addreportstourlservlet.java
      |
      ...
      |
59. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 302:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
62. public void init() throws ServletException //rrdlistservlet.java
      |
      ...
      |
65. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 303:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //error.jsp
      |
      ...
      |
51. catch(Exception e)
```

[Table of Contents] [Description]

Path 304:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //index.jsp
      ...
      180. } catch (Exception e) { /*if factory can't be initialized, status is
      already 'Unknown'*/ }
```

[Table of Contents] [Description]

Path 305:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
50. private void sendSNMPRestartEvent(int nodeid, String primeInt) throws
ServletException //updatesnmp.jsp
      ...
      60. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 306:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
180. private String getEventsList() //rearrangeevents.jsp
      ...
      187. } catch (Exception e ) {
```

[Table of Contents] [Description]

Path 307:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //assignnotices.jsp
      ...
      49. catch( Exception e ) { throw new ServletException(e); }
```

[Table of Contents] [Description]

Path 308:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //choosecommands.jsp
      ...
      52. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 309:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
157. public String buildCommands(Path path, int index) //choosecommands.jsp
       ...
183. } catch (Exception e)
```

[Table of Contents] [Description]

Path 310:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
192. public String buildCommandSelect(Path path, int index, String
name) //choosecommands.jsp
       ...
223. } catch (Exception e)
```

[Table of Contents] [Description]

Path 311:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //choosetargets.jsp
       ...
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 312:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
278. public Map getUsers(Collection targets) //choosetargets.jsp
       ...
309. } catch (Exception e)
```

[Table of Contents] [Description]

Path 313:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
317. public Map getGroups(Collection targets) //choosetargets.jsp
       ...
339. } catch (Exception e)
```

[Table of Contents] [Description]

Path 314:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
347. public Map getEmails(Collection targets) //choosetargets.jsp
       ...
364. } catch (Exception e)
```

[Table of Contents] [Description]

Path 315:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //destinationpaths.jsp
       ...
49. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 316:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
46. public void init() throws ServletException //escalations.jsp
       ...
53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 317:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
46. public void init() throws ServletException //groupintervals.jsp
       ...
52. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 318:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
46. public void init() throws ServletException //pathoutline.jsp
       ...
50. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 319:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //buildrule.jsp
      ...
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 320:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //choosepath.jsp
      ...
49. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 321:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
220. public String buildPathSelect(String currentPath) //choosepath.jsp
      ...
242. } catch (Exception e)
```

[Table of Contents] [Description]

Path 322:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //chooseueis.jsp
      ...
49. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 323:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //eventnotices.jsp
      ...
50. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 324:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
46. public void init() throws ServletException //validaterule.jsp
      |
      ...
      |
49. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 325:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //addpollerconfig.jsp
      |
      ...
      |
111. catch(Exception e)
```

[Table of Contents] [Description]

Path 326:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //index.jsp
      |
      ...
      |
111. catch(Exception e)
```

[Table of Contents] [Description]

Path 327:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
18. public void sendOutagesChangedEvent() throws ServletException //editoutage.jsp
      |
      ...
      |
31. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 328:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
13. public void sendOutagesChangedEvent() throws ServletException //index.jsp
      |
      ...
      |
26. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 329:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //groupdetail.jsp
       ...
52. catch (Exception e)
```

[Table of Contents] [Description]

Path 330:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //list.jsp
       ...
53. catch(Exception e)
```

[Table of Contents] [Description]

Path 331:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
433. private String getAllUsersMinusInGroup(Group group) //modifygroup.jsp
       ...
461. catch(Exception e)
```

[Table of Contents] [Description]

Path 332:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //list.jsp
       ...
53. catch(Exception e)
```

[Table of Contents] [Description]

Path 333:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //modifyuser.jsp
       ...
55. catch(Exception e)
```

[Table of Contents] [Description]

Path 334:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //userdetail.jsp
      |
      ...
52. catch (Exception e)
```

[Table of Contents] [Description]

Path 335:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //list.jsp
      |
      ...
53. catch(Exception e)
```

[Table of Contents] [Description]

Path 336:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
381. private String getAllUsersMinusInView(View view) //modifyview.jsp
      |
      ...
424. catch(Exception e)
```

[Table of Contents] [Description]

Path 337:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
463. private String getAllGroupsMinusInView(View view) //modifyview.jsp
      |
      ...
507. catch(Exception e)
```

[Table of Contents] [Description]

Path 338:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //viewdetail.jsp
      |
      ...
52. catch (Exception e)
```

[Table of Contents] [Description]

Path 339:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //index.jsp
       ...
54. catch(Exception e)
```

[Table of Contents] [Description]

Path 340:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //reportgenerating.jsp
       ...
52. catch(Exception e) {
```

[Table of Contents] [Description]

Path 341:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //interface.jsp
       ...
56. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 342:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //interface.jsp
       ...
63. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 343:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //interface.jsp
       ...
70. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 344:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //interface.jsp
      ...
77. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 345:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //interface.jsp
      ...
84. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 346:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      ...
64. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 347:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      ...
71. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 348:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      ...
78. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 349:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      |...
      |85. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 350:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      |...
      |92. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 351:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      |...
      |99. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 352:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
55. public void init() throws ServletException //node.jsp
      |...
      |106. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 353:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
19. public void init() throws ServletException //header.jsp
      |...
      |23. catch( Exception e ) {/*notice status will be unknown if the factory can't
      |be initialized*/}
```

[Table of Contents] [Description]

Path 354:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <%-- //header.jsp
      ...
      58. } catch (Exception e) { /*if factory can't be initialized, status is
      already 'Unknown'*/ }
```

[Table of Contents] [Description]

Path 355:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      ...
      48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 356:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      ...
      58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 357:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      ...
      69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 358:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //ksc-box.jsp
      ...
      71. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 359:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
50. public void init() throws ServletException //performance-box.jsp
      ...
54. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 360:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
54. public void init() throws ServletException //response-box.jsp
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 361:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void init() throws ServletException //custom_graph1.jsp
      ...
49. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 362:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
44. public void init() throws ServletException //custom_graph2.jsp
      ...
48. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 363:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 364:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      |
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 365:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      |
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 366:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      |
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 367:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      |
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 368:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      |
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 369:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 370:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 371:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 372:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public void init() throws ServletException //include_init1.jsp
      ...
47. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 373:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public void init() throws ServletException //include_init1.jsp
      ...
47. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 374:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public void init() throws ServletException //include_init1.jsp
      |
      ...
47. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 375:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
45. public void saveFactory() throws ServletException //form_proc_report.jsp
      |
      ...
50. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 376:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      |
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 377:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      |
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 378:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      |
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 379:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
42. public void init() throws ServletException //include_init1.jsp
      |
      ...
47. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 380:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      |
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 381:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      |
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 382:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      |
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 383:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
43. public void init() throws ServletException //include_init2.jsp
      |
      ...
48. catch (Exception e) {
```

[Table of Contents] [Description]

Path 384:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public void initPerfModel() throws ServletException //include_init2.jsp
      |
      ...
58. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 385:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
63. public void initReportFactory() throws ServletException //include_init2.jsp
      |
      ...
69. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 386:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
1. <!-- //index.jsp
      |
      ...
60. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 387:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
57. public String getBgColor(Notification n) //browser.jsp
      |
      ...
61. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 388:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //adhoc.jsp
      |
      ...
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 389:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //adhoc2.jsp
      |
      ...
      |
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 390:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //chooseinterface.jsp
      |
      ...
      |
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 391:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
48. public void init() throws ServletException //choosereportanddate.jsp
      |
      ...
      |
52. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 392:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
47. public void init() throws ServletException //index.jsp
      |
      ...
      |
51. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 393:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
50. public void init() throws ServletException //results.jsp
      |
      ...
      |
54. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 394:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
50. public void init() throws ServletException //zoom.jsp
      ...
54. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 395:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() throws ServletException //adhoc.jsp
      ...
53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 396:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() throws ServletException //adhoc2.jsp
      ...
53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 397:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() throws ServletException //chooseinterface.jsp
      ...
53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 398:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() throws ServletException //choosereportanddate.jsp
      ...
53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 399:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
49. public void init() throws ServletException //index.jsp
      |
      ...
      53. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 400:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
52. public void init() throws ServletException //results.jsp
      |
      ...
      56. catch( Exception e ) {
```

[Table of Contents] [Description]

Path 401:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
245. public static void main(String[] args) //inetechoxchange.java
       |
       ...
       275. catch(Exception e)
```

[Table of Contents] [Description]

Path 402:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
89. public snmpagent(String managerAddress, int managerPort, //snmpagent.java
      |
      ...
      108. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 403:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
138. public void sendV1Trap(String message) //snmpagent.java
       |
       ...
       158. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 404:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
169. public void sendV2Trap(String message) //snmpagent.java
       ...
185. } catch (Exception e) {
```

[Table of Contents] [Description]

Path 405:

Query Name - **Overly_Broad_Catch**

Severity - **Information**

```
53. public static void main (String args[]) //trapd.java
       ...
66. catch (Exception e)
```

[Table of Contents] [Description]

Overly-Broad Throws Declaration

CWE ID	397
Description	Throwing overly broad exceptions promotes complex error handling code that is more likely to contain security vulnerabilities.

Demonstrative Examples	The following method throws three types of exceptions.
	
	public void doExchange() throws IOException, InvocationTargetException, SQLException {
	  ...
	}
	
	While it might seem tidier to write public void doExchange() throws Exception { ... } doing so hampers the caller's ability to understand and handle the exceptions that occur. Further, if a later revision of doExchange() introduces a new type of exception that should be treated differently than previous exceptions, there is no easy way to enforce this requirement.

Context Notes	Declaring a method to throw Exception or Throwable makes it difficult for callers to do good error handling and error recovery. Java's exception mechanism is set up to make it easy for callers to anticipate what can go wrong and write code to handle each specific exceptional circumstance. Declaring that a method throws a generic form of exception defeats this system.
Node Relationships	ChildOf - Error Conditions, Return Values, Status Codes (389)
Source Taxonomies	7 Pernicious Kingdoms - Overly-Broad Throws Declaration
Applicable Platforms	C

C++

Java

.NET

Path 1:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
31. public void testNothing() throws Exception //javamailertest.java
      '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
159. public void parse(String column) throws Exception //column.java
      '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
193. public void parseColumnType(String col_type) throws Exception //column.java
      '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
215. public int getColumnSqlType() throws Exception //column.java
      '
```

[Table of Contents] [Description]

Path 5:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
247. public static String normalizeColumnType(String column, boolean hasSize) throws
Exception //column.java
      '
```

[Table of Contents] [Description]

Path 6:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
279. public int columnTypeSize(String type) throws Exception //column.java
```

[Table of Contents] [Description]

Path 7:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
121. public void setForeignDelType(String fdeltype) throws
Exception //constraint.java
```

[Table of Contents] [Description]

Path 8:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
163. public void parse(String constraint) throws Exception //constraint.java
```

[Table of Contents] [Description]

Path 9:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
177. public void install(String[] argv) throws Exception //installer.java
...
457. public String getNextArg(String[] argv, int i, char letter)
```

[Table of Contents] [Description]

Path 10:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
308. public void loadProperties() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 11:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
338. public String fetchProperty(String property) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 12:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
349. public void parseArguments(String[] argv) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 13:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
457. public String getNextArg(String[] argv, int i, char letter) //installer.java
```

[Table of Contents] [Description]

Path 14:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
480. public void readTables() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 15:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
484. public void readTables(Reader reader) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 16:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
611. public void databaseConnect(String database) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 17:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
617. public void databaseDisconnect() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 18:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 19:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
661. public void databaseCheckLanguage() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 20:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
719. public void checkOldTables() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 21:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
750. public String[][] getForeignKeyConstraints() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 22:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 23:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
831. public String getForeignConstraintWhere(String table, String
column, //installer.java
     |
```

[Table of Contents] [Description]

Path 24:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
843. public void fixConstraint() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 25:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
938. public void databaseAddDB() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 26:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
     |
```

[Table of Contents] [Description]

Path 27:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
      |
```

[Table of Contents] [Description]

Path 28:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 29:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1165. public void fixData() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 30:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1178. void insertData() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 31:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 32:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 33:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1449. public boolean functionExists(String function, String
columns, //installer.java
```

[Table of Contents] [Description]

Path 34:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1469. public boolean functionExists(String function, int[]
columnTypes, //installer.java
```

[Table of Contents] [Description]

Path 35:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1487. public void checkWebappOldOpennmsDir() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 36:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1501. public void checkServerXmlOldOpennmsContext() throws
Exception //installer.java
```

[Table of Contents] [Description]

Path 37:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1535. public void installWebApp() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 38:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1556. public void installLink(String source, String destination, //installer.java
```

[Table of Contents] [Description]

Path 39:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1587. public void updateTomcatConf() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 40:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 41:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
```

[Table of Contents] [Description]

Path 42:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1740. public String getXFromSQL(String item, String regex, int
itemGroup, //installer.java
```

[Table of Contents] [Description]

Path 43:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1756. public String getTableFromSQL(String table) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 44:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1761. public String getIndexFromSQL(String index) throws Exception //installer.java
```

[Table of Contents] [Description]

Path 45:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1766. public String getFunctionFromSQL(String function) throws
Exception //installer.java
```

[Table of Contents] [Description]

Path 46:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1772. public String getLanguageFromSQL(String language) throws
Exception //installer.java
```

[Table of Contents] [Description]

Path 47:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1777. public List getTableColumnsFromSQL(String table) throws
Exception //installer.java
```

[Table of Contents] [Description]

Path 48:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1849. public boolean tableColumnExists(String table, String column) //installer.java
```

[Table of Contents] [Description]

Path 49:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1854. public List getTableColumnsFromDB(String table) throws
Exception //installer.java
     '
```

[Table of Contents] [Description]

Path 50:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
     '
```

[Table of Contents] [Description]

Path 51:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
2449. public static void main(String[] argv) throws Exception //installer.java
     '
```

[Table of Contents] [Description]

Path 52:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
25. protected void setUp() throws Exception //installerdbtest.java
     '
```

[Table of Contents] [Description]

Path 53:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
56. public void tearDown() throws Exception //installerdbtest.java
     '
```

[Table of Contents] [Description]

Path 54:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
178. public void testBug931ConstraintsOkayTwoTables() throws
Exception //installerdbtest.java
     '
```

[Table of Contents] [Description]

Path 55:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
186. public void testBug931ConstraintsOkayOneTable() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 56:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
194. public void testBug931ConstraintsBadTwoTables() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 57:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
202. public void testBug931ConstraintsBadOneTable() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 58:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
210. public void testConstraintsFixedNullTwoTables() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 59:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
218. public void testConstraintsFixedNullOneTable() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 60:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
226. public void testConstraintsFixedDelTwoTables() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 61:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
235. public void testConstraintsFixedDelOneTable() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 62:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
244. public void testBogusConstraintName() throws Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 63:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
254. public void testBogusConstraintTable() throws Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 64:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
264. public void testBogusConstraintColumn() throws Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 65:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
275. public void testConstraintAfterConstrainedColumn() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 66:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
297. public void testConstraintAtEndOfTable() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 67:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
319. public void testConstraintOnBogusColumn() throws
Exception //installerdbtest.java
```

[Table of Contents] [Description]

Path 68:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
355. public void doTestBogusConstraint(String constraint, String
errorSubstring) //installerdbtest.java
```

[Table of Contents] [Description]

Path 69:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
375. public void doTestBug931(boolean dropForeignTable, int
badRows, //installerdbtest.java
```

[Table of Contents] [Description]

Path 70:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
72. public void tearDown() throws Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 71:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
98. public void testWebappInstall() throws Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 72:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
103. public void testWebappExistingOpennms() throws
Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 73:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
119. public void testWebappNoOpennms() throws Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 74:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
123. public void testServerXmlContext() throws Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 75:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
153. public void testServerXmlNoContext() throws
Exception //installerwebapptest.java
     |
```

[Table of Contents] [Description]

Path 76:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
157. public void testServerXmlNoFile() throws Exception //installerwebapptest.java
```

[Table of Contents] [Description]

Path 77:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
```

[Table of Contents] [Description]

Path 78:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
```

[Table of Contents] [Description]

Path 79:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
303. public static void generateReport(String logourl, String categoryName, String
format, String monthFormat, String startMonth, String startDate, String startYear)
throws Exception //availabilityreport.java
```

[Table of Contents] [Description]

Path 80:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
60. protected void setUp() throws Exception //availabilityreporttest.java
```

[Table of Contents] [Description]

Path 81:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
294. protected void tearDown() throws Exception //availabilityreporttest.java
     |
```

[Table of Contents] [Description]

Path 82:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
14. protected void setUp() throws Exception //calendartablebuildertest.java
     |
```

[Table of Contents] [Description]

Path 83:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
18. protected void tearDown() throws Exception //calendartablebuildertest.java
     |
```

[Table of Contents] [Description]

Path 84:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
204. private void initDriver() throws Exception //pdfwriter.java
     |
```

[Table of Contents] [Description]

Path 85:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
231. private XMLReader createParser() throws Exception //pdfwriter.java
     |
```

[Table of Contents] [Description]

Path 86:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
180. protected abstract boolean checkProtocol(Socket socket, ConnectionConfig
```

```
config) throws Exception; //abstracttcpplugin.java
     |
```

[Table of Contents] [Description]

Path 87:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
302. protected Socket wrapSocket(Socket socket, ConnectionConfig config) throws
Exception //abstracttcpplugin.java
     |
```

[Table of Contents] [Description]

Path 88:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
118. protected Socket wrapSocket(Socket socket, ConnectionConfig config) throws
Exception //httpsplugin.java
     |
```

[Table of Contents] [Description]

Path 89:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
29. protected void setUp() throws Exception //eventconffactorytest.java
     |
```

[Table of Contents] [Description]

Path 90:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
38. protected void tearDown() throws Exception //eventconffactorytest.java
     |
```

[Table of Contents] [Description]

Path 91:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
180. public synchronized void saveGroups() throws Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 92:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
294. public synchronized void saveGroup(String name, Group details) throws
Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 93:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
308. public synchronized void deleteUser(String name) throws
Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 94:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
330. public synchronized void deleteGroup(String name) throws
Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 95:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
350. public synchronized void renameGroup(String oldName, String newName) throws
Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 96:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
369. public synchronized void renameUser(String oldName, String newName) throws
Exception //groupmanager.java
     |
```

[Table of Contents] [Description]

Path 97:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
837. public Map rebuildParamterMap(int notifId, final String resolutionPrefix)
throws Exception //notificationmanager.java
```

[Table of Contents] [Description]

Path 98:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
236. public void update() throws Exception; //pollerconfig.java
```

[Table of Contents] [Description]

Path 99:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
78. public abstract void update() throws Exception; //polloutagesconfig.java
```

[Table of Contents] [Description]

Path 100:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
119. protected void setUp() throws Exception //snmpeventinfotest.java
```

[Table of Contents] [Description]

Path 101:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
110. protected void setUp() throws Exception //snmppeerfactorytest.java
```

[Table of Contents] [Description]

Path 102:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
107. public synchronized void saveUser(String name, User details) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 103:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
463. public synchronized void saveUsers(Collection usersList) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 104:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
479. public synchronized void deleteUser(String name) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 105:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
500. private synchronized void saveCurrent() throws Exception //usermanager.java
```

[Table of Contents] [Description]

Path 106:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
538. public synchronized void renameUser(String oldName, String newName) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 107:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
573. public void setEncryptedPassword(String userID, String aPassword) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 108:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
590. public void setUnencryptedPassword(String userID, String aPassword) throws
Exception //usermanager.java
```

[Table of Contents] [Description]

Path 109:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
152. public synchronized void saveView(String name, View details) throws
Exception //viewfactory.java
```

[Table of Contents] [Description]

Path 110:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
170. public synchronized void deleteUser(String name) throws
Exception //viewfactory.java
```

[Table of Contents] [Description]

Path 111:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
215. public synchronized void renameUser(String oldName, String newName) throws
Exception //viewfactory.java
```

[Table of Contents] [Description]

Path 112:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
304. public synchronized void renameView(String oldName, String newName) throws
Exception //viewfactory.java
```

[Table of Contents] [Description]

Path 113:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
332. public synchronized void deleteView(String name) throws
Exception //viewfactory.java
```

[Table of Contents] [Description]

Path 114:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
223. public static synchronized void saveCurrent() throws
Exception //vulnscandconfigfactory.java
```

[Table of Contents] [Description]

Path 115:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
48. protected void setUp() throws Exception //mockcategoryfactorytest.java
```

[Table of Contents] [Description]

Path 116:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
55. protected void tearDown() throws Exception //mockcategoryfactorytest.java
```

[Table of Contents] [Description]

Path 117:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
50. protected void setUp() throws Exception //mockdatabasetest.java
```

[Table of Contents] [Description]

Path 118:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
74. protected void tearDown() throws Exception //mockdatabasetest.java
      '
```

[Table of Contents] [Description]

Path 119:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
45. protected void setUp() throws Exception //mocklogappendertest.java
      '
```

[Table of Contents] [Description]

Path 120:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
50. protected void tearDown() throws Exception //mocklogappendertest.java
      '
```

[Table of Contents] [Description]

Path 121:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
189. protected void setUp() throws Exception //mocknetworktest.java
      '
```

[Table of Contents] [Description]

Path 122:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
219. protected void tearDown() throws Exception //mocknetworktest.java
      '
```

[Table of Contents] [Description]

Path 123:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
470. public void testQueryManager() throws Exception //mocknetworktest.java
```

[Table of Contents] [Description]

Path 124:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
578. public void testSetPollStatus() throws Exception //mocknetworktest.java
```

[Table of Contents] [Description]

Path 125:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
215. private void sendResolvedNotifications(Collection notifIDs, Event event, String
acknowledge, String[] match, String resolutionPrefix) throws
Exception //broadcasteventprocessor.java
```

[Table of Contents] [Description]

Path 126:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
266. protected void sendResolvedNotificationsToUser(String queueID, String
targetName, String[] commands, Map params) throws
Exception //broadcasteventprocessor.java
```

[Table of Contents] [Description]

Path 127:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
804. public Map rebuildParameterMap(int notifId, String resolutionPrefix) throws
Exception //broadcasteventprocessor.java
```

[Table of Contents] [Description]

Path 128:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
213. public void notifyGrowlOf( String inNotificationName, NSData
inIconData, //growl.java
```

```
'
```

[Table of Contents] [Description]

Path 129:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
254. public void notifyGrowlOf( String inNotificationName, String
inTitle, //growl.java
'
```

[Table of Contents] [Description]

Path 130:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
276. public void notifyGrowlOf( String inNotificationName, NSImage
inImage, //growl.java
'
```

[Table of Contents] [Description]

Path 131:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
299. public void notifyGrowlOf( String inNotificationName, String
inImagePath, //growl.java
'
```

[Table of Contents] [Description]

Path 132:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
376. public void setDefaultNotifications( NSArray inDefNotes ) throws
Exception //growl.java
'
```

[Table of Contents] [Description]

Path 133:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
398. public void setDefaultNotifications( String [] inDefNotes ) throws
Exception //growl.java
'
```

[Table of Contents] [Description]

Path 134:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
108. private String[] parseArguments(List arguments) throws
Exception //growlnotificationstrategy.java
```

[Table of Contents] [Description]

Path 135:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
374. protected void setUp() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 136:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
443. protected void tearDown() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 137:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
457. public void testWicktorBug_1022_1031() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 138:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
484. public void testNotifdStatus() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 139:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
499. public void testMockNotificationBasic() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 140:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
524. public void testMockNotificationInitialDelay() throws
Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 141:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
539. public void testInterval() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 142:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
556. public void testEscalate() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 143:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
571. public void testManualAcknowledge1() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 144:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
586. public void testManualAcknowledge2() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 145:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
605. public void testAutoAcknowledge1() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 146:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
625. public void testAutoAcknowledge2() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 147:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
652. public void testBug731() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 148:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
672. public void testBug1114() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 149:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
693. public void testRebuildParameterMap() throws Exception //notifdtest.java
```

[Table of Contents] [Description]

Path 150:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
727. public void testGetUsersNotified() throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 151:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
769. private long anticipateNotificationsForGroup(String subject, String textMsg,
String groupName, Date startTime, long interval) throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 152:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
773. private long anticipateNotificationsForGroup(String subject, String textMsg,
String groupName, long startTime, long interval) throws Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 153:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
791. private Collection getUsersInGroup(String groupName) throws
Exception //notifdtest.java
     |
```

[Table of Contents] [Description]

Path 154:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
105. private String[] parseArguments(List arguments) throws
Exception //xmppgroupnotificationstrategy.java
     |
```

[Table of Contents] [Description]

Path 155:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
105. private String[] parseArguments(List arguments) throws
Exception //xmppnotificationstrategy.java
```

[Table of Contents] [Description]

Path 156:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
66. protected void setUp() throws Exception //outagetest.java
```

[Table of Contents] [Description]

Path 157:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
110. protected void tearDown() throws Exception //outagetest.java
```

[Table of Contents] [Description]

Path 158:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
160. public void testNodeLostRegainedService() throws Exception //outagetest.java
```

[Table of Contents] [Description]

Path 159:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
81. protected void setUp() throws Exception //pollcontexttest.java
```

[Table of Contents] [Description]

Path 160:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
154. protected void tearDown() throws Exception //pollcontexttest.java
```

[Table of Contents] [Description]

Path 161:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
199. public void testCreateEvent() throws Exception //pollcontexttest.java
```

[Table of Contents] [Description]

Path 162:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
224. public void testOpenResolveOutage() throws Exception //pollcontexttest.java
```

[Table of Contents] [Description]

Path 163:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
335. private void scheduleExistingServices() throws Exception //poller.java
```

[Table of Contents] [Description]

Path 164:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
428. public void testNodeOutageProcessingDisabled() throws
Exception //pollertest.java
```

[Table of Contents] [Description]

Path 165:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
454. public void testNodeOutageProcessingEnabled() throws
Exception //pollertest.java
```

[Table of Contents] [Description]

Path 166:

Query Name - **Overly_Broad_Throws**

Severity

- **Information**

```
484. public void testNodeLostRegainedService() throws Exception //pollertest.java
     |
```

[Table of Contents] [Description]

Path 167:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
526. public void testPolling() throws Exception //pollertest.java
     |
```

[Table of Contents] [Description]

Path 168:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
63. public void testMeetsCriteriaWithIPAddr() throws
Exception //snmpmonitortest.java
     |
```

[Table of Contents] [Description]

Path 169:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
18. public static NrpePacket executeQuery(String host, String buffer) throws
Exception //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 170:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
23. public static NrpePacket executeQuery(String host, String buffer, int padding)
throws Exception //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 171:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
27. public static NrpePacket executeQuery(String host, int port, String
```

```
buffer, //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 172:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
39. public static NrpePacket sendPacket(short type, short resultCode, String buffer)
throws Exception //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 173:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
51. public static void main(String[] argv) throws Exception //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 174:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
102. public static String nextArg(String[] argv, int i) throws
Exception //checknrpe.java
     |
```

[Table of Contents] [Description]

Path 175:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
151. protected void setUp() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 176:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
365. protected void tearDown() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 177:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
401. public void testCreateService() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 178:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
523. public void testDeleteInterface() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 179:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
542. public void testDeleteNode() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 180:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
700. public void testStatus() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 181:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
748. public void testInterfaceStatus() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 182:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
771. public void testFindMemberWithDescendent() throws
Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 183:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
784. public void testPropagateUnresponsive() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 184:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
886. public void testPollService() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 185:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
910. public void testPollAllUp() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 186:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
923. public void testPollIfUpNonCritSvcDown() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 187:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
940. public void testPollIfUpCritSvcDownPoll() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 188:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
958. public void testPollIfDownNonCritSvcUp() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 189:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
985. public void testPollIfDownCritSvcUp() throws Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 190:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1016. public void testPollIfUpCritSvcUndefSvcDown() throws
Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 191:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1034. public void testPollIfDownCritSvcUndefSvcDown() throws
Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 192:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1067. public void testPollIfUpCritSvcUndefSvcDownNoPoll() throws
Exception //pollablestest.java
     |
```

[Table of Contents] [Description]

Path 193:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1085. public void testPollIfDownCritSvcUndefSvcDownNoPoll() throws
Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 194:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1118. public void testPollNode() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 195:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1167. public void testServiceEvent() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 196:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1201. public void testInterfaceEvent() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 197:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1224. public void testNodeEvent() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 198:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1247. public void testLingeringSvcDownOnIfUp() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 199:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1282. public void testLingeringSvcDownOnNodeUp() throws
Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 200:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1395. public void testIndependentOutageEventsUpTogether() throws
Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 201:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1430. public void testIndependentOutageEventsUpSeparately() throws
Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 202:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1798. public void testLoadService() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 203:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1828. public void testLoadInterface() throws Exception //pollablestest.java
      |
```

[Table of Contents] [Description]

Path 204:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1859. public void testLoadNode() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 205:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1892. public void testLoadIndependentOutageEventsUpTogether() throws
Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 206:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1939. public void testLoadIndependentOutageEventsUpSeparately() throws
Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 207:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
1996. public void testLock() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 208:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
2025. public void testLockTimeout() throws Exception //pollablestest.java
```

[Table of Contents] [Description]

Path 209:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
79. public void closeFile(Object rrd) throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 210:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
105. public Object createDefinition(String creator, String directory, String dsName,
int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List rraList)
throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 211:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
137. public void createFile(Object rrdDef) throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 212:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
147. public Object openFile(String fileName) throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 213:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
160. public void updateFile(Object rrd, String data) throws
Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 214:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
170. public void initialize() throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 215:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
180. public void graphicsInitialize() throws Exception //jnirrdstrategy.java
```

[Table of Contents] [Description]

Path 216:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
74. public void closeFile(Object rrdFile) throws Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 217:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
81. public Object createDefinition(String creator, String directory, String dsName,
int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List rraList)
throws Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 218:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
112. public void createFile(Object rrdDef) throws Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 219:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
120. public Object openFile(String fileName) throws
Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 220:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
128. public void updateFile(Object rrdFile, String data) throws
Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 221:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
137. public synchronized void initialize() throws Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 222:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
149. public void graphicsInitialize() throws Exception //jrobinrrdstrategy.java
```

[Table of Contents] [Description]

Path 223:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
242. abstract Object process(Object rrd) throws Exception; //queuingrrdstrategy.java
     ...
332. Object process(Object rrd) throws Exception {
```

[Table of Contents] [Description]

Path 224:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
255. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
```

[Table of Contents] [Description]

Path 225:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
289. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 226:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
332. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 227:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
708. public void closeFile(Object rrd) throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 228:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
717. public Object createDefinition(String creator, String directory, String dsName,
int step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List rraList)
throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 229:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
728. public void createFile(Object op) throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 230:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
741. public void initialize() throws Exception //queuingrrdstrategy.java
     |
```

[Table of Contents] [Description]

Path 231:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
750. public void graphicsInitialize() throws Exception //queuingrrdstrategy.java
```

[Table of Contents] [Description]

Path 232:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
759. public Object openFile(String fileName) throws
Exception //queuingrrdstrategy.java
```

[Table of Contents] [Description]

Path 233:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
768. public void updateFile(Object rrdFile, String data) throws
Exception //queuingrrdstrategy.java
```

[Table of Contents] [Description]

Path 234:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
58. void initialize() throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 235:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
64. void graphicsInitialize() throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 236:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
93. Object createDefinition(String creator, String directory, String dsName, int
step, String dsType, int dsHeartbeat, String dsMin, String dsMax, List rraList)
throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 237:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
104. void createFile(Object rrdDef) throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 238:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
118. Object openFile(String fileName) throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 239:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
131. void updateFile(Object rrd, String data) throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 240:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
141. void closeFile(Object rrd) throws Exception; //rrdstrategy.java
```

[Table of Contents] [Description]

Path 241:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
173. public static void main(final String[] args) throws
Exception //rrdstresser.java
```

```
      '
```

[Table of Contents] [Description]

Path 242:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
269. private static void rrdCloseFile(Object rrdFile) throws
Exception //rrdstresser.java
      '
```

[Table of Contents] [Description]

Path 243:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
273. private static Object rrdCreateDefinition(int fileNum) throws
Exception //rrdstresser.java
      '
```

[Table of Contents] [Description]

Path 244:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
285. private static void rrdCreateFile(Object rrdDef) throws
Exception //rrdstresser.java
      '
```

[Table of Contents] [Description]

Path 245:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
289. private static void rrdInitialize() throws Exception //rrdstresser.java
      '
```

[Table of Contents] [Description]

Path 246:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
301. private static Object rrdOpenFile(String fileName) throws
Exception //rrdstresser.java
      '
```

[Table of Contents] [Description]

Path 247:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
309. private static void rrdUpdateFile(Object rrdFile, String data) throws
Exception //rrdstresser.java
```

[Table of Contents] [Description]

Path 248:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
314. public void execute(String[] args) throws Exception //rrdstresser.java
```

[Table of Contents] [Description]

Path 249:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
87. protected void setUp() throws Exception //scheduletest.java
```

[Table of Contents] [Description]

Path 250:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
98. protected void tearDown() throws Exception //scheduletest.java
```

[Table of Contents] [Description]

Path 251:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
101. public static Attribute getAttribute
(org.opennms.netmgt.config.service.Attribute attrib) throws Exception //manager.java
```

[Table of Contents] [Description]

Path 252:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
110. public static Object getArgument(Argument arg) throws Exception //manager.java
```

[Table of Contents] [Description]

Path 253:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
103. static synchronized public EventConfFactory getInstance() throws
Exception //eventconffactory.java
```

[Table of Contents] [Description]

Path 254:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
109. static synchronized public EventConfFactory getInstance(String fileName) throws
Exception //eventconffactory.java
```

[Table of Contents] [Description]

Path 255:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
130. public synchronized void reloadXML() throws Exception //eventconffactory.java
```

[Table of Contents] [Description]

Path 256:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
24. public void setUp() throws Exception //examplewebtestcase.java
```

[Table of Contents] [Description]

Path 257:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
14. protected void setUp() throws Exception //mhtest.java
```

[Table of Contents] [Description]

Path 258:

Query Name - **Overly_Broad_Throws**

Severity - **Information**

```
138. public void sendV1Trap(String message) //snmpagent.java
```

[Table of Contents] [Description]

	Use of system output stream
CWE ID	10033
Description	A dedicated logging class should be used, rather than using default output stream, to ease the monitoring process
Likelihood of Exploit	Low
Common Consequences	Difficulty to maintain the application and to follow log files
Potential Mitigations	Using dedicated logging class
Applicable Platforms	All

Path 1:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
    ...
35. System.out.println("usage: ParseMib [-debug] files");
```

[Table of Contents] [Description]

Path 2:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
31. public static void main(String args[]) //parsemib.java
    ...
72. System.err.println("ERROR: the file '" + arg +
```

[Table of Contents] [Description]

Path 3:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
106. public class SimpleNode //simplenode.java
     ...
```

```
235. System.out.println("building oidNames of oids");
```

[Table of Contents] [Description]

Path 4:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
106. public class SimpleNode //simplenode.java
      ...
277. System.err.println("exception during oidNames init" + e.getMessage());
```

[Table of Contents] [Description]

Path 5:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1033. public void writeOids() //simplenode.java
      ...
1073. System.out.print(sb.toString());
```

[Table of Contents] [Description]

Path 6:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
123. public void setupCallouts (NodeList areaspecNodeList) //calloutemitter.java
      ...
158. System.out.println("Unexpected element in areaset: "
```

[Table of Contents] [Description]

Path 7:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
86. public void formatTextCallout(Emitter rtfEmitter, //formatcallout.java
      ...
104. System.out.println("Transformer Exception in formatTextCallout");
```

[Table of Contents] [Description]

Path 8:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
47. public void formatCallout(Emitter rtfEmitter, //formatgraphiccallout.java
```

```
...
85. System.out.println("Transformer Exception in graphic formatCallout");
```

[Table of Contents] [Description]

Path 9:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
51. public void formatCallout(Emitter rtfEmitter, //formatunicodecallout.java
...
95. System.out.println("Transformer Exception in graphic formatCallout");
```

[Table of Contents] [Description]

Path 10:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
42. public ImageIntrinsics(String imageFn) //imageintrinsics.java
...
94. System.err.println("Failed to interpret image: " + imageFn);
```

[Table of Contents] [Description]

Path 11:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
110. public static int convertLength(String length) //table.java
...
149. System.out.println(digits + " is not a number; 1 used instead.");
```

[Table of Contents] [Description]

Path 12:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
107. public void process( Context context ) throws TransformerException //text.java
...
144. System.out.println("Cannot open " + href);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
66. protected static String getVariable(Context context, String
```

```
varName) //unwraplinks.java
    ...
75. System.out.println("Undefined variable: " + varName);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
105. protected static String getVariable(Context context, String
varName) //verbatim.java
    ...
114. System.out.println("Undefined variable: " + varName);
```

[Table of Contents] [Description]

Path 15:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
76. public void formatTextCallout(DOMBuilder rtf, //formatcallout.java
    ...
94. System.out.println("SAX Exception in text formatCallout");
```

[Table of Contents] [Description]

Path 16:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
39. public void formatCallout(DOMBuilder rtf, //formatdingbatcallout.java
    ...
79. System.out.println("SAX Exception in graphics formatCallout");
```

[Table of Contents] [Description]

Path 17:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
43. public void formatCallout(DOMBuilder rtf, //formatgraphiccallout.java
    ...
80. System.out.println("SAX Exception in graphics formatCallout");
```

[Table of Contents] [Description]

Path 18:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
43. public void formatCallout(DOMBuilder rtf, //formatunicodecallout.java
      ...
      84. System.out.println("SAX Exception in unicode formatCallout");
```

[Table of Contents] [Description]

Path 19:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
32. public DocumentFragment doSomething(NodeIterator rtf) //func.java
      33. System.out.println("Got here 2: " + rtf);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
14. public static String getString(ExpressionContext context, //params.java
      ...
      21. System.out.println("$" + varName + " is not a defined parameter.");
```

[Table of Contents] [Description]

Path 21:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
111. public static int convertLength(String length) //table.java
      ...
      150. System.out.println(digits + " is not a number; 1 used instead.");
```

[Table of Contents] [Description]

Path 22:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
67. public String insertfile(XSLProcessorContext context, //text.java
      ...
      93. System.out.println("Cannot find base URI for " + baseURI);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
158. public DocumentFragment numberLines (ExpressionContext context, //verbatim.java
```

```
...
174. System.out.println("PCE!");
```

[Table of Contents] [Description]

Path 24:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
214. public void mailSend() throws JavaMailerException //javamailer.java
...
297. System.out.println("Response: " + t.getLastServerResponse());
```

[Table of Contents] [Description]

Path 25:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
177. public void install(String[] argv) throws Exception //installer.java
...
286. System.out.println();
```

[Table of Contents] [Description]

Path 26:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
297. m_out.println("================================================"
```

[Table of Contents] [Description]

Path 27:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
...
299. m_out.println("OpenNMS Installer Version " + s_version);
```

[Table of Contents] [Description]

Path 28:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
```

```
...
300. m_out.println("============================================="
```

[Table of Contents] [Description]

Path 29:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
...
302. m_out.println("");
```

[Table of Contents] [Description]

Path 30:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
...
303. m_out.println("Configures PostgreSQL tables, users, and other "
```

[Table of Contents] [Description]

Path 31:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
296. public void printHeader() //installer.java
...
305. m_out.println("");
```

[Table of Contents] [Description]

Path 32:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
471. public void printDiagnostics() //installer.java
472. m_out.println("* using '" + m_user + "' as the PostgreSQL "
```

[Table of Contents] [Description]

Path 33:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
471. public void printDiagnostics() //installer.java
```

```
...
474. m_out.println("* using '" + m_pass + "' as the PostgreSQL "
```

[Table of Contents] [Description]

Path 34:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
471. public void printDiagnostics() //installer.java
...
476. m_out.println("* using '" + m_database + "' as the PostgreSQL "
```

[Table of Contents] [Description]

Path 35:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
624. m_out.print("- checking database version... ");
```

[Table of Contents] [Description]

Path 36:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
...
657. m_out.println(Float.toString(m_pg_version));
```

[Table of Contents] [Description]

Path 37:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
623. public void databaseCheckVersion() throws Exception //installer.java
...
658. m_out.println(" - Full version string: " + versionString);
```

[Table of Contents] [Description]

Path 38:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
719. public void checkOldTables() throws Exception //installer.java
```

```
...
725. m_out.print("- checking database for old backup tables... ");
```

[Table of Contents] [Description]

Path 39:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
719. public void checkOldTables() throws Exception //installer.java
...
736. m_out.println("NONE");
```

[Table of Contents] [Description]

Path 40:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
...
787. m_out.print("- checking for rows that violate constraints... ");
```

[Table of Contents] [Description]

Path 41:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
784. public void checkConstraints() throws Exception //installer.java
...
828. m_out.println("NONE");
```

[Table of Contents] [Description]

Path 42:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
843. public void fixConstraint() throws Exception //installer.java
...
847. m_out.print("- fixing rows that violate constraint "
```

[Table of Contents] [Description]

Path 43:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
843. public void fixConstraint() throws Exception //installer.java
```

```
...
899. m_out.println(change_text + " " + num + (num == 1 ? " ROW" : " ROWS"));
```

[Table of Contents] [Description]

Path 44:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
949. m_out.println("- creating sequences... ");
```

[Table of Contents] [Description]

Path 45:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
968. m_out.print(" - checking \"" + sequence + "\" minimum value... ");
```

[Table of Contents] [Description]

Path 46:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
989. m_out.println(Integer.toString(minvalue));
```

[Table of Contents] [Description]

Path 47:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
991. m_out.print(" - removing sequence \"" + sequence + "\"... ");
```

[Table of Contents] [Description]

Path 48:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
```

```
...
999. m_out.println("REMOVED");
```

[Table of Contents] [Description]

Path 49:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
1001. m_out.println("CLEAN");
```

[Table of Contents] [Description]

Path 50:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
1004. m_out.print(" - creating sequence \"" + sequence + "\"... ");
```

[Table of Contents] [Description]

Path 51:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
1007. m_out.println("OK");
```

[Table of Contents] [Description]

Path 52:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
945. public void createSequences() throws Exception //installer.java
...
1010. m_out.println("- creating sequences... DONE");
```

[Table of Contents] [Description]

Path 53:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
```

```
...
1018. m_out.println("- creating tables...");
```

[Table of Contents] [Description]

Path 54:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1035. m_out.print(" - removing old table... ");
```

[Table of Contents] [Description]

Path 55:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1038. m_out.println("REMOVED");
```

[Table of Contents] [Description]

Path 56:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1040. m_out.println("CLEAN");
```

[Table of Contents] [Description]

Path 57:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1043. m_out.print(" - creating table \"" + table + "\"... ");
```

[Table of Contents] [Description]

Path 58:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
```

```
...
1045. m_out.println("CREATED");
```

[Table of Contents] [Description]

Path 59:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1047. m_out.print(" - giving \"" + m_user + "\" permissions on \""
```

[Table of Contents] [Description]

Path 60:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1050. m_out.println("GRANTED");
```

[Table of Contents] [Description]

Path 61:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1052. m_out.print(" - checking table \"" + table + "\"... ");
```

[Table of Contents] [Description]

Path 62:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1060. m_out.println("UPTODATE");
```

[Table of Contents] [Description]

Path 63:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
```

```
...
1067. m_out.println("CREATED");
```

[Table of Contents] [Description]

Path 64:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1013. public void createTables() throws Exception //installer.java
...
1075. m_out.println("- creating tables... DONE");
```

[Table of Contents] [Description]

Path 65:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
...
1082. m_out.println("- creating indexes...");
```

[Table of Contents] [Description]

Path 66:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
...
1089. m_out.print(" - creating index \"" + index + "\"... ");
```

[Table of Contents] [Description]

Path 67:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
...
1097. m_out.println("EXISTS");
```

[Table of Contents] [Description]

Path 68:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
```

```
...
1100. m_out.println("OK");
```

[Table of Contents] [Description]

Path 69:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1078. public void createIndexes() throws Exception //installer.java
...
1104. m_out.println("- creating indexes... DONE");
```

[Table of Contents] [Description]

Path 70:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1178. void insertData() throws Exception //installer.java
...
1186. m_out.print("- inserting initial table data for \"" + table
```

[Table of Contents] [Description]

Path 71:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1178. void insertData() throws Exception //installer.java
...
1208. m_out.println("EXISTS");
```

[Table of Contents] [Description]

Path 72:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1178. void insertData() throws Exception //installer.java
...
1210. m_out.println("OK");
```

[Table of Contents] [Description]

Path 73:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
```

```
...
1219. m_out.print("- checking if database \"" + m_database
```

[Table of Contents] [Description]

Path 74:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1226. m_out.println("ALREADY UNICODE");
```

[Table of Contents] [Description]

Path 75:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1230. m_out.println("NOT UNICODE, CONVERTING");
```

[Table of Contents] [Description]

Path 76:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1244. m_out.print(" - dumping data to " + dumpFile + "... ");
```

[Table of Contents] [Description]

Path 77:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1252. m_out.println("OK");
```

[Table of Contents] [Description]

Path 78:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
```

```
...
1254. m_out.print(" - waiting 3s for PostgreSQL to notice "
```

[Table of Contents] [Description]

Path 79:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1257. m_out.print(".");
```

[Table of Contents] [Description]

Path 80:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1259. m_out.print(".");
```

[Table of Contents] [Description]

Path 81:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1261. m_out.println(" OK");
```

[Table of Contents] [Description]

Path 82:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1263. m_out.print(" - dropping old database... ");
```

[Table of Contents] [Description]

Path 83:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
```

```
...
1271. m_out.println("OK");
```

[Table of Contents] [Description]

Path 84:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1273. m_out.print(" - creating new unicode database... ");
```

[Table of Contents] [Description]

Path 85:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1281. m_out.println("OK");
```

[Table of Contents] [Description]

Path 86:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1283. m_out.print(" - recreating tables... ");
```

[Table of Contents] [Description]

Path 87:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1291. m_out.println("OK");
```

[Table of Contents] [Description]

Path 88:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
```

```
...
1293. m_out.print(" - restoring data... ");
```

[Table of Contents] [Description]

Path 89:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1215. public void checkUnicode() throws Exception //installer.java
...
1301. m_out.println("OK");
```

[Table of Contents] [Description]

Path 90:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1340. public void verifyFileExists(boolean isDir, String file, //installer.java
...
1351. m_out.print("- using " + description + "... ");
```

[Table of Contents] [Description]

Path 91:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1340. public void verifyFileExists(boolean isDir, String file, //installer.java
...
1375. m_out.println(f.getAbsolutePath());
```

[Table of Contents] [Description]

Path 92:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1382. m_out.print("- adding stored procedures... ");
```

[Table of Contents] [Description]

Path 93:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
```

```
...
1398. m_out.print("\n - " + list[i].getName() + "... ");
```

[Table of Contents] [Description]

Path 94:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1437. m_out.print("OK (dropped and re-added)");
```

[Table of Contents] [Description]

Path 95:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1439. m_out.print("EXISTS");
```

[Table of Contents] [Description]

Path 96:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1443. m_out.print("OK");
```

[Table of Contents] [Description]

Path 97:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1446. m_out.println("");
```

[Table of Contents] [Description]

Path 98:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1487. public void checkWebappOldOpennmsDir() throws Exception //installer.java
```

```
...
1490. m_out.print("- Checking for old opennms webapp directory in " +
f.getAbsolutePath() + "... ");
```

[Table of Contents] [Description]

Path 99:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1487. public void checkWebappOldOpennmsDir() throws Exception //installer.java
...
1498. m_out.println("OK");
```

[Table of Contents] [Description]

Path 100:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1501. public void checkServerXmlOldOpennmsContext() throws
Exception //installer.java
...
1508. m_out.print("- Checking for old opennms context in " + f.getAbsolutePath
() + "... ");
```

[Table of Contents] [Description]

Path 101:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1501. public void checkServerXmlOldOpennmsContext() throws
Exception //installer.java
...
1511. m_out.println("DID NOT CHECK (file does not exist)");
```

[Table of Contents] [Description]

Path 102:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1501. public void checkServerXmlOldOpennmsContext() throws
Exception //installer.java
...
1530. m_out.println("OK");
```

[Table of Contents] [Description]

Path 103:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1535. public void installWebApp() throws Exception //installer.java
      ...
1538. m_out.println("- Install OpenNMS webapp... ");
```

[Table of Contents] [Description]

Path 104:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1535. public void installWebApp() throws Exception //installer.java
      ...
1553. m_out.println("- Installing OpenNMS webapp... DONE");
```

[Table of Contents] [Description]

Path 105:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1556. public void installLink(String source, String destination, //installer.java
      ...
1565. m_out.print(" - " + destination + " exists, removing... ");
```

[Table of Contents] [Description]

Path 106:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1556. public void installLink(String source, String destination, //installer.java
      ...
1567. m_out.println("REMOVED");
```

[Table of Contents] [Description]

Path 107:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1556. public void installLink(String source, String destination, //installer.java
      ...
1570. m_out.print(" - creating link to " + destination + "... ");
```

[Table of Contents] [Description]

Path 108:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1556. public void installLink(String source, String destination, //installer.java
      ...
1584. m_out.println("DONE");
```

[Table of Contents] [Description]

Path 109:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1587. public void updateTomcatConf() throws Exception //installer.java
      ...
1597. m_out.print("- setting tomcat4 user to 'root'... ");
```

[Table of Contents] [Description]

Path 110:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1587. public void updateTomcatConf() throws Exception //installer.java
      ...
1622. m_out.println("done");
```

[Table of Contents] [Description]

Path 111:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      ...
1655. m_out.print("- checking for stale iplike references... ");
```

[Table of Contents] [Description]

Path 112:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      ...
1658. m_out.println("REMOVED");
```

[Table of Contents] [Description]

Path 113:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1665. m_out.println("CLEAN");
```

[Table of Contents] [Description]

Path 114:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1674. m_out.print("- checking for stale eventtime.so references... ");
```

[Table of Contents] [Description]

Path 115:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1677. m_out.println("REMOVED");
```

[Table of Contents] [Description]

Path 116:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1684. m_out.println("CLEAN");
```

[Table of Contents] [Description]

Path 117:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1690. m_out.print("- adding iplike database function... ");
```

[Table of Contents] [Description]

Path 118:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1651. public void updateIplike() throws Exception //installer.java
      |...
      1693. m_out.println("OK");
```

[Table of Contents] [Description]

Path 119:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1700. m_out.print("- adding PL/pgSQL call handler... ");
```

[Table of Contents] [Description]

Path 120:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1704. m_out.println("EXISTS");
```

[Table of Contents] [Description]

Path 121:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1708. m_out.println("OK");
```

[Table of Contents] [Description]

Path 122:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1711. m_out.print("- adding PL/pgSQL language module... ");
```

[Table of Contents] [Description]

Path 123:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1719. m_out.println("EXISTS");
```

[Table of Contents] [Description]

Path 124:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1696. public void updatePlPgsql() throws Exception //installer.java
      |...
      1723. m_out.println("OK");
```

[Table of Contents] [Description]

Path 125:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2054. m_out.println("SCHEMA DOES NOT MATCH");
```

[Table of Contents] [Description]

Path 126:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2055. m_out.println(" - differences:");
```

[Table of Contents] [Description]

Path 127:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2064. m_out.println(" - column \"" + newColumn.getName()
```

[Table of Contents] [Description]

Path 128:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      |2067. m_out.println(" - old column: "
```

[Table of Contents] [Description]

Path 129:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      |2070. m_out.println(" - new column: " + newColumn);
```

[Table of Contents] [Description]

Path 130:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      |2105. m_out.println(message + ". Ignoring due to '-N'");
```

[Table of Contents] [Description]

Path 131:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      |2117. m_out.println(message + ". Ignoring due to '-N'");
```

[Table of Contents] [Description]

Path 132:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
```

```
newColumns) //installer.java
    ...
2139. m_out.println(" * WARNING: column \""
```

[Table of Contents] [Description]

Path 133:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
    ...
2153. m_out.print(" - creating temporary table... ");
```

[Table of Contents] [Description]

Path 134:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
    ...
2156. m_out.println("done");
```

[Table of Contents] [Description]

Path 135:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
    ...
2160. m_out.print(" - creating new '" + table + "' table... ");
```

[Table of Contents] [Description]

Path 136:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
    ...
2163. m_out.println("done");
```

[Table of Contents] [Description]

Path 137:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2169. m_out.print(" - optimizing table " + table + "... ");
```

[Table of Contents] [Description]

Path 138:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2171. m_out.println("DONE");
```

[Table of Contents] [Description]

Path 139:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2174. m_out.println("FAILED! Not reverting due to '-R' being "
```

[Table of Contents] [Description]

Path 140:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |...
      2195. m_out.println("FAILED! Old data restored, however indexes and "
```

[Table of Contents] [Description]

Path 141:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2040. public void changeTable(String table, List oldColumns, List
newColumns) //installer.java
      |
```

```
...
2204. m_out.println(" - completed updating table... ");
```

[Table of Contents] [Description]

Path 142:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2240. m_out.print(" - transforming data into the new table...\r");
```

[Table of Contents] [Description]

Path 143:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2258. m_out.println(" - performing select: " + dbcmd);
```

[Table of Contents] [Description]

Path 144:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2267. m_out.println(" - performing insert: " + dbcmd);
```

[Table of Contents] [Description]

Path 145:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2298. m_out.println(" - don't know what to do "
```

[Table of Contents] [Description]

Path 146:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
```

```
...
2312. m_out.println(" - " + name + " was NULL but is a "
```

[Table of Contents] [Description]

Path 147:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2322. m_out.println(" - " + name
```

[Table of Contents] [Description]

Path 148:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2328. m_out.println(" - " + obj + " -> " + newObj);
```

[Table of Contents] [Description]

Path 149:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2334. m_out.println(" - " + name + " = " + obj);
```

[Table of Contents] [Description]

Path 150:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2338. m_out.println(" - " + name + " = undefined");
```

[Table of Contents] [Description]

Path 151:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
```

```
...
2366. System.err.print(" - transforming data into the new "
```

[Table of Contents] [Description]

Path 152:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2207. public void transformData(String table, String oldTable, //installer.java
...
2385. m_out.println(" - transforming data into the new table... "
```

[Table of Contents] [Description]

Path 153:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
2390. m_out.println("usage:");
```

[Table of Contents] [Description]

Path 154:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2391. m_out.println(" $OPENNMS_HOME/bin/install -h");
```

[Table of Contents] [Description]

Path 155:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2392. m_out.println(" $OPENNMS_HOME/bin/install "
```

[Table of Contents] [Description]

Path 156:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2394. m_out.println(" [-y] [-X]");
```

[Table of Contents] [Description]

Path 157:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2395. m_out.println(" "
```

[Table of Contents] [Description]

Path 158:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2397. m_out.println(" "
```

[Table of Contents] [Description]

Path 159:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2399. m_out.println(" "
```

[Table of Contents] [Description]

Path 160:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2401. m_out.println(" "
```

[Table of Contents] [Description]

Path 161:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2403. m_out.println(" "
```

[Table of Contents] [Description]

Path 162:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2405. m_out.println(" " + "[-C <constraint>]");
```

[Table of Contents] [Description]

Path 163:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2406. m_out.println("");
```

[Table of Contents] [Description]

Path 164:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2407. m_out.println(m_required_options);
```

[Table of Contents] [Description]

Path 165:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2408. m_out.println("");
```

[Table of Contents] [Description]

Path 166:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2409. m_out.println(" -h this help");
```

[Table of Contents] [Description]

Path 167:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2410. m_out.println("");
```

[Table of Contents] [Description]

Path 168:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2411. m_out.println(" -d perform database actions");
```

[Table of Contents] [Description]

Path 169:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2412. m_out.println(" -i insert data into the database");
```

[Table of Contents] [Description]

Path 170:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2413. m_out.println(" -s update iplike postgres function");
```

[Table of Contents] [Description]

Path 171:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2414. m_out.println(" -U upgrade database to unicode, if needed");
```

[Table of Contents] [Description]

Path 172:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2415. m_out.println(" -y install web application (see -w and -W)");
```

[Table of Contents] [Description]

Path 173:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2416. m_out.println("");
```

[Table of Contents] [Description]

Path 174:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2417. m_out.println(" -u username of the PostgreSQL "
```

[Table of Contents] [Description]

Path 175:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2419. m_out.println(" -p password of the PostgreSQL "
```

[Table of Contents] [Description]

Path 176:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2421. m_out.println(" -c drop and recreate tables that already "
```

[Table of Contents] [Description]

Path 177:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2423. m_out.println("");
```

[Table of Contents] [Description]

Path 178:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2424. m_out.println(" -T location of tomcat.conf");
```

[Table of Contents] [Description]

Path 179:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2425. m_out.println(" -w location of tomcat's webapps directory");
```

[Table of Contents] [Description]

Path 180:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2426. m_out.println(" -W location of tomcat's server/lib "
```

[Table of Contents] [Description]

Path 181:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2428. m_out.println("");
```

[Table of Contents] [Description]

Path 182:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2429. m_out.println(" -r run as an RPM install (does nothing)");
```

[Table of Contents] [Description]

Path 183:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2430. m_out.println(" -x turn on debugging for database data "
```

[Table of Contents] [Description]

Path 184:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2432. m_out.println(" -N ignore NOT NULL constraint checks when "
```

[Table of Contents] [Description]

Path 185:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2434. m_out.println(" useful after a table is reverted by a "
```

[Table of Contents] [Description]

Path 186:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
```

```
...
2436. m_out.println(" -R do not revert a table to the original if "
```

[Table of Contents] [Description]

Path 187:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2438. m_out.println(" transforming data -- only used for debugging");
```

[Table of Contents] [Description]

Path 188:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2439. m_out.println(" -C fix rows that violate the specified "
```

[Table of Contents] [Description]

Path 189:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2441. m_out.println(" affected rows to NULL by default");
```

[Table of Contents] [Description]

Path 190:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
...
2443. .println(" -X drop rows that violate constraint instead of marking key
column in");
```

[Table of Contents] [Description]

Path 191:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
2389. public void printHelp() //installer.java
      |...
      2444. m_out.println(" affected rows to NULL (used with \"-C\")");
```

[Table of Contents] [Description]

Path 192:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
280. public void run() //snmpportal.java
      |...
      325. System.out.println(new Date() + " - Exception: " + e.getMessage());
```

[Table of Contents] [Description]

Path 193:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
      |...
      266. out.print("0x");
```

[Table of Contents] [Description]

Path 194:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
      |...
      267. out.print(Integer.toHexString((b >> 4) & 0xf));
```

[Table of Contents] [Description]

Path 195:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
      |...
      268. out.print(Integer.toHexString(b & 0xf));
```

[Table of Contents] [Description]

Path 196:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
     ...
269. out.print(" ");
```

[Table of Contents] [Description]

Path 197:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
259. static void dumpHex(PrintStream out, byte[] data, int offset, int
length) //snmputil.java
     ...
274. out.println("");
```

[Table of Contents] [Description]

Path 198:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
152. public void print () //calendartablebuilder.java
     ...
161. System.out.println("index: " + index + "visible: " +
```

[Table of Contents] [Description]

Path 199:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
40. private void testVisibleDays(Locale locale, int year, int month, int
dayOfMonth) //calendartablebuildertest.java
     ...
44. System.out.println("day " + i + " " + builder.m_days[i].getVisible());
```

[Table of Contents] [Description]

Path 200:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
475. public static void main(String[] args) //eventsarchiver.java
     ...
487. System.err.println(ae.getMessage());
```

[Table of Contents] [Description]

Path 201:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
998. public static void main(String[] args) //dbifserviceentry.java
       ...
1001.  System.out.println(entry.toString());
```

[Table of Contents] [Description]

Path 202:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1107. public static void main(String[] args) //dbipinterfaceentry.java
       ...
1110.  System.out.println(entry.toString());
```

[Table of Contents] [Description]

Path 203:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1055. public static void main(String[] args) //dbsnmpinterfaceentry.java
       ...
1059.  System.out.println(entry.toString());
```

[Table of Contents] [Description]

Path 204:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
244. public String getEventLabel(String uei) //eventconffactory.java
       245.  System.out.println("looking for " + uei);
```

[Table of Contents] [Description]

Path 205:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
285. public void onEvent(Event event) //mocknetworktest.java
       286.  System.err.println("onEvent: " + event.getUei());
```

[Table of Contents] [Description]

Path 206:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
209. public synchronized boolean checkAnticipated() //outageanticipator.java
       ...
       227. System.err.println("Expected outage "+expectedOutage.toDetailedString()+"
       not in current Set");
```

[Table of Contents] [Description]

Path 207:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
51. public static void main(String[] argv) throws Exception //checknrpe.java
       ...
       61. System.out.print(s_usage);
```

[Table of Contents] [Description]

Path 208:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
51. public static void main(String[] argv) throws Exception //checknrpe.java
       ...
       98. System.out.println(p.getBuffer());
```

[Table of Contents] [Description]

Path 209:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
59. public static void main(String[] args) //checknsc.java
       ...
       139. System.out.println("NsclientPlugin: "
```

[Table of Contents] [Description]

Path 210:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
151. private static void usage(String message, Exception e) //checknsc.java
       ...
       170. System.out.print(sb);
```

[Table of Contents] [Description]

Path 211:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
173. public static void main(String[] argv) //interface.java
       ...
181. System.out.println("filename=" + filename);
```

[Table of Contents] [Description]

Path 212:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
255. Object process(Object rrd) throws Exception //queuingrrdstrategy.java
       ...
258. System.err.println("WHAT! rrd open but not created?");
```

[Table of Contents] [Description]

Path 213:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
241. private static void printHeader() //rrdstresser.java
       242. System.out.println
       ("************************************************************************");
```

[Table of Contents] [Description]

Path 214:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
383. public static void main(String[] args) throws IOException //execrunner.java
       ...
398. System.out.println("Testing ExecRunner with StringWriter...");
```

[Table of Contents] [Description]

Path 215:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
428. public void doTestLoadLibraries() //manager.java
       ...
439. System.err.println(message);
```

[Table of Contents] [Description]

Path 216:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
468. public static void main(String[] args) //manager.java
      |...
      |487. System.out.println("");
```

[Table of Contents] [Description]

Path 217:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
108. private void initREs() //nessusparser.java
      |...
      |159. System.out.println("FATAL ERROR in regex in NessusScan.java. Correct this
      |error and rebuild.");
```

[Table of Contents] [Description]

Path 218:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
      |...
      |230. System.out.println("Add Capsd Info 1");
```

[Table of Contents] [Description]

Path 219:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
285. public void deleteThese(java.util.List deleteServices) throws
IOException //pollerconfigservlet.java
      |...
      |307. System.out.println("Package removed " + svc.getName());
```

[Table of Contents] [Description]

Path 220:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
60. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //servicenoticeupdateservlet.java
      |...
      |68. System.out.println(checkedServices[i]);
```

[Table of Contents] [Description]

Path 221:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updatenotifdstatusservlet.java
      |
      ...
      54. System.out.println("status = " + request.getParameter("status"));
```

[Table of Contents] [Description]

Path 222:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
294. public void deleteThese(java.util.List deleteServices) throws
IOException //pollerconfigservlet.java
      |
      ...
      316. System.out.println("Package removed " + svc.getName());
```

[Table of Contents] [Description]

Path 223:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
      |
      ...
      114. out.println(buffer.toString());
```

[Table of Contents] [Description]

Path 224:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
59. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //exportassetsservlet.java
      |
      ...
      206. out.println(buffer.toString());
```

[Table of Contents] [Description]

Path 225:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
      |
```

```
...
234. out.println("<table width=\"100%\" border=\"1\" cellspacing=\"0\" " +
"cellpadding=\"2\" bordercolor=\"black\" " + "bgcolor=\"#cccccc\">");
```

[Table of Contents] [Description]

Path 226:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java

...
242. out.println("<tr bgcolor=\"#999999\">");
```

[Table of Contents] [Description]

Path 227:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java

...
243. out.println("<td width=\"50%\"><b>" + sectionName + "</b></td>");
```

[Table of Contents] [Description]

Path 228:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java

...
244. out.println("<td width=\"20%\" align=\"right\">" +
"<b>Outages</b></td>");
```

[Table of Contents] [Description]

Path 229:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java

...
245. out.println("<td width=\"30%\" align=\"right\">" + "<b>24hr
Avail</b></td>");
```

[Table of Contents] [Description]

Path 230:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
246. out.println("</tr>");
```

[Table of Contents] [Description]

Path 231:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
274. out.println("<tr>");
```

[Table of Contents] [Description]

Path 232:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
276. out.println("<td><a href=\"rtc/category.jsp?category=" +
URLEncoder.encode(response.encodeURL(categoryName), "UTF-8") + "\" title=\"" +
title + "\">" + categoryName + "</a></td>");
```

[Table of Contents] [Description]

Path 233:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
277. out.println("<td bgcolor=\"" + outageColor + "\" align=\"right\"
title=\"Updated: " + lastUpdated + "\">" + outageText + "</td>");
```

[Table of Contents] [Description]

Path 234:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
```

```
IOException, MarshalException, ValidationException //categorylist.java
    ...
278. out.println("<td bgcolor=\"" + availColor + "\" align=\"right\"
title=\"Updated: " + lastUpdated + "\">" + availText + "</td>");
```

[Table of Contents] [Description]

Path 235:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
279. out.println("<!-- Last updated " + lastUpdated + " -->");
```

[Table of Contents] [Description]

Path 236:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
280. out.println("<!-- Epoch time: " + lastUpdatedTime + " -->");
```

[Table of Contents] [Description]

Path 237:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
282. out.println("</tr>");
```

[Table of Contents] [Description]

Path 238:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
    ...
286. out.println("<tr bgcolor=\"#999999\">");
```

[Table of Contents] [Description]

Path 239:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
288. out.println("<td colspan=\"3\"><font color=\"#bb1111\">" + "OpenNMS
Disconnect -- is the OpenNMS daemon " + "running?<br/>Last update: " +
(earliestUpdate > 0 ? new Date(earliestUpdate).toString() : "one or more
categories have never been updated.") + "</font></td>");
```

[Table of Contents] [Description]

Path 240:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
290. out.println("<td colspan=\"3\">Percentage over last " + "24 hours</td>");
```

[Table of Contents] [Description]

Path 241:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
293. out.println("</tr>");
```

[Table of Contents] [Description]

Path 242:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
230. private void printBox(JspWriter out, HttpServletResponse response) throws
IOException, MarshalException, ValidationException //categorylist.java
     ...
294. out.println("</table>");
```

[Table of Contents] [Description]

Path 243:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //rtcpostservlet.java
      |...
      |128. out.println("Category data parsed successfully.");
```

[Table of Contents] [Description]

Path 244:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
122. private void log(String message) //documentgenerator.java
      |...
      |124. System.err.println(message);
```

[Table of Contents] [Description]

Path 245:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
65. private void log(String message) //mapnodefactory.java
      |66. System.err.println(message);
```

[Table of Contents] [Description]

Path 246:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
556. public static Notification[] getNotices(SortStyle sortStyle, AcknowledgeType
ackType, Filter[] filters, int limit, int offset) throws
SQLException //noticefactory.java
      |...
      |586. System.out.println(select.toString());
```

[Table of Contents] [Description]

Path 247:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
277. public int execute() //command.java
      |...
      |340. System.out.println(e);
```

[Table of Contents] [Description]

Path 248:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
66. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
      |
      ...
      75. out.println(nodes[i].nodeLabel + ", " + nodes[i].nodeId);
```

[Table of Contents] [Description]

Path 249:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
66. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
      |
      ...
      78. out.println("&lt;blank&gt; (change this later)");
```

[Table of Contents] [Description]

Path 250:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
      |
      ...
      79. out.println(nodes[i].nodeLabel + ", " + nodes[i].nodeId);
```

[Table of Contents] [Description]

Path 251:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
70. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdlistservlet.java
      |
      ...
      82. out.println("&lt;blank&gt; (change this later)");
```

[Table of Contents] [Description]

Path 252:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <%@page language="java" contentType="text/html" session="true" //editoutage.jsp
      |
      ...
      183. System.out.println("Checking "+name);
```

[Table of Contents] [Description]

Path 253:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
463. private String getAllGroupsMinusInView(View view) //modifyview.jsp
     ...
488. System.out.println("groups are null in view");
```

[Table of Contents] [Description]

Path 254:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //index.jsp
     ...
146. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 255:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //unknownexception.jsp
     ...
112. out.print("Caused by: ");
```

[Table of Contents] [Description]

Path 256:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //notification-box.jsp
     ...
75. out.println( java.text.MessageFormat.format( format, new Object[] { new
Integer(count) } ));
```

[Table of Contents] [Description]

Path 257:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //notification-box.jsp
     ...
89. out.println( java.text.MessageFormat.format( format, new Object[] { new
Integer(count) } ));
```

[Table of Contents] [Description]

Path 258:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //list.jsp
       ...
142. <tr <% if( i%2 == 0 ) out.print( "BGCOLOR=\"#cccccc\""); %> valign="top">
```

[Table of Contents] [Description]

Path 259:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //current.jsp
       ...
166. <tr valign="top" <% if( nodeIndex%2 == 0 ) out.print
( "BGCOLOR=\"#cccccc\""); %>>
```

[Table of Contents] [Description]

Path 260:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       ...
108. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
out.print("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 261:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       ...
120. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
" ); %>>
```

[Table of Contents] [Description]

Path 262:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       ...
135. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
```

```
("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 263:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
...
147. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
>>
```

[Table of Contents] [Description]

Path 264:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
...
174. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
out.print("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 265:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
...
186. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
" ); %>>
```

[Table of Contents] [Description]

Path 266:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
...
201. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 267:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
       |
       ...
       213. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
       >>
```

[Table of Contents] [Description]

Path 268:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       |
       ...
       110. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
       out.print("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 269:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       |
       ...
       122. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
       " ); %>>
```

[Table of Contents] [Description]

Path 270:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       |
       ...
       137. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
       ("selected=\"selected\" ");%>><%=MONTHS[i]%></option>
```

[Table of Contents] [Description]

Path 271:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //adhoc3.jsp
       |
       ...
       149. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
       >>
```

[Table of Contents] [Description]

Path 272:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      ...
      174. <option value="<%=i%>" <% if( yesterday.get( Calendar.MONTH ) == i )
      out.print("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 273:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      ...
      186. <option value="<%=i%>" <% if( yesterdayHour == i ) out.print( "selected
      " ); %>>
```

[Table of Contents] [Description]

Path 274:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      ...
      201. <option value="<%=i%>" <% if( now.get( Calendar.MONTH ) == i ) out.print
      ("selected ");%>><%=months[i]%></option>
```

[Table of Contents] [Description]

Path 275:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
1. <!-- //choosereportanddate.jsp
      ...
      213. <option value="<%=i%>" <% if( nowHour == i ) out.print( "selected " ); %
      >>
```

[Table of Contents] [Description]

Path 276:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
165. public void run() //inetechoxchange.java
      ...
      190. System.out.println("Invalid Stream Marker (" + len + " != " +
```

```
    compdata.length + ")");
```

[Table of Contents] [Description]

Path 277:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
245. public static void main(String[] args) //inetechoxchange.java
    ...
260. System.out.println("Starting new echo thread");
```

[Table of Contents] [Description]

Path 278:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
179. public void run() //sockettest.java
    ...
248. System.out.println("SocketTest: finished starting client & handler pair
number " + client_id);
```

[Table of Contents] [Description]

Path 279:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
382. public void run() //sockettest.java
    ...
408. System.out.println("SocketTest: client '" + clnt.getName() + "' appears
to be hung!");
```

[Table of Contents] [Description]

Path 280:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
89. public snmpagent(String managerAddress, int managerPort, //snmpagent.java
    ...
109. System.out.println("SNMP agent creation error");
```

[Table of Contents] [Description]

Path 281:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
276. public void snmpInternalError(SnmpSession session, int err, SnmpSyntax
pdu) //snmpwalk.java
      ...
278. System.err.println("An unexpected error occured with the SNMP Session");
```

[Table of Contents] [Description]

Path 282:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
313. public void snmpReceivedPdu(SnmpSession session, int cmd, SnmpPduPacket
pdu) //snmpwalk.java
      ...
333. System.out.println("End of mib reached");
```

[Table of Contents] [Description]

Path 283:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
361. public void snmpInternalError(SnmpSession session, int err, SnmpSyntax
pdu) //snmpwalkmv.java
      ...
363. System.err.println("An unexpected error occured with the SNMP Session");
```

[Table of Contents] [Description]

Path 284:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
398. public void snmpReceivedPdu(SnmpSession session, int cmd, SnmpPduPacket
pdu) //snmpwalkmv.java
      ...
418. System.out.println("End of mib reached");
```

[Table of Contents] [Description]

Path 285:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
53. public static void main (String args[]) //trapd.java
      ...
58. System.out.println("SNMP Trap Receiver Started");
```

[Table of Contents] [Description]

Path 286:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
83. public void snmpReceivedTrap(SnmpTrapSession session, //trapd.java
      ...
      104. System.out.print("Varbind[" + i + "] := " + vb.getName().toString());
```

[Table of Contents] [Description]

Path 287:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
15. public SimulatorController(InetAddress controlChannel) throws
RuntimeException //simulatorcontroller.java
      ...
      19. System.out.println(this._localInetAddress = lookupLocalInetAddress());
```

[Table of Contents] [Description]

Path 288:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
40. protected void sendCommand(String command) //simulatorcontroller.java
      ...
      62. System.err.println("Process timed out executing command '" + command +
      "'");
```

[Table of Contents] [Description]

Path 289:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
      73. out.print("ID");
```

[Table of Contents] [Description]

Path 290:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
```

```
74. out.print("\t\t");
```

[Table of Contents] [Description]

Path 291:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
...
77. out.print(this.getOption(i));
```

[Table of Contents] [Description]

Path 292:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
...
78. out.print("\t\t");
```

[Table of Contents] [Description]

Path 293:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
...
80. out.println();
```

[Table of Contents] [Description]

Path 294:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
...
84. out.print(j);
```

[Table of Contents] [Description]

Path 295:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
      85. out.print("\t\t");
```

[Table of Contents] [Description]

Path 296:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
      90. out.print(poss[k]);
```

[Table of Contents] [Description]

Path 297:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
      91. out.print("\t\t");
```

[Table of Contents] [Description]

Path 298:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
71. public void dump(java.io.PrintStream out) throws
java.io.IOException //chinesemenu.java
      ...
      93. out.println();
```

[Table of Contents] [Description]

Path 299:

Query Name - **Use_of_System_Output_Stream**

Severity - **Information**

```
11. public static void main(String[] args) throws UnknownHostException,
java.io.IOException //tester.java
      ...
      15. System.err.println("Usage: <cmd> <interval> <controllerIp> <int1Ip>
      <int2Ip>");
```

SQL Injection

CWE ID	89
Description	SQL injection attacks are another instantiation of injection attack, in which SQL commands are injected into data-plane input in order to effect the execution of predefined SQL commands.
Likelihood of Exploit	Very High
Common Consequences	Confidentiality: Since SQL databases generally hold sensitive data, loss of confidentiality is a frequent problem with SQL injection vulnerabilities. Authentication: If poor SQL commands are used to check user names and passwords, it may be possible to connect to a system as another user with no previous knowledge of the password. Authorization: If authorization information is held in a SQL database, it may be possible to change this information through the successful exploitation of a SQL injection vulnerability. Integrity: Just as it may be possible to read sensitive information, it is also possible to make changes or even delete this information with a SQL injection attack.
Potential Mitigations	Requirements specification: A non-SQL style database which is not subject to this flaw may be chosen. Design: Follow the principle of least privilege when creating user accounts to a SQL database. Users should only have the minimum privileges necessary to use their account. If the requirements of the system indicate that a user can read and modify their own data, then limit their privileges so they cannot read/write others' data. Design: Duplicate any filtering done on the client-side on the server side. Implementation: Implement SQL strings using prepared statements that bind variables. Prepared statements that do not bind variables can be vulnerable to attack. Implementation: Use vigorous white-list style checking on any user input that may be used in a SQL command. Rather than escape meta-characters, it is safest to disallow them entirely. Reason: Later use of data that have been entered in the database may neglect to escape meta-characters before use. Narrowly define the set of safe characters based on the expected value of the parameter in the request.
Demonstrative	The following code dynamically constructs and executes a

Examples

SQL query that searches for items matching a specified name. The query restricts the items displayed to those where owner matches the user name of the currently-authenticated user.

C# Example:

```
...
string userName = ctx.getAuthenticatedUserName();
string query = "SELECT * FROM items WHERE owner = '"
+ userName + "' AND itemname = '" + ItemName.Text +
"'";
sda = new SqlDataAdapter(query, conn);
DataTable dt = new DataTable();
sda.Fill(dt);
...
```

The query that this code intends to execute follows: SELECT * FROM items WHERE owner = <userName> AND itemname = <itemName>; However, because the query is constructed dynamically by concatenating a constant base query string and a user input string, the query only behaves correctly if itemName does not contain a single-quote character. If an attacker with the user name wiley enters the string "name' OR 'a'='a" for itemName, then the query becomes the following: SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name' OR 'a'='a'; The addition of the OR 'a'='a' condition causes the where clause to always evaluate to true, so the query becomes logically equivalent to the much simpler query: SELECT * FROM items; This simplification of the query allows the attacker to bypass the requirement that the query only return items owned by the authenticated user; the query now returns all entries stored in the items table, regardless of their specified owner.

This example examines the effects of a different malicious value passed to the query constructed and executed in the above example. If an attacker with the user name hacker enters the string "hacker'); DELETE FROM items; --" for itemName, then the query becomes the following two queries:

SQL Example:

```
SELECT * FROM items WHERE owner = 'wiley' AND
itemname = 'name';
DELETE FROM items;
```

--'

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database. Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [19]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in this way, the general attack could still be made effective using a trick similar to the one shown in Example 1. If an attacker enters the string "name'); DELETE FROM items; SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name'; DELETE FROM items; SELECT * FROM items WHERE 'a'='a'; One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, blacklisting is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers can: - Target fields that are not quoted - Find ways to bypass the need for certain escaped meta-characters - Use stored procedures to hide the injected meta-characters Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks. Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against many others. For example, the following PL/SQL procedure is vulnerable to the same SQL injection attack shown in the first example. procedure get_item (itm_cv IN OUT ItmCurTyp, usr in varchar2, itm in varchar2) is open itm_cv for ' SELECT * FROM items WHERE ' || 'owner = '|| usr || ' AND itemname = ' || itm || '; end get_item; Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL

injection attacks.

MS SQL has a built in function that enables shell command execution. An SQL injection in such a context could be disastrous. For example, a query of the form:

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='$user_input' ORDER BY PRICE

Where $user_input is taken from the user and unfiltered.

If the user provides the string:

' exec master..xp_cmdshell 'vol' --

The query will take the following form: "

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' exec master..xp_cmdshell 'vol' --' ORDER BY PRICE

Now, this query can be broken down into: [1] a first SQL query: SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' [2] a second SQL query, which executes a shell command: exec master..xp_cmdshell 'vol' [3] an MS SQL comment: --' ORDER BY PRICE As can be seen, the malicious input changes the semantics of the query into a query, a shell command execution and a comment.

Observed Examples

CVE-2004-0366 -

CVE-2004-0343 -

CVE-2003-0779 -

CVE-2003-0500 -

CVE-2003-0377 -

Context Notes

SQL injection has become a common issue with database-driven web sites. The flaw is easily detected, and easily exploited, and as such, any site or software package with even a minimal user base is likely to be subject to an attempted attack of this kind. Essentially, the attack is accomplished by placing a meta character into data input to then place SQL commands in the control plane, which did not exist there before. This flaw depends on the fact that SQL makes no real distinction between the control

	and data planes. If successful, SQL Injection attacks can give an attacker access to backend database contents, the ability to remotely execute system commands, or in some circumstances the means to take control of the Windows server hosting the database. Dynamically generating queries that include user input can lead to SQL injection attacks. An attacker can insert SQL commands or modifiers in the user input that can cause the query to behave in an unsafe manner. Constructing a dynamic SQL statement with user input may allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands. Factors: resultant to special character mismanagement, MAID, or blacklist/whitelist problems. Can be primary to authentication errors.
References	M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.
Node Relationships	ChildOf - Injection (74) ChildOf - Weaknesses in OWASP Top Ten (629) ChildOf - Weaknesses Examined by SAMATE (630) ChildOf - Weaknesses Used by NVD (635) ParentOf - SQL Injection: Hibernate (564)
Source Taxonomies	PLOVER - SQL injection 7 Pernicious Kingdoms - SQL Injection CLASP - SQL injection
Applicable Platforms	All

Path 1:

Query Name - **SQL_injection**

Severity - **Critical**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
```

```
212. private java.util.List getList(String array[]) //deletenodesservlet.java
      ...
      217. newList.add(array[i]);
      ...
      221. return newList;
```

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
```

```
...
105. java.util.List nodeDataList = getList(request.getParameterValues
("nodeData"));
...
130. select.append((String) nodeDataList.get(j));
...
138. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 2:

Query Name - **SQL_injection**

Severity - **Critical**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
74. String nodeId = request.getParameter("node");
...
89. intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
...
437. select.append(nodeId);
...
447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 3:

Query Name - **SQL_injection**

Severity - **Critical**

```
1. <!-- //nodelist.jsp
...
52. String column = request.getParameter( "column" );
...
63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
```

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
...
238. PreparedStatement stmt = conn.prepareStatement("SELECT ASSETS.NODEID,
NODE.NODELABEL, ASSETS." + columnName + " FROM ASSETS, NODE WHERE LOWER
(ASSETS." + columnName + ") LIKE ? AND ASSETS.NODEID=NODE.NODEID ORDER BY
NODE.NODELABEL");
...
241. ResultSet rs = stmt.executeQuery();
```

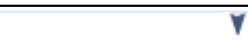
[Table of Contents] [Description]

Path 4:

Query Name - **SQL_injection**

Severity - **Critical**

```
1. <!-- //chooseinterface.jsp
      ...
      60. String nodeId = request.getParameter("node");
      ...
      71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
```



```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
      ...
      437. select.append(nodeId);
      ...
      447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

SQL Injection

CWE ID	89
Description	SQL injection attacks are another instantiation of injection attack, in which SQL commands are injected into data-plane input in order to effect the execution of predefined SQL commands.
Likelihood of Exploit	Very High
Common Consequences	Confidentiality: Since SQL databases generally hold sensitive data, loss of confidentiality is a frequent problem with SQL injection vulnerabilities. Authentication: If poor SQL commands are used to check user names and passwords, it may be possible to connect to a system as another user with no previous knowledge of the password. Authorization: If authorization information is held in a SQL database, it may be possible to change this information through the successful exploitation of a SQL injection vulnerability. Integrity: Just as it may be possible to read sensitive information, it is also possible to make changes or even delete this information with a SQL injection attack.
Potential Mitigations	Requirements specification: A non-SQL style database which is not subject to this flaw may be chosen. Design: Follow the principle of least privilege when creating user accounts to a SQL database. Users should only have the minimum privileges necessary to use their account. If the requirements of the system indicate that a user can read and modify their own data, then limit their privileges so they cannot read/write others' data. Design: Duplicate any filtering done on the client-side on the server side.

Implementation: Implement SQL strings using prepared statements that bind variables. Prepared statements that do not bind variables can be vulnerable to attack.

Implementation: Use vigorous white-list style checking on any user input that may be used in a SQL command. Rather than escape meta-characters, it is safest to disallow them entirely. Reason: Later use of data that have been entered in the database may neglect to escape meta-characters before use. Narrowly define the set of safe characters based on the expected value of the parameter in the request.

Demonstrative Examples

The following code dynamically constructs and executes a SQL query that searches for items matching a specified name. The query restricts the items displayed to those where owner matches the user name of the currently-authenticated user.

C# Example:

```
...
string userName = ctx.getAuthenticatedUserName();
string query = "SELECT * FROM items WHERE owner = '"
+ userName + "' AND itemname = '" + ItemName.Text +
"'";
sda = new SqlDataAdapter(query, conn);
DataTable dt = new DataTable();
sda.Fill(dt);
...
```

The query that this code intends to execute follows: SELECT * FROM items WHERE owner = <userName> AND itemname = <itemName>; However, because the query is constructed dynamically by concatenating a constant base query string and a user input string, the query only behaves correctly if itemName does not contain a single-quote character. If an attacker with the user name wiley enters the string "name' OR 'a'='a" for itemName, then the query becomes the following: SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name' OR 'a'='a'; The addition of the OR 'a'='a' condition causes the where clause to always evaluate to true, so the query becomes logically equivalent to the much simpler query: SELECT * FROM items; This simplification of the query allows the attacker to bypass the requirement that the query only return items owned by the authenticated user; the query now returns all entries stored in the items table, regardless of their specified owner.

This example examines the effects of a different malicious

value passed to the query constructed and executed in the above example. If an attacker with the user name hacker enters the string "hacker'); DELETE FROM items; --" for itemName, then the query becomes the following two queries:

SQL Example:

```
SELECT * FROM items WHERE owner = 'wiley' AND
itemname = 'name';
DELETE FROM items;
--'
```

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database. Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [19]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in this way, the general attack could still be made effective using a trick similar to the one shown in Example 1. If an attacker enters the string "name'); DELETE FROM items; SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name'; DELETE FROM items; SELECT * FROM items WHERE 'a'='a'; One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, blacklisting is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers can: - Target fields that are not quoted - Find ways to bypass the need for certain escaped meta-characters - Use stored procedures to hide the injected meta-characters Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks. Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against

many others. For example, the following PL/SQL procedure is vulnerable to the same SQL injection attack shown in the first example. procedure get_item (itm_cv IN OUT ItmCurTyp, usr in varchar2, itm in varchar2) is open itm_cv for ' SELECT * FROM items WHERE ' || 'owner = '|| usr || ' AND itemname = ' || itm || '; end get_item; Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL injection attacks.

MS SQL has a built in function that enables shell command execution. An SQL injection in such a context could be disastrous. For example, a query of the form:

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='$user_input' ORDER BY PRICE

Where $user_input is taken from the user and unfiltered.

If the user provides the string:

' exec master..xp_cmdshell 'vol' --

The query will take the following form: "

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' exec master..xp_cmdshell 'vol' --' ORDER BY PRICE

Now, this query can be broken down into: [1] a first SQL query: SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' [2] a second SQL query, which executes a shell command: exec master..xp_cmdshell 'vol' [3] an MS SQL comment: --' ORDER BY PRICE As can be seen, the malicious input changes the semantics of the query into a query, a shell command execution and a comment.

Observed Examples

CVE-2004-0366 -

CVE-2004-0343 -

CVE-2003-0779 -

	CVE-2003-0500 -
	CVE-2003-0377 -
Context Notes	SQL injection has become a common issue with database-driven web sites. The flaw is easily detected, and easily exploited, and as such, any site or software package with even a minimal user base is likely to be subject to an attempted attack of this kind. Essentially, the attack is accomplished by placing a meta character into data input to then place SQL commands in the control plane, which did not exist there before. This flaw depends on the fact that SQL makes no real distinction between the control and data planes. If successful, SQL Injection attacks can give an attacker access to backend database contents, the ability to remotely execute system commands, or in some circumstances the means to take control of the Windows server hosting the database. Dynamically generating queries that include user input can lead to SQL injection attacks. An attacker can insert SQL commands or modifiers in the user input that can cause the query to behave in an unsafe manner. Constructing a dynamic SQL statement with user input may allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands. Factors: resultant to special character mismanagement, MAID, or blacklist/whitelist problems. Can be primary to authentication errors.
References	M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.
Node Relationships	ChildOf - Injection (74) ChildOf - Weaknesses in OWASP Top Ten (629) ChildOf - Weaknesses Examined by SAMATE (630) ChildOf - Weaknesses Used by NVD (635) ParentOf - SQL Injection: Hibernate (564)
Source Taxonomies	PLOVER - SQL injection 7 Pernicious Kingdoms - SQL Injection CLASP - SQL injection
Applicable Platforms	All

Path 1:

Query Name - **Second_Order_SQL_Injection**

Severity - **Critical**

```
484. public void readTables(Reader reader) throws Exception //installer.java
        |
        ...
```

```
502. while ((line = r.readLine()) != null) {
...
512. m_seqmapping.put(m.group(1), a);
...
577. ((LinkedList) m_inserts.get(table)).add(line);
```

```
480. public void readTables() throws Exception //installer.java
481. readTables(new FileReader(m_create_sql));
```

```
177. public void install(String[] argv) throws Exception //installer.java
...
210. readTables();
...
249. insertData();
```

```
1178. void insertData() throws Exception //installer.java
...
1189. for (Iterator j = ((LinkedList) m_inserts.get(table)).iterator(); j
...
1192. st.execute((String) j.next());
```

[Table of Contents] [Description]

Path 2:

Query Name - **Second_Order_SQL_Injection**

Severity - **Critical**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1401. while ((line = r.readLine()) != null) {
1402. line = line.trim();
...
1411. create.append(line);
1412. create.append("\n");
...
1421. create.toString());
...
1427. String function = m.group(1);
...
1434. st.execute("DROP FUNCTION " + function + "(" + columns
```

[Table of Contents] [Description]

Path 3:

Query Name - **Second_Order_SQL_Injection**

Severity - **Critical**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1401. while ((line = r.readLine()) != null) {
1402. line = line.trim();
...
1411. create.append(line);
1412. create.append("\n");
...
```

```
1436. st.execute(create.toString());
```

[Table of Contents] [Description]

Path 4:

Query Name - **Second_Order_SQL_Injection**

Severity - **Critical**

```
1378. public void addStoredProcedures() throws Exception //installer.java
...
1401. while ((line = r.readLine()) != null) {
1402. line = line.trim();
...
1411. create.append(line);
1412. create.append("\n");
...
1442. st.execute(create.toString());
```

[Table of Contents] [Description]

Cross-site Scripting (XSS)

CWE ID	79
Description	A cross-site scripting weakness occurs when dynamically generated web pages display unvalidated, unfiltered, and unencoded user input allowing an attacker to embed malicious scripts into the generated page. This can be leveraged to execute scripting code as if it came from the site's server on the computer of anyone who uses the site.
Alternate Terms	"CSS" was once used as the acronym for this problem, but this can cause confusion with the "Cascading Style Sheets," so this acronym has declined significantly. Its use is discouraged by CWE.
Likelihood of Exploit	High to Very High
Weakness Ordinality	Resultant (Weakness is typically related to the presence of some other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Common Consequences	Confidentiality: The most common attack performed with cross-site scripting involves the disclosure of information stored in user cookies. Access control: In some circumstances it may be possible to run arbitrary code on a victim's computer when cross-site scripting is combined with other flaws. If successful, cross-site scripting vulnerabilities can be exploited to manipulate or steal cookies, create requests that can be mistaken for those of a valid user, compromise confidential information, or execute malicious code on the end user systems for a variety of nefarious purposes.
Potential Mitigations	Carefully check each input parameter against a rigorous positive specification (white list) defining the specific characters and format allowed. All input should be

sanitized, not just parameters that the user is supposed to specify, but all data in the request, including hidden fields, cookies, headers, the URL itself, and so forth. A common mistake that leads to continuing XSS vulnerabilities is to validate only fields that are expected to be redisplayed by the site. We often encounter data from the request that is reflected by the application server or the application that the development team did not anticipate. Also, a field that is not currently reflected may be used by a future developer. Therefore, validating ALL parts of the HTTP request is recommended.

This involves "HTML Entity Encoding" all non-alphanumeric characters from data that was received from the user and is now being written to the request.

With Struts, you should write all data from form beans with the bean's filter attribute set to true.

To help mitigate XSS attacks against the user's session cookie, set the session cookie to be HttpOnly. In browsers that support the HttpOnly feature (such as Internet Explorer), this attribute prevents the user's session cookie from being accessed by client-side scripts, including scripts inserted due to a XSS attack.

Demonstrative Examples

The following JSP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

JSP Example:

```
<% String eid = request.getParameter("eid"); %>
...
Employee ID: <%= eid %>
```

The following ASP.NET code segment reads an employee ID number from an HTTP request and displays it to the user.

ASP.NET Example:

```
protected System.Web.UI.WebControls.TextBox Login;
protected System.Web.UI.WebControls.Label EmployeeID;
  ...
EmployeeID.Text = Login.Text;
```

The code in this example operates correctly if eid contains only standard alphanumeric text. If eid has a value that includes meta-characters or source code, then the code will be executed by the web browser as it displays the

HTTP response. Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use e-mail or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

The following JSP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

JSP Example:

```
<%...
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("select * from emp
where id="+eid);
if (rs != null) {
rs.next();
String name = rs.getString("name");
%>

Employee Name: <%= name %>
```

The following ASP.NET code segment queries a database for an employee with a given employee ID and prints the name corresponding with the ID.

ASP.NET Example:

```
protected System.Web.UI.WebControls.Label
EmployeeName;
  ...
string query = "select * from emp where id=" + eid;
sda = new SqlDataAdapter(query, conn);
sda.Fill(dt);
string name = dt.Rows[0]["Name"];
  ...
EmployeeName.Text = name;
```

This code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if

	they are not. Again, this code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker can execute malicious commands in the user's web browser. This type of exploit, known as Stored XSS, is particularly insidious because the indirection caused by the data store makes it more difficult to identify the threat and increases the possibility that the attack will affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code. As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim: * As in the previous example, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or e-mailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that may include session information, from the user's machine to the attacker or perform other nefarious activities. * As in this example, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Stored XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. * A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.
Context Notes	Cross-site scripting (XSS) vulnerabilities occur when an

attacker uses a web application to send malicious code, generally JavaScript, to a different end user. When a web application uses input from a user in the output it generates without filtering it, an attacker can insert an attack in that input and the web application sends the attack to other users. The end user trusts the web application, and the attacks exploit that trust to do things that would not normally be allowed. Attackers frequently use a variety of methods to encode the malicious portion of the tag, such as using Unicode, so the request looks less suspicious to the user. XSS attacks can generally be categorized into two categories: stored and reflected. Stored attacks are those where the injected code is permanently stored on the target servers in a database, message forum, visitor log, and so forth. Reflected attacks are those where the injected code takes another route to the victim, such as in an email message, or on some other server. When a user is tricked into clicking a link or submitting a form, the injected code travels to the vulnerable web server, which reflects the attack back to the user's browser. The browser then executes the code because it came from a 'trusted' server. For a reflected XSS attack to work, the victim must submit the attack to the server. This is still a very dangerous attack given the number of possible ways to trick a victim into submitting such a malicious request, including clicking a link on a malicious Web site, in an email, or in an inner-office posting. XSS flaws are very likely in web applications, as they require a great deal of developer discipline to avoid them in most applications. It is relatively easy for an attacker to find XSS vulnerabilities. Some of these vulnerabilities can be found using scanners, and some exist in older web application servers. The consequence of an XSS attack is the same regardless of whether it is stored or reflected. The difference is in how the payload arrives at the server. XSS can cause a variety of problems for the end user that range in severity from an annoyance to complete account compromise. The most severe XSS attacks involve disclosure of the user's session cookie, which allows an attacker to hijack the user's session and take over their account. Other damaging attacks include the disclosure of end user files, installation of Trojan horse programs, redirecting the user to some other page or site, and modifying presentation of content. Cross-site scripting (XSS) vulnerabilities occur when: 1. Data enters a Web application through an untrusted source, most frequently a web request. 2. The data is included in dynamic content that is sent to a web user without being validated for malicious code. The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like cookies or other session information to the attacker, redirecting the victim

to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Cross-site scripting attacks can occur wherever an untrusted user has the ability to publish content to a trusted web site. Typically, a malicious user will craft a client-side script, which -- when parsed by a web browser -- performs some activity (such as sending all site cookies to a given E-mail address). If the input is unchecked, this script will be loaded and run by each user visiting the web site. Since the site requesting to run the script has access to the cookies in question, the malicious script does also. There are several other possible attacks, such as running "Active X" controls (under Microsoft Internet Explorer) from sites that a user perceives as trustworthy; cookie theft is however by far the most common. All of these attacks are easily prevented by ensuring that no script tags -- or for good measure, HTML tags at all -- are allowed in data to be posted publicly.

Cross-site scripting attacks may occur anywhere that possibly malicious users are allowed to post unregulated material to a trusted web site for the consumption of other valid users. The most common example can be found in bulletin-board web sites which provide web based mailing list-style functionality.

When used in conjunction with Cross-Site Request Forgery (CSRF), XSS worms can cause significant damage in web sites with large user populations.

References

Jeremiah Grossman, Robert "RSnake" Hansen, Petko "pdp" D. Petkov, Anton Rager and Seth Fogie. "XSS Attacks". Syngress. 2007.

M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.

Node Relationships

ChildOf - Injection (74)
ResultsIn - Mobile Code: Invoking Untrusted Mobile Code (494)
Peer - Cross-Site Request Forgery (CSRF) (352)
ChildOf - Weaknesses in OWASP Top Ten (629)
ChildOf - Weaknesses Used by NVD (635)
ParentOf - Basic XSS (80)
ParentOf - XSS in Error Pages (81)
ParentOf - Script in IMG Tags (82)
ParentOf - XSS using Script in Attributes (83)
ParentOf - XSS using Script Via Encoded URI Schemes (84)
ParentOf - Doubled Character XSS Manipulations (85)
ParentOf - Invalid Characters in Identifiers (86)
ParentOf - Alternate XSS Syntax (87)

Source Taxonomies

PLOVER - Cross-site scripting (XSS)

7 Pernicious Kingdoms - Cross-site Scripting

Applicable Platforms All

Path 1:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
      214. String name1 = request.getParameter("name1");
      ...
      225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
      282. response.sendRedirect("error.jsp?error=1&name=" + name);
```

[Table of Contents] [Description]

Path 2:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
      215. String protoArray1 = request.getParameter("protArray1");
      ...
      225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
      ...
      319. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
      ".port ");
```

[Table of Contents] [Description]

Path 3:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
      215. String protoArray1 = request.getParameter("protArray1");
      ...
      225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
    ...
    357. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
    ".capsd-class ");
```

[Table of Contents] [Description]

Path 4:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    214. String name1 = request.getParameter("name1");
    ...
    221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
    ...
    371. response.sendRedirect("error.jsp?error=1&name=" + name);
```

[Table of Contents] [Description]

Path 5:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    215. String protoArray1 = request.getParameter("protArray1");
    ...
    221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
    ...
    399. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
    ".monitor ");
```

[Table of Contents] [Description]

Path 6:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    215. String protoArray1 = request.getParameter("protArray1");
    ...
    221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
    ...
    437. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
    ".port");
```

[Table of Contents] [Description]

Path 7:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java
    ...
    248. String redirectPage = request.getParameter("nextPage");
    249. redirectString.append(redirectPage);
    ...
    252. redirectString.append("?").append(Util.makeQueryString(request,
    ignores));
    ...
    308. response.sendRedirect(redirectString.toString());
```

[Table of Contents] [Description]

Path 8:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
60. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //servicenoticeupdateservlet.java
    ...
    64. String checkedServices[] = request.getParameterValues("serviceCheck");
    ...
    68. System.out.println(checkedServices[i]);
```

[Table of Contents] [Description]

Path 9:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
52. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //updatenotifdstatusservlet.java
    ...
    54. System.out.println("status = " + request.getParameter("status"));
```

[Table of Contents] [Description]

Path 10:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
    ...
180. String redirectPage = request.getParameter("nextPage");
...
198. redirectString.append(redirectPage).append(makeQueryString(params));
...
257. response.sendRedirect(redirectString.toString());
```

[Table of Contents] [Description]

Path 11:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
89. StringBuffer buffer = request.getRequestURL();
...
101. buffer.setLength(schemeHostPortIndex);
...
108. buffer.append(context);
...
112. buffer.append("/");
...
114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
    ...
281. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=1&name=" + name);
```

[Table of Contents] [Description]

Path 12:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
89. StringBuffer buffer = request.getRequestURL();
...
101. buffer.setLength(schemeHostPortIndex);
...
108. buffer.append(context);
...
112. buffer.append("/");
```

```
...
114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java

    ...
    316. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".port ");
```

[Table of Contents] [Description]

Path 13:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java

    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java

    ...
    353. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".capsd-class ");
```

[Table of Contents] [Description]

Path 14:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java

    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
```

```
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    367. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=1&name=" + name);
```

[Table of Contents] [Description]

Path 15:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    392. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".monitor ");
```

[Table of Contents] [Description]

Path 16:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    432. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".port");
```

[Table of Contents] [Description]

Path 17:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
    257. String redirectPage = request.getParameter("redirect");
    ...
    261. response.sendRedirect(redirectPage);
```

[Table of Contents] [Description]

Path 18:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcpostservlet.java
    ...
    70. String pathInfo = request.getPathInfo();
    ...
    82. pathInfo = pathInfo.substring(1, pathInfo.length());
    ...
    87. String categoryName = Util.decode(pathInfo);
```

▼

```
157. public static String decode(String string) //util.java
    158. return URLDecoder.decode(string);
```

▼

```
68. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rtcpostservlet.java
    ...
    87. String categoryName = Util.decode(pathInfo);
    ...
    118. response.sendError(HttpServletResponse.SC_BAD_REQUEST, "No category info
    found for " + categoryName);
```

[Table of Contents] [Description]

Path 19:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
65. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noderescanservlet.java
    ...
    68. String returnUrl = request.getParameter("returnUrl");
    ...
    92. response.sendRedirect(request.getContextPath() + "/" + returnUrl);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
125. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventbyfilterservlet.java
     126. String redirectValue = request.getParameter("redirect");
     ...
     129. return (redirectValue);
```

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
     ...
     115. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 21:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
120. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventservlet.java
     121. String redirectValue = request.getParameter("redirect");
     ...
     124. return (redirectValue);
```

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
     ...
     110. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 22:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
74. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //nodelabelchangeservlet.java
     75. String nodeIdString = request.getParameter("node");
     ...
     104. response.sendRedirect(request.getContextPath() + "/element/node.jsp?
     node=" + nodeIdString);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
106. protected String getRedirectString(HttpServletRequest
```

```
request) //acknowledgenoticeservlet.java
    107. String redirectValue = request.getParameter("redirect");
    ...
    110. return (redirectValue);
```



```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
    ...
    96. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 24:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```



```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    126. String queryString = Util.makeQueryString(request);
```

[Table of Contents] [Description]

Path 25:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```



```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
```

```
120. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 26:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    101. buffer.setLength(schemeHostPortIndex);
    ...
    108. buffer.append(context);
    ...
    112. buffer.append("/");
    ...
    114. return buffer.toString();
```

▼

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    125. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
    queryString);
```

[Table of Contents] [Description]

Path 27:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //error.jsp
    ...
    49. name = request.getParameter( "name" );
    ...
    86. An error has occured due to a missing parameter <%= name %> in the poller
    configuration file
```

[Table of Contents] [Description]

Path 28:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //error.jsp
    ...
    49. name = request.getParameter( "name" );
    ...
    90. An error has occured since the <%= name %> poller already exists
```

[Table of Contents] [Description]

Path 29:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //interfacedeleted.jsp
       ...
       46. String ipAddr = request.getParameter( "intf" );
       ...
       70. <title><%=ipAddr%> | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 30:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //interfacedeleted.jsp
       ...
       46. String ipAddr = request.getParameter( "intf" );
       ...
       99. <h3>Finished Deleting Interface <%= ipAddr %></h3>
```

[Table of Contents] [Description]

Path 31:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //newinterface.jsp
       ...
       111. <h3> The IP address <%=request.getParameter("ipAddress")%> already
       exists. Please type in a different IP address.</h3>
```

[Table of Contents] [Description]

Path 32:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //servicedeleted.jsp
       ...
       44. String ipAddr = request.getParameter( "intf" );
       ...
       83. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
       Web Console</title>
```

[Table of Contents] [Description]

Path 33:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //servicedeleted.jsp
       ...
       44. String ipAddr = request.getParameter( "intf" );
       ...
       114. <h3>Finished Deleting <%=service_db.getServiceName()%> Service on <%
       =ipAddr%>.</h3>
```

[Table of Contents] [Description]

Path 34:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //snmpselect.jsp
       ...
       59. String nodeLabel = request.getParameter( "nodelabel" );
       ...
       176. <%=listNodeName(nodeId, nodeLabel)%>
```

▼

```
232. public String listNodeName(int intnodeid, String nodelabel) //snmpselect.jsp
       ...
       242. nodename.append(nodelabel);
       243. nodename.append("</td></tr>\n");
       244. nodename.append("<tr><td> </td></tr>\n");
       ...
       246. return nodename.toString();
```

▼

```
1. <!-- //snmpselect.jsp
       ...
       176. <%=listNodeName(nodeId, nodeLabel)%>
```

[Table of Contents] [Description]

Path 35:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
       ...
       304. Enumeration names = request.getParameterNames();
       ...
       307. String name = (String) names.nextElement();
       308. String[] values = request.getParameterValues(name);
       ...
       315. buffer.append(values[i]);
       316. buffer.append("\" />");
       317. buffer.append("\n");
       ...
       345. return (buffer.toString());
```

▼

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
       271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

```
1. <!-- //choosecommands.jsp
     ...
     123. <%=Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 36:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
     ...
     304. Enumeration names = request.getParameterNames();
     ...
     307. String name = (String) names.nextElement();
     308. String[] values = request.getParameterValues(name);
     ...
     315. buffer.append(values[i]);
     316. buffer.append("\" />");
     317. buffer.append("\n");
     ...
     345. return (buffer.toString());
```

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
     271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

```
1. <!-- //choosetargets.jsp
     ...
     182. <%=Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 37:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //escalations.jsp
     ...
     64. pathName = request.getParameter("path");
     ...
     104. <h3><%=(pathName!=null ? "Editing path " + pathName + "<br>" : "")%></h3>
```

[Table of Contents] [Description]

Path 38:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
      ...
      304. Enumeration names = request.getParameterNames();
      ...
      307. String name = (String) names.nextElement();
      308. String[] values = request.getParameterValues(name);
      ...
      315. buffer.append(values[i]);
      316. buffer.append("\" />");
      317. buffer.append("\n");
      ...
      345. return (buffer.toString());
```

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
      271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
      219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

```
1. <!-- //escalations.jsp
      ...
      107. <%=Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 39:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
      ...
      304. Enumeration names = request.getParameterNames();
      ...
      307. String name = (String) names.nextElement();
      308. String[] values = request.getParameterValues(name);
      ...
      315. buffer.append(values[i]);
      316. buffer.append("\" />");
      317. buffer.append("\n");
      ...
      345. return (buffer.toString());
```

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
      271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```



```
1. <!-- //groupintervals.jsp
     ...
     96. <%=Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 40:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //validaterule.jsp
     ...
     57. String newRule = (String)request.getParameter("newRule");
     ...
     116. <td align="left"> <%=newRule%>
```

[Table of Contents] [Description]

Path 41:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
     ...
     304. Enumeration names = request.getParameterNames();
     ...
     307. String name = (String) names.nextElement();
     308. String[] values = request.getParameterValues(name);
     ...
     315. buffer.append(values[i]);
     316. buffer.append("\" />");
     317. buffer.append("\n");
     ...
     345. return (buffer.toString());
```



```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
     271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```



```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```



```
1. <!-- //validaterule.jsp
     ...
     121. <%=Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 42:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //groupdetail.jsp
      ...
      45. String groupName = request.getParameter("groupName");
      ...
      50. group = groupFactory.getGroup(groupName);
```

```
172. public Group getGroup(String name) throws IOException, MarshalException,
ValidationException //groupmanager.java
      ...
      175. return (Group) m_groups.get(name);
```

```
1. <!-- //groupdetail.jsp
      ...
      50. group = groupFactory.getGroup(groupName);
      ...
      89. <h2>Details for Group: <%=group.getName()%></h2>
```

[Table of Contents] [Description]

Path 43:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //groupdetail.jsp
      ...
      45. String groupName = request.getParameter("groupName");
      ...
      50. group = groupFactory.getGroup(groupName);
```

```
172. public Group getGroup(String name) throws IOException, MarshalException,
ValidationException //groupmanager.java
      ...
      175. return (Group) m_groups.get(name);
```

```
1. <!-- //groupdetail.jsp
      ...
      50. group = groupFactory.getGroup(groupName);
      ...
      96. <%=group.getComments()%>
```

[Table of Contents] [Description]

Path 44:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //groupdetail.jsp
```

```
...
45. String groupName = request.getParameter("groupName");
...
50. group = groupFactory.getGroup(groupName);
```



```
172. public Group getGroup(String name) throws IOException, MarshalException,
ValidationException //groupmanager.java

...
175. return (Group) m_groups.get(name);
```



```
1. <!-- //groupdetail.jsp

...
50. group = groupFactory.getGroup(groupName);
...
109. <% Collection users = group.getUserCollection();
...
122. <% Iterator usersIter = (Iterator)users.iterator();
...
126. <%=(String)usersIter.next()%>
```

[Table of Contents] [Description]

Path 45:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //newuser.jsp

...
100. <h3>The user <%=request.getParameter("userID")%> already exists. Please
type in a different user id.</h3>
```

[Table of Contents] [Description]

Path 46:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp

...
46. String userID = request.getParameter("userID");
...
50. user = userFactory.getUser(userID);
```



```
227. public User getUser(String name) throws IOException, MarshalException,
ValidationException //usermanager.java

...
231. return (User) m_users.get(name);
```



```
1. <!-- //userdetail.jsp

...
50. user = userFactory.getUser(userID);
...
```

```
88. <h2>Details for User: <%=user.getUserId()%></h2>
```

[Table of Contents] [Description]

Path 47:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      50. user = userFactory.getUser(userID);
```

▼

```
227. public User getUser(String name) throws IOException, MarshalException,
ValidationException //usermanager.java
      ...
      231. return (User) m_users.get(name);
```

▼

```
1. <!-- //userdetail.jsp
      ...
      50. user = userFactory.getUser(userID);
      ...
      95. <%=user.getFullName()%>
```

[Table of Contents] [Description]

Path 48:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      50. user = userFactory.getUser(userID);
```

▼

```
227. public User getUser(String name) throws IOException, MarshalException,
ValidationException //usermanager.java
      ...
      231. return (User) m_users.get(name);
```

▼

```
1. <!-- //userdetail.jsp
      ...
      50. user = userFactory.getUser(userID);
      ...
      104. <%=user.getUserComments()%>
```

[Table of Contents] [Description]

Path 49:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      129. <%=userFactory.getEmail(userID)%>
```

```
270. public String getEmail(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
      ...
      274. User user = (User) m_users.get(userid);
      ...
      278. Enumeration contacts = user.enumerateContact();
      ...
      280. Contact contact = (Contact) contacts.nextElement();
      ...
      283. value = contact.getInfo();
      ...
      288. return value;
```

```
1. <!-- //userdetail.jsp
      ...
      129. <%=userFactory.getEmail(userID)%>
```

[Table of Contents] [Description]

Path 50:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      138. <%=userFactory.getPagerEmail(userID)%>
```

```
298. public String getPagerEmail(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
      ...
      302. User user = (User) m_users.get(userid);
      ...
      306. Enumeration contacts = user.enumerateContact();
      ...
      308. Contact contact = (Contact) contacts.nextElement();
      ...
      311. value = contact.getInfo();
      ...
      316. return value;
```

```
1. <!-- //userdetail.jsp
      ...
      138. <%=userFactory.getPagerEmail(userID)%>
```

[Table of Contents] [Description]

Path 51:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      147. <%=userFactory.getXMPPAddress(userID)%>
```

```
327. public String getXMPPAddress(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
      ...
      331. User user = (User) m_users.get(userid);
      ...
      335. Enumeration contacts = user.enumerateContact();
      ...
      337. Contact contact = (Contact) contacts.nextElement();
      ...
      340. value = contact.getInfo();
      ...
      345. return value;
```

```
1. <!-- //userdetail.jsp
      ...
      147. <%=userFactory.getXMPPAddress(userID)%>
```

[Table of Contents] [Description]

Path 52:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
      ...
      46. String userID = request.getParameter("userID");
      ...
      156. <%=userFactory.getNumericPage(userID)%>
```

```
356. public String getNumericPage(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
      ...
      360. User user = (User) m_users.get(userid);
      ...
      364. Enumeration contacts = user.enumerateContact();
      ...
      366. Contact contact = (Contact) contacts.nextElement();
      ...
      369. value = contact.getServiceProvider();
      ...
      374. return value;
```

```
1. <!-- //userdetail.jsp
```

```
...
156. <%=userFactory.getNumericPage(userID)%>
```

[Table of Contents] [Description]

Path 53:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
...
46. String userID = request.getParameter("userID");
...
165. <%=userFactory.getNumericPin(userID)%>
```

▼

```
440. public String getNumericPin(String userid) throws IOException,
MarshalException, ValidationException //usermanager.java
...
444. User user = (User) m_users.get(userid);
...
448. Enumeration contacts = user.enumerateContact();
...
450. Contact contact = (Contact) contacts.nextElement();
...
453. value = contact.getInfo();
...
458. return value;
```

▼

```
1. <!-- //userdetail.jsp
...
165. <%=userFactory.getNumericPin(userID)%>
```

[Table of Contents] [Description]

Path 54:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
...
46. String userID = request.getParameter("userID");
...
174. <%=userFactory.getTextPage(userID)%>
```

▼

```
412. public String getTextPage(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
...
416. User user = (User) m_users.get(userid);
...
420. Enumeration contacts = user.enumerateContact();
...
422. Contact contact = (Contact) contacts.nextElement();
...
425. value = contact.getServiceProvider();
```

```
...
430. return value;
```

```
1. <!-- //userdetail.jsp
...
174. <%=userFactory.getTextPage(userID)%>
```

[Table of Contents] [Description]

Path 55:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //userdetail.jsp
...
46. String userID = request.getParameter("userID");
...
183. <%=userFactory.getTextPin(userID)%>
```

```
384. public String getTextPin(String userid) throws IOException, MarshalException,
ValidationException //usermanager.java
...
388. User user = (User) m_users.get(userid);
...
392. Enumeration contacts = user.enumerateContact();
...
394. Contact contact = (Contact) contacts.nextElement();
...
397. value = contact.getInfo();
...
402. return value;
```

```
1. <!-- //userdetail.jsp
...
183. <%=userFactory.getTextPin(userID)%>
```

[Table of Contents] [Description]

Path 56:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //viewdetail.jsp
...
45. String viewName = request.getParameter("viewName");
...
50. view = viewFactory.getView(viewName);
```

```
432. public View getView(String name) //viewfactory.java
433. return (View) m_views.get(name);
```

```
1. <!-- //viewdetail.jsp
      ...
      50. view = viewFactory.getView(viewName);
      ...
      90. <%= view.getName() %></h2>
```

[Table of Contents] [Description]

Path 57:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //viewdetail.jsp
      ...
      45. String viewName = request.getParameter("viewName");
      ...
      50. view = viewFactory.getView(viewName);
```

```
432. public View getView(String name) //viewfactory.java
      433. return (View) m_views.get(name);
```

```
1. <!-- //viewdetail.jsp
      ...
      50. view = viewFactory.getView(viewName);
      ...
      98. <% if(view != null){ String title = view.getTitle(); %>
      99. <%= (title != null || title.equals(""))? "":title %>
```

[Table of Contents] [Description]

Path 58:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //viewdetail.jsp
      ...
      45. String viewName = request.getParameter("viewName");
      ...
      50. view = viewFactory.getView(viewName);
```

```
432. public View getView(String name) //viewfactory.java
      433. return (View) m_views.get(name);
```

```
1. <!-- //viewdetail.jsp
      ...
      50. view = viewFactory.getView(viewName);
      ...
      108. <% if(view != null){ String comment = view.getComment(); %>
      109. <%= (comment != null || comment.equals(""))? "":comment %>
```

[Table of Contents] [Description]

Path 59:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //viewdetail.jsp
      ...
      45. String viewName = request.getParameter("viewName");
      ...
      50. view = viewFactory.getView(viewName);
```

```
432. public View getView(String name) //viewfactory.java
      433. return (View) m_views.get(name);
```

```
1. <!-- //viewdetail.jsp
      ...
      50. view = viewFactory.getView(viewName);
      ...
      118. <% if(view != null){ Common common = view.getCommon(); %>
      119. <%= (common != null || common.getRule().equals(""))? "":common.getRule()
      %>
```

[Table of Contents] [Description]

Path 60:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //nodelist.jsp
      ...
      52. String column = request.getParameter( "column" );
      ...
      63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
```

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
      ...
      250. asset.columnSearched = columnName;
      ...
      252. vector.addElement(asset);
      ...
      264. assets[i] = (MatchingAsset) vector.elementAt(i);
      ...
      267. return (assets);
```

```
1. <!-- //nodelist.jsp
      ...
      63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
      ...
      104. <td><a href="asset/modify.jsp?node=<%=assets[i].nodeId%>"><%=assets
      [i].nodeLabel%></a></td>
```

[Table of Contents] [Description]

Path 61:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //nodelist.jsp
      ...
      52. String column = request.getParameter( "column" );
      ...
      63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
```

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
      ...
      250. asset.columnSearched = columnName;
      ...
      252. vector.addElement(asset);
      ...
      264. assets[i] = (MatchingAsset) vector.elementAt(i);
      ...
      267. return (assets);
```

```
1. <!-- //nodelist.jsp
      ...
      63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
      ...
      105. <td><%=assets[i].matchingValue%></td>
```

[Table of Contents] [Description]

Path 62:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //interface.jsp
      ...
      93. String ipAddr = request.getParameter( "intf" );
      ...
      160. <title><%=ipAddr%> | Interface | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 63:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //interface.jsp
      ...
      94. String ifindexString = request.getParameter( "ifindex" );
      ...
      202. <input type="hidden" name="ifindex" value="<%=(ifindexString == null ?
      "" : ifindexString)%>">
```

[Table of Contents] [Description]

Path 64:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //rescan.jsp
      ...
      45. String ipAddr = request.getParameter("ipaddr");
      ...
      110. (<%=ipAddr%>) interface?
```

[Table of Contents] [Description]

Path 65:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
      ...
      89. StringBuffer buffer = request.getRequestURL();
      ...
      101. buffer.setLength(schemeHostPortIndex);
      ...
      108. buffer.append(context);
      ...
      112. buffer.append("/");
      ...
      114. return buffer.toString();
```

▼

```
1. <!-- //rescan.jsp
      ...
      120. <input type="button" value="Cancel" onClick="window.open('<%
      =Util.calculateUrlBase(request) + "/" + returnUrl%>', '_self')" />
```

[Table of Contents] [Description]

Path 66:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //service.jsp
      ...
      45. String ipAddr = request.getParameter( "intf" );
      ...
      84. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
      Web Console</title>
```

[Table of Contents] [Description]

Path 67:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //detail.jsp
```

```
...
208. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 68:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
```

```
...
304. Enumeration names = request.getParameterNames();
...
307. String name = (String) names.nextElement();
308. String[] values = request.getParameterValues(name);
...
315. buffer.append(values[i]);
316. buffer.append("\" />");
317. buffer.append("\n");
...
345. return (buffer.toString());
```

▼

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
```

```
271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

▼

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
```

```
219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

▼

```
1. <!-- //list.jsp
```

```
...
200. <%=org.opennms.web.Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 69:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
```

```
...
304. Enumeration names = request.getParameterNames();
...
307. String name = (String) names.nextElement();
308. String[] values = request.getParameterValues(name);
...
315. buffer.append(values[i]);
316. buffer.append("\" />");
317. buffer.append("\n");
...
345. return (buffer.toString());
```

▼

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
     271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

```
1. <!-- //list.jsp
     ...
     259. <%=org.opennms.web.Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 70:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //about.jsp
     ...
     106. <td><%=request.getHeader( "User-Agent" )%></td>
```

[Table of Contents] [Description]

Path 71:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //eventlist.jsp
     ...
     154. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
     request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 72:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //eventlist.jsp
     ...
     76. String header = request.getParameter( "header" );
     ...
     160. <td colspan="4"><b><%=header%></b></td>
```

[Table of Contents] [Description]

Path 73:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <%-- //header.jsp
      ...
      28. String title = request.getParameter( "title" );
      ...
      50. <b><%=title%></b>
```

[Table of Contents] [Description]

Path 74:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <%-- //header.jsp
      ...
      30. String[] breadcrumbs = request.getParameterValues( "breadcrumb" );
      ...
      78. <% for( int i = 0; i < breadcrumbs.length; i++ ) { %> &gt; <%=breadcrumbs
      [i]%> <% } %>
```

[Table of Contents] [Description]

Path 75:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //custom_graph1.jsp
      ...
      124. <input type="hidden" name="intf" value="<%=request.getParameter("intf")%
      >" >
```

[Table of Contents] [Description]

Path 76:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //custom_graph3.jsp
      ...
      59. int nodeId = Integer.parseInt(nodeIdString);
      ...
      171. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

▼

```
288. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //performancemodel.java
      ...
      305. descr = ifLabel + " (Not Currently Updated)";
      ...
      339. return descr;
```

▼

```
1. <!-- //custom_graph3.jsp
      ...
      171. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 77:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //form_proc_graph.jsp
      ...
      49. String action = request.getParameter("action");
      ...
      84. <h3> Action: <%=action%> </h3>
```

[Table of Contents] [Description]

Path 78:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //form_proc_report.jsp
      ...
      62. String action = request.getParameter("action");
      ...
      121. <h3> Action: <%=action%> </h3>
```

[Table of Contents] [Description]

Path 79:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //form_proc_view.jsp
      ...
      49. String report_action = request.getParameter("action");
      ...
      111. <h3>Action: <%=report_action%></h3>
```

[Table of Contents] [Description]

Path 80:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
292. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
      ...
      304. Enumeration names = request.getParameterNames();
      ...
      307. String name = (String) names.nextElement();
      308. String[] values = request.getParameterValues(name);
      ...
      315. buffer.append(values[i]);
      316. buffer.append("\" />");
      317. buffer.append("\n");
```

```
...
345. return (buffer.toString());
```

```
270. public static String makeHiddenTags(HttpServletRequest request, Map additions,
String[] ignores) //util.java
     271. return (makeHiddenTags(request, additions, ignores, IgnoreType.BOTH));
```

```
218. public static String makeHiddenTags(HttpServletRequest request) //util.java
     219. return (makeHiddenTags(request, new HashMap(), new String[0]));
```

```
1. <!-- //browser.jsp
     ...
     236. <%=org.opennms.web.Util.makeHiddenTags(request)%>
```

[Table of Contents] [Description]

Path 81:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //detail.jsp
     ...
     222. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
     request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 82:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //list.jsp
     ...
     49. String username = request.getParameter( "username" );
     ...
     122. <h3>Notifications for <%=username%> </h3>
```

[Table of Contents] [Description]

Path 83:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc2.jsp
     ...
     65. String intf = request.getParameter("intf");
     ...
     133. Node: <%=nodeLabel%>    Interface: <%
     =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 84:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc4.jsp
      ...
      57. String startHour = request.getParameter( "startHour" );
      ...
      105. startCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( startHour ));
      106. startCal.set( Calendar.MINUTE, 0 );
      107. startCal.set( Calendar.SECOND, 0 );
      108. startCal.set( Calendar.MILLISECOND, 0 );
      ...
      119. String start = Long.toString( startCal.getTime().getTime() );
      ...
      123. String startPretty = new Date( Long.parseLong( start )).toString();
      ...
      163. <b>From</b> <%=startPretty%> <br>
```

[Table of Contents] [Description]

Path 85:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc4.jsp
      ...
      62. String endHour = request.getParameter( "endHour" );
      ...
      114. endCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( endHour ));
      115. endCal.set( Calendar.MINUTE, 0 );
      116. endCal.set( Calendar.SECOND, 0 );
      117. endCal.set( Calendar.MILLISECOND, 0 );
      ...
      120. String end = Long.toString( endCal.getTime().getTime() );
      ...
      124. String endPretty = new Date( Long.parseLong( end )).toString();
      ...
      164. <b>To</b> <%=endPretty%>
```

[Table of Contents] [Description]

Path 86:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //chooseinterface.jsp
      ...
      58. String nodeId = request.getParameter("node");
      ...
      69. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
      70. Arrays.sort(intfs);
      ...
      147. <option value="<%=intfs[i]%>" <%=(i==0) ? "selected" : ""%>><%
      =this.model.getHumanReadableNameForIfLabel(Integer.parseInt(nodeId), intfs
      [i])%></option>
```

[Table of Contents] [Description]

Path 87:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //chooseinterface.jsp
       ...
       58. String nodeId = request.getParameter("node");
       ...
       69. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
       70. Arrays.sort(intfs);
       ...
       147. <option value="<%=intfs[i]%>" <%=(i==0) ? "selected" : ""%>><%
       =this.model.getHumanReadableNameForIfLabel(Integer.parseInt(nodeId), intfs
       [i])%></option>
```

[Table of Contents] [Description]

Path 88:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //choosereportanddate.jsp
       ...
       68. String intf = request.getParameter("intf");
       ...
       77. graphs = this.model.getQueries(nodeId, intf, includeNodeQueries);
       ...
       82. Arrays.sort(graphs);
       ...
       163. <option VALUE=<%=graphs[i].getName()%>><%=graphs[i].getTitle()%></option>
```

[Table of Contents] [Description]

Path 89:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //choosereportanddate.jsp
       ...
       68. String intf = request.getParameter("intf");
       ...
       77. graphs = this.model.getQueries(nodeId, intf, includeNodeQueries);
       ...
       82. Arrays.sort(graphs);
       ...
       163. <option VALUE=<%=graphs[i].getName()%>><%=graphs[i].getTitle()%></option>
```

[Table of Contents] [Description]

Path 90:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       207. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 91:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       247. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 92:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       233. reportList = reportList + "&reports=" + reports[i];
       ...
       247. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 93:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       251. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 94:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       233. reportList = reportList + "&reports=" + reports[i];
       ...
       251. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 95:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       255. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 96:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       233. reportList = reportList + "&reports=" + reports[i];
       ...
       255. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 97:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       259. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 98:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       233. reportList = reportList + "&reports=" + reports[i];
       ...
       259. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 99:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       78. String start = request.getParameter( "start" );
       ...
       156. Date startDate = new Date( Long.parseLong( start ));
       ...
       280. <b>From</b> <%=startDate%> <br>
```

[Table of Contents] [Description]

Path 100:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       79. String end = request.getParameter( "end" );
       ...
       157. Date endDate = new Date( Long.parseLong( end ));
       ...
       281. <b>To</b> <%=endDate%>
```

[Table of Contents] [Description]

Path 101:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
```

```
339. onclick="top.location = '/opennms/performance/results.jsp?
relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 102:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
62. String reports[] = request.getParameterValues( "reports" );
...
325. reportList = reportList + "&reports=" + reports[i];
...
339. onclick="top.location = '/opennms/performance/results.jsp?
relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 103:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
75. String intf = request.getParameter( "intf" );
...
343. onclick="top.location = '/opennms/performance/results.jsp?
relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 104:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
62. String reports[] = request.getParameterValues( "reports" );
...
325. reportList = reportList + "&reports=" + reports[i];
...
343. onclick="top.location = '/opennms/performance/results.jsp?
relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 105:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       347. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 106:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       325. reportList = reportList + "&reports=" + reports[i];
       ...
       347. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 107:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       75. String intf = request.getParameter( "intf" );
       ...
       351. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 108:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       62. String reports[] = request.getParameterValues( "reports" );
       ...
       325. reportList = reportList + "&reports=" + reports[i];
       ...
       351. onclick="top.location = '/opennms/performance/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 109:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //zoom.jsp
      ...
      75. String intf = request.getParameter( "intf" );
      ...
      137. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 110:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //zoom.jsp
      ...
      78. String start = request.getParameter( "start" );
      ...
      82. Date startDate = new Date( Long.parseLong( start ));
      ...
      160. <b>From</b> <%=startDate%> <br>
```

[Table of Contents] [Description]

Path 111:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //zoom.jsp
      ...
      79. String end = request.getParameter( "end" );
      ...
      83. Date endDate = new Date( Long.parseLong( end ));
      ...
      161. <b>To</b> <%=endDate%>
```

[Table of Contents] [Description]

Path 112:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc2.jsp
      ...
      67. String intf = request.getParameter("intf");
      ...
      133. Node: <%=nodeLabel%>    Interface: <%
      =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
        ...
        312. descr = ifLabel;
        ...
        341. return descr;
```

```
1. <!-- //adhoc2.jsp
        ...
        133. Node: <%=nodeLabel%>    Interface: <%
        =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 113:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc4.jsp
        ...
        59. String startHour = request.getParameter( "startHour" );
        ...
        107. startCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( startHour ));
        108. startCal.set( Calendar.MINUTE, 0 );
        109. startCal.set( Calendar.SECOND, 0 );
        110. startCal.set( Calendar.MILLISECOND, 0 );
        ...
        121. String start = Long.toString( startCal.getTime().getTime() );
        ...
        125. String startPretty = new Date( Long.parseLong( start )).toString();
        ...
        165. <b>From</b> <%=startPretty%> <br>
```

[Table of Contents] [Description]

Path 114:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //adhoc4.jsp
        ...
        64. String endHour = request.getParameter( "endHour" );
        ...
        116. endCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( endHour ));
        117. endCal.set( Calendar.MINUTE, 0 );
        118. endCal.set( Calendar.SECOND, 0 );
        119. endCal.set( Calendar.MILLISECOND, 0 );
        ...
        122. String end = Long.toString( endCal.getTime().getTime() );
        ...
        126. String endPretty = new Date( Long.parseLong( end )).toString();
        ...
        166. <b>To</b> <%=endPretty%>
```

[Table of Contents] [Description]

Path 115:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
       ...
       312. descr = ifLabel;
       ...
       341. return descr;
```

```
1. <!-- //results.jsp
       ...
       212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 116:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       252. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 117:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       238. reportList = reportList + "&reports=" + reports[i];
       ...
       252. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 118:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       256. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 119:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       238. reportList = reportList + "&reports=" + reports[i];
       ...
       256. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 120:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       260. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 121:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       238. reportList = reportList + "&reports=" + reports[i];
       ...
       260. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 122:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       264. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 123:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       238. reportList = reportList + "&reports=" + reports[i];
       ...
       264. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 124:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       83. String start = request.getParameter( "start" );
       ...
       161. Date startDate = new Date( Long.parseLong( start ));
       ...
       285. <b>From</b> <%=startDate%> <br>
```

[Table of Contents] [Description]

Path 125:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       84. String end = request.getParameter( "end" );
       ...
```

```
162. Date endDate = new Date( Long.parseLong( end ));
...
286. <b>To</b> <%=endDate%>
```

[Table of Contents] [Description]

Path 126:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
77. String intf = request.getParameter( "intf" );
...
344. onclick="top.location = '/opennms/response/results.jsp?
relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 127:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
64. String reports[] = request.getParameterValues( "reports" );
...
330. reportList = reportList + "&reports=" + reports[i];
...
344. onclick="top.location = '/opennms/response/results.jsp?
relativetime=lastday&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 128:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
77. String intf = request.getParameter( "intf" );
...
348. onclick="top.location = '/opennms/response/results.jsp?
relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
></input><br>
```

[Table of Contents] [Description]

Path 129:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       330. reportList = reportList + "&reports=" + reports[i];
       ...
       348. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastweek&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 130:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       352. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 131:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       330. reportList = reportList + "&reports=" + reports[i];
       ...
       352. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastmonth&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 132:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       77. String intf = request.getParameter( "intf" );
       ...
       356. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 133:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
       ...
       64. String reports[] = request.getParameterValues( "reports" );
       ...
       330. reportList = reportList + "&reports=" + reports[i];
       ...
       356. onclick="top.location = '/opennms/response/results.jsp?
       relativetime=lastyear&intf=<%=intf%>&node=<%=nodeId%><%=reportList%>'"
       ></input><br>
```

[Table of Contents] [Description]

Path 134:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //category.jsp
       ...
       66. String categoryName = request.getParameter("category");
       ...
       72. Category category = this.model.getCategory(categoryName);
```

```
94. public Category getCategory(String categoryName) //categorymodel.java
       ...
       99. return (Category) this.categories.get(categoryName);
```

```
1. <!-- //category.jsp
       ...
       72. Category category = this.model.getCategory(categoryName);
       ...
       103. <title> <%=category.getName()%> | Category | SLM | OpenNMS Web
       Console</title>
```

[Table of Contents] [Description]

Path 135:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //category.jsp
       ...
       66. String categoryName = request.getParameter("category");
       ...
       72. Category category = this.model.getCategory(categoryName);
```

```
94. public Category getCategory(String categoryName) //categorymodel.java
       ...
       99. return (Category) this.categories.get(categoryName);
```

```
1. <!-- //category.jsp
       ...
       72. Category category = this.model.getCategory(categoryName);
       ...
       126. <h3><%=category.getName()%></h3>
```

[Table of Contents] [Description]

Path 136:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //category.jsp
       ...
       66. String categoryName = request.getParameter("category");
       ...
       72. Category category = this.model.getCategory(categoryName);
```

```
94. public Category getCategory(String categoryName) //categorymodel.java
       ...
       99. return (Category) this.categories.get(categoryName);
```

```
1. <!-- //category.jsp
       ...
       72. Category category = this.model.getCategory(categoryName);
       ...
       128. <p><%=category.getComment()%></p>
```

[Table of Contents] [Description]

Path 137:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
       ...
       218. String name1 = request.getParameter("name1");
       ...
       229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
       ...
       281. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
       error=1&name=" + name);
```

[Table of Contents] [Description]

Path 138:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    218. String name1 = request.getParameter("name1");
    ...
    225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
    request);
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    367. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=1&name=" + name);
```

[Table of Contents] [Description]

Path 139:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    219. String protoArray1 = request.getParameter("protArray1");
    ...
    229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
    ...
    316. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".port ");
```

[Table of Contents] [Description]

Path 140:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    219. String protoArray1 = request.getParameter("protArray1");
    ...
    229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
```

```
...
353. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".capsd-class ");
```

[Table of Contents] [Description]

Path 141:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
219. String protoArray1 = request.getParameter("protArray1");
...
225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
request);
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
392. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".monitor ");
```

[Table of Contents] [Description]

Path 142:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
219. String protoArray1 = request.getParameter("protArray1");
...
225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
request);
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
432. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".port");
```

[Table of Contents] [Description]

Path 143:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
125. protected String getRedirectString(HttpServletRequest
```

```
request) //acknowledgeeventbyfilterservlet.java
    ...
133. String redirectParms = request.getParameter("redirectParms");
...
138. buffer.append(redirectParms);
139. redirectValue = buffer.toString();
...
142. return (redirectValue);
```

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
    ...
115. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 144:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
120. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventservlet.java
    ...
128. String redirectParms = request.getParameter("redirectParms");
...
133. buffer.append(redirectParms);
134. redirectValue = buffer.toString();
...
137. return (redirectValue);
```

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
    ...
110. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 145:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
106. protected String getRedirectString(HttpServletRequest
request) //acknowledgenoticeservlet.java
    ...
114. String redirectParms = request.getParameter("redirectParms");
...
119. buffer.append(redirectParms);
120. redirectValue = buffer.toString();
...
123. return (redirectValue);
```

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
    ...
```

```
96. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 146:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
70. String endUrl = request.getParameter("endUrl");
...
106. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 147:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
70. String endUrl = request.getParameter("endUrl");
...
120. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 148:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
75. String endUrl = request.getParameter("endUrl");
...
125. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 149:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //rescan.jsp
...
44. String nodeIdString = request.getParameter("node");
```

```
...
56. returnUrl = "element/node.jsp?node=" + nodeIdString;
...
120. <input type="button" value="Cancel" onClick="window.open('<%
=Util.calculateUrlBase(request) + "/" + returnUrl%>', '_self')" />
```

[Table of Contents] [Description]

Path 150:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //rescan.jsp
...
45. String ipAddr = request.getParameter("ipaddr");
...
59. returnUrl = "element/interface.jsp?node=" + nodeIdString + "&intf=" +
ipAddr;
...
120. <input type="button" value="Cancel" onClick="window.open('<%
=Util.calculateUrlBase(request) + "/" + returnUrl%>', '_self')" />
```

[Table of Contents] [Description]

Path 151:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //detail.jsp
...
208. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 152:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //eventlist.jsp
...
154. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 153:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //detail.jsp
...
222. <input type="hidden" name="redirect" value="<%=request.getContextPath() +
```

```
request.getServletPath() + "?" + request.getQueryString()%>" />
```

[Table of Contents] [Description]

Path 154:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
118. String startHour = request.getParameter( "startHour" );
...
136. startCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( startHour ));
137. startCal.set( Calendar.MINUTE, 0 );
138. startCal.set( Calendar.SECOND, 0 );
139. startCal.set( Calendar.MILLISECOND, 0 );
...
150. start = Long.toString( startCal.getTime().getTime() );
...
156. Date startDate = new Date( Long.parseLong( start ));
...
280. <b>From</b> <%=startDate%> <br>
```

[Table of Contents] [Description]

Path 155:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
123. String endHour = request.getParameter( "endHour" );
...
145. endCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( endHour ));
146. endCal.set( Calendar.MINUTE, 0 );
147. endCal.set( Calendar.SECOND, 0 );
148. endCal.set( Calendar.MILLISECOND, 0 );
...
151. end = Long.toString( endCal.getTime().getTime() );
...
157. Date endDate = new Date( Long.parseLong( end ));
...
281. <b>To</b> <%=endDate%>
```

[Table of Contents] [Description]

Path 156:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
123. String startHour = request.getParameter( "startHour" );
...
141. startCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( startHour ));
142. startCal.set( Calendar.MINUTE, 0 );
143. startCal.set( Calendar.SECOND, 0 );
```

```
144. startCal.set( Calendar.MILLISECOND, 0 );
...
155. start = Long.toString( startCal.getTime().getTime() );
...
161. Date startDate = new Date( Long.parseLong( start ));
...
285. <b>From</b> <%=startDate%> <br>
```

[Table of Contents] [Description]

Path 157:

Query Name - **Reflected_XSS_All_Clients**

Severity - **Critical**

```
1. <!-- //results.jsp
...
128. String endHour = request.getParameter( "endHour" );
...
150. endCal.set( Calendar.HOUR_OF_DAY, Integer.parseInt( endHour ));
151. endCal.set( Calendar.MINUTE, 0 );
152. endCal.set( Calendar.SECOND, 0 );
153. endCal.set( Calendar.MILLISECOND, 0 );
...
156. end = Long.toString( endCal.getTime().getTime() );
...
162. Date endDate = new Date( Long.parseLong( end ));
...
286. <b>To</b> <%=endDate%>
```

[Table of Contents] [Description]

Cross-site Scripting (XSS)

CWE ID	79
Description	A cross-site scripting weakness occurs when dynamically generated web pages display unvalidated, unfiltered, and unencoded user input allowing an attacker to embed malicious scripts into the generated page. This can be leveraged to execute scripting code as if it came from the site's server on the computer of anyone who uses the site.
Alternate Terms	"CSS" was once used as the acronym for this problem, but this can cause confusion with the "Cascading Style Sheets," so this acronym has declined significantly. Its use is discouraged by CWE.
Likelihood of Exploit	High to Very High
Weakness Ordinality	Resultant (Weakness is typically related to the presence of some other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Common Consequences	Confidentiality: The most common attack performed with cross-site scripting involves the disclosure of information stored in user cookies. Access control: In some circumstances it may be possible to run arbitrary code on a victim's computer when cross-site scripting is combined with other flaws.

If successful, cross-site scripting vulnerabilities can be exploited to manipulate or steal cookies, create requests that can be mistaken for those of a valid user, compromise confidential information, or execute malicious code on the end user systems for a variety of nefarious purposes.

Potential Mitigations

Carefully check each input parameter against a rigorous positive specification (white list) defining the specific characters and format allowed. All input should be sanitized, not just parameters that the user is supposed to specify, but all data in the request, including hidden fields, cookies, headers, the URL itself, and so forth. A common mistake that leads to continuing XSS vulnerabilities is to validate only fields that are expected to be redisplayed by the site. We often encounter data from the request that is reflected by the application server or the application that the development team did not anticipate. Also, a field that is not currently reflected may be used by a future developer. Therefore, validating ALL parts of the HTTP request is recommended.

This involves "HTML Entity Encoding" all non-alphanumeric characters from data that was received from the user and is now being written to the request.

With Struts, you should write all data from form beans with the bean's filter attribute set to true.

To help mitigate XSS attacks against the user's session cookie, set the session cookie to be HttpOnly. In browsers that support the HttpOnly feature (such as Internet Explorer), this attribute prevents the user's session cookie from being accessed by client-side scripts, including scripts inserted due to a XSS attack.

Demonstrative Examples

The following JSP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

JSP Example:

```
<% String eid = request.getParameter("eid"); %>
...
Employee ID: <%= eid %>
```

The following ASP.NET code segment reads an employee ID number from an HTTP request and displays it to the user.

ASP.NET Example:

```
protected System.Web.UI.WebControls.TextBox Login;
protected System.Web.UI.WebControls.Label EmployeeID;
```

```
  ...
EmployeeID.Text = Login.Text;
```

The code in this example operates correctly if eid contains only standard alphanumeric text. If eid has a value that includes meta-characters or source code, then the code will be executed by the web browser as it displays the HTTP response. Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use e-mail or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

The following JSP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

JSP Example:

```
<%...
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("select * from emp
where id="+eid);
if (rs != null) {
rs.next();
String name = rs.getString("name");
%>

Employee Name: <%= name %>
```

The following ASP.NET code segment queries a database for an employee with a given employee ID and prints the name corresponding with the ID.

ASP.NET Example:

```
protected System.Web.UI.WebControls.Label
EmployeeName;
  ...
string query = "select * from emp where id=" + eid;
sda = new SqlDataAdapter(query, conn);
```

```
sda.Fill(dt);
string name = dt.Rows[0]["Name"];
  ...
EmployeeName.Text = name;
```

This code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if they are not. Again, this code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker can execute malicious commands in the user's web browser. This type of exploit, known as Stored XSS, is particularly insidious because the indirection caused by the data store makes it more difficult to identify the threat and increases the possibility that the attack will affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code. As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim: * As in the previous example, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or e-mailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that may include session information, from the user's machine to the attacker or perform other nefarious activities. * As in this example, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Stored XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is

valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. * A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.

Context Notes

Cross-site scripting (XSS) vulnerabilities occur when an attacker uses a web application to send malicious code, generally JavaScript, to a different end user. When a web application uses input from a user in the output it generates without filtering it, an attacker can insert an attack in that input and the web application sends the attack to other users. The end user trusts the web application, and the attacks exploit that trust to do things that would not normally be allowed. Attackers frequently use a variety of methods to encode the malicious portion of the tag, such as using Unicode, so the request looks less suspicious to the user. XSS attacks can generally be categorized into two categories: stored and reflected. Stored attacks are those where the injected code is permanently stored on the target servers in a database, message forum, visitor log, and so forth. Reflected attacks are those where the injected code takes another route to the victim, such as in an email message, or on some other server. When a user is tricked into clicking a link or submitting a form, the injected code travels to the vulnerable web server, which reflects the attack back to the user's browser. The browser then executes the code because it came from a 'trusted' server. For a reflected XSS attack to work, the victim must submit the attack to the server. This is still a very dangerous attack given the number of possible ways to trick a victim into submitting such a malicious request, including clicking a link on a malicious Web site, in an email, or in an inner-office posting. XSS flaws are very likely in web applications, as they require a great deal of developer discipline to avoid them in most applications. It is relatively easy for an attacker to find XSS vulnerabilities. Some of these vulnerabilities can be found using scanners, and some exist in older web application servers. The consequence of an XSS attack is the same regardless of whether it is stored or reflected. The difference is in how the payload arrives at the server. XSS can cause a variety of problems for the end user that range in severity from an annoyance to complete account compromise. The most severe XSS attacks involve disclosure of the user's session cookie, which allows an attacker to hijack the user's session and take over their account. Other damaging attacks include the disclosure of end user files, installation of Trojan horse programs, redirecting the user to some other page or site, and modifying presentation of content. Cross-site scripting (XSS) vulnerabilities occur when: 1. Data enters a Web application through an untrusted source, most frequently a

web request. 2. The data is included in dynamic content that is sent to a web user without being validated for malicious code. The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Cross-site scripting attacks can occur wherever an untrusted user has the ability to publish content to a trusted web site. Typically, a malicious user will craft a client-side script, which -- when parsed by a web browser -- performs some activity (such as sending all site cookies to a given E-mail address). If the input is unchecked, this script will be loaded and run by each user visiting the web site. Since the site requesting to run the script has access to the cookies in question, the malicious script does also. There are several other possible attacks, such as running "Active X" controls (under Microsoft Internet Explorer) from sites that a user perceives as trustworthy; cookie theft is however by far the most common. All of these attacks are easily prevented by ensuring that no script tags -- or for good measure, HTML tags at all -- are allowed in data to be posted publicly.

Cross-site scripting attacks may occur anywhere that possibly malicious users are allowed to post unregulated material to a trusted web site for the consumption of other valid users. The most common example can be found in bulletin-board web sites which provide web based mailing list-style functionality.

When used in conjunction with Cross-Site Request Forgery (CSRF), XSS worms can cause significant damage in web sites with large user populations.

References

Jeremiah Grossman, Robert "RSnake" Hansen, Petko "pdp" D. Petkov, Anton Rager and Seth Fogie. "XSS Attacks". Syngress. 2007.

M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.

Node Relationships

ChildOf - Injection (74)
ResultsIn - Mobile Code: Invoking Untrusted Mobile Code (494)
Peer - Cross-Site Request Forgery (CSRF) (352)
ChildOf - Weaknesses in OWASP Top Ten (629)
ChildOf - Weaknesses Used by NVD (635)
ParentOf - Basic XSS (80)
ParentOf - XSS in Error Pages (81)
ParentOf - Script in IMG Tags (82)

	ParentOf - XSS using Script in Attributes (83) ParentOf - XSS using Script Via Encoded URI Schemes (84) ParentOf - Doubled Character XSS Manipulations (85) ParentOf - Invalid Characters in Identifiers (86) ParentOf - Alternate XSS Syntax (87)
Source Taxonomies	PLOVER - Cross-site scripting (XSS) 7 Pernicious Kingdoms - Cross-site Scripting CLASP - Cross-site scripting
Applicable Platforms	All

Path 1:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      99. ResultSet rs = stmt.executeQuery();
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
268. public String buildTableRows(List nodes, int start, int stop) //delete.jsp
      ...
      277. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
      ());
      ...
      291. row.append(nodelabel);
      292. row.append("</td>\n");
      293. row.append("</tr>\n");
      ...
      296. return row.toString();
```

```
1. <!-- //delete.jsp
      ...
      215. <%=buildTableRows(nodes, 0, midNodeIndex)%>
```

[Table of Contents] [Description]

Path 2:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      99. ResultSet rs = stmt.executeQuery();
      ...
      102. label = rs.getString("NODELABEL");
      ...
```

```
111. return (label);
```

```
268. public String buildTableRows(List nodes, int start, int stop) //delete.jsp
     ...
     277. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
     ());
     ...
     291. row.append(nodelabel);
     292. row.append("</td>\n");
     293. row.append("</tr>\n");
     ...
     296. return row.toString();
```

```
1. <!-- //delete.jsp
     ...
     236. <%=buildTableRows(nodes, midNodeIndex, nodes.size())%>
```

[Table of Contents] [Description]

Path 3:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     279. ResultSet rs = stmt.executeQuery();
     ...
     283. nodeLabel = rs.getString(1);
     ...
     295. return new NodeLabel(nodeLabel, temp[0]);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
     185. m_nodeLabel = nodeLabel;
```

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     295. return new NodeLabel(nodeLabel, temp[0]);
```

```
1. <!-- //nodelabel.jsp
     ...
     67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
     ...
     102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
     for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
     (currentLabel.getSource()))%>)
```

```
194. public String getLabel() //nodelabel.java
     195. return m_nodeLabel;
```

```
1. <!-- //nodelabel.jsp
      ...
      102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
      for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
      (currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 4:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
      279. ResultSet rs = stmt.executeQuery();
      ...
      283. nodeLabel = rs.getString(1);
      ...
      295. return new NodeLabel(nodeLabel, temp[0]);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
      185. m_nodeLabel = nodeLabel;
```

▼

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
      295. return new NodeLabel(nodeLabel, temp[0]);
```

▼

```
1. <!-- //nodelabel.jsp
      ...
      67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
      ...
      102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
      for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
      (currentLabel.getSource()))%>)
```

▼

```
203. public char getSource() //nodelabel.java
      204. return m_nodeLabelSource;
```

▼

```
1. <!-- //nodelabel.jsp
      ...
      102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
      for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
      (currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 5:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    442. ResultSet rs = stmt.executeQuery();
    ...
    446. netbiosName = rs.getString(1);
    ...
    458. netbiosName = netbiosName.substring(0, MAX_NODE_LABEL_LENGTH);
    ...
    462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
    185. m_nodeLabel = nodeLabel;
```

▼

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
    463. return nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
    ...
    126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
    (autoLabel.getSource()))%>)</td>
```

▼

```
194. public String getLabel() //nodelabel.java
    195. return m_nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
    (autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 6:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    442. ResultSet rs = stmt.executeQuery();
    ...
    446. netbiosName = rs.getString(1);
    ...
    458. netbiosName = netbiosName.substring(0, MAX_NODE_LABEL_LENGTH);
    ...
    462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
```

```
185. m_nodeLabel = nodeLabel;
```

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
463. return nodeLabel;
```

```
1. <!-- //nodelabel.jsp
...
68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
...
126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
(autoLabel.getSource()))%>)</td>
```

```
203. public char getSource() //nodelabel.java
204. return m_nodeLabelSource;
```

```
1. <!-- //nodelabel.jsp
...
126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
(autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 7:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
170. public String buildTableRows(List nodes, int start, int stop) //snmpmanage.jsp
...
179. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
());
...
190. row.append(nodelabel);
191. row.append("\">");
192. row.append(nodelabel);
193. row.append("</a>");
194. row.append("</td>\n");
195. row.append("</tr>\n");
...
198. return row.toString();
```

```
1. <!-- //snmpmanage.jsp
```

```
...
134. <%=buildTableRows(nodes, 0, midNodeIndex)%>
```

[Table of Contents] [Description]

Path 8:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
170. public String buildTableRows(List nodes, int start, int stop) //snmpmanage.jsp
...
179. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
());
...
190. row.append(nodelabel);
191. row.append("\">");
192. row.append(nodelabel);
193. row.append("</a>");
194. row.append("</td>\n");
195. row.append("</tr>\n");
...
198. return row.toString();
```

```
1. <!-- //snmpmanage.jsp
...
153. <%=buildTableRows(nodes, midNodeIndex, nodes.size())%>
```

[Table of Contents] [Description]

Path 9:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //managenode.jsp
...
160. nodeLabel = NetworkElementFactory.getNodeLabel(firstInterface.getNodeid
```

```
());
...
195. <h2>Node: <%=nodeLabel%></h2>
```

[Table of Contents] [Description]

Path 10:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    99. ResultSet rs = stmt.executeQuery();
    ...
    102. label = rs.getString("NODELABEL");
    ...
    111. return (label);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
    ...
    269. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeID);
    ...
    272. " <td width=\"30%\">" + nodeLabel + "</td> " +
    ...
    297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
    ...
    203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 11:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    99. ResultSet rs = stmt.executeQuery();
    ...
    102. label = rs.getString("NODELABEL");
    ...
    111. return (label);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
    ...
    269. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeID);
```

```
...
272. " <td width=\"30%\">" + nodeLabel + "</td> " +
...
297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp

...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
...
224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 12:

Query Name - **Stored_XSS**

Severity - **Critical**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java

...
687. ResultSet rset = stmt.executeQuery("SELECT servicename FROM service");
...
691. services.add(rset.getString(1));
...
703. return services;
```

```
197. public String buildServiceOptions(String rule) //buildrule.jsp

...
200. List services = NotificationFactory.getInstance().getServiceNames();
...
209. buffer.append("<option selected VALUE='" + services.get(i) + "'>" +
services.get(i) + "</option>");
...
213. buffer.append("<option VALUE='" + services.get(i) + "'>" + services.get
(i) + "</option>");
...
217. return buffer.toString();
```

```
1. <!-- //buildrule.jsp

...
154. Services:<br><select size="10" multiple name="services"><%
=buildServiceOptions(newRule)%></select>
```

[Table of Contents] [Description]

Path 13:

Query Name - **Stored_XSS**

Severity - **Critical**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java


...
687. ResultSet rset = stmt.executeQuery("SELECT servicename FROM service");
...
691. services.add(rset.getString(1));
```

```
...
703. return services;
```

```
220. public String buildNotServiceOptions(String rule) //buildrule.jsp

     ...
     223. List services = NotificationFactory.getInstance().getServiceNames();
     ...
     234. buffer.append("<option selected VALUE='" + services.get(i) + "'>" +
     services.get(i) + "</option>");
     ...
     238. buffer.append("<option VALUE='" + services.get(i) + "'>" + services.get
     (i) + "</option>");
     ...
     242. return buffer.toString();
```

```
1. <!-- //buildrule.jsp

     ...
     160. "NOT" Services:<br><select size="10" multiple name="notServices"><%
     =buildNotServiceOptions(newRule)%></select>
```

[Table of Contents] [Description]

Path 14:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

     ...
     99. ResultSet rs = stmt.executeQuery();
     ...
     102. label = rs.getString("NODELABEL");
     ...
     111. return (label);
```

```
1. <!-- //index.jsp

     ...
     162. <li> <%=asset.getAssetNumber()%>: <a href="asset/modify.jsp?node=<%
     =asset.getNodeId()%>"><%=NetworkElementFactory.getNodeLabel(asset.getNodeId
     ())%></a>
```

[Table of Contents] [Description]

Path 15:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

     ...
     99. ResultSet rs = stmt.executeQuery();
     ...
     102. label = rs.getString("NODELABEL");
     ...
```

```
111. return (label);
```

```
1. <!-- //index.jsp
    ...
    169. <li> <%=asset.getAssetNumber()%>: <a href="asset/modify.jsp?node=<%
    =asset.getNodeId()%>"><%=NetworkElementFactory.getNodeLabel(asset.getNodeId
    ())%></a>
```

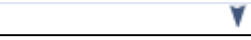
[Table of Contents] [Description]

Path 16:

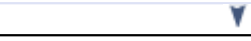
Query Name - **Stored_XSS**

Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
    ...
    833. ResultSet rs = stmt.executeQuery("SELECT SERVICEID, SERVICENAME FROM
    SERVICE");
    ...
    837. String name = rs.getString("SERVICENAME");
    ...
    840. nameMap.put(name, new Integer(id));
    ...
    850. serviceName2IdMap = nameMap;
```



```
818. public static Map getServiceNameToIdMap() throws
SQLException //networkelementfactory.java
    ...
    820. createServiceIdNameMaps();
    ...
    823. return (new HashMap(serviceName2IdMap));
```



```
1. <!-- //index.jsp
    ...
    53. Map serviceNameMap = new TreeMap
    (NetworkElementFactory.getServiceNameToIdMap());
    54. Set serviceNameSet = serviceNameMap.keySet();
    55. Iterator serviceNameIterator = serviceNameSet.iterator();
    ...
    112. <% String name = (String)serviceNameIterator.next(); %>
    113. <option value="<%=serviceNameMap.get(name)%>"><%=name%></option>
```

[Table of Contents] [Description]

Path 17:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    99. ResultSet rs = stmt.executeQuery();
    ...
```

```
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //interface.jsp

...
257. <td><a href="element/node.jsp?node=<%=intf_db.getNodeId()%>"><%
=NetworkElementFactory.getNodeLabel(intf_db.getNodeId())%></a></td>
```

[Table of Contents] [Description]

Path 18:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //rescan.jsp

...
52. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeId);
...
106. <p>Are you sure you want to rescan the <nobr><%=nodeLabel%></nobr>
```

[Table of Contents] [Description]

Path 19:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //service.jsp

...
154. <td><a href="element/node.jsp?node=<%=service_db.getNodeId()%>"><%
=NetworkElementFactory.getNodeLabel(service_db.getNodeId())%></a></td>
```

[Table of Contents] [Description]

Path 20:

Query Name - **Stored_XSS**

Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
      ...
      833. ResultSet rs = stmt.executeQuery("SELECT SERVICEID, SERVICENAME FROM
      SERVICE");
      ...
      837. String name = rs.getString("SERVICENAME");
      ...
      840. nameMap.put(name, new Integer(id));
      ...
      850. serviceName2IdMap = nameMap;
```

```
818. public static Map getServiceNameToIdMap() throws
SQLException //networkelementfactory.java
      ...
      820. createServiceIdNameMaps();
      ...
      823. return (new HashMap(serviceName2IdMap));
```

```
1. <!-- //advquerypanel.jsp
      ...
      42. Map serviceNameMap = new TreeMap
      (NetworkElementFactory.getServiceNameToIdMap());
      43. Set serviceNameSet = serviceNameMap.keySet();
      44. Iterator serviceNameIterator = serviceNameSet.iterator();
      ...
      102. <% String name = (String)serviceNameIterator.next(); %>
      103. <option value="<%=serviceNameMap.get(name)%>"><%=name%></option>
```

[Table of Contents] [Description]

Path 21:

Query Name - **Stored_XSS**

Severity - **Critical**

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      280. ResultSet rs = stmt.executeQuery();
      ...
      282. Event[] events = rs2Events(rs);
```

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      994. element = rs.getString("eventUei");
      995. event.uei = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    106. <h3>Event <%=event.getId()%></h3>
```

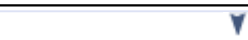
[Table of Contents] [Description]

Path 22:

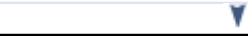
Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    994. element = rs.getString("eventUei");
    995. event.uei = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```



```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 23:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        994. element = rs.getString("eventUei");
        995. event.uei = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```



```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
```

[Table of Contents] [Description]

Path 24:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        994. element = rs.getString("eventUei");
        995. event.uei = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 25:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
994. element = rs.getString("eventUei");
995. event.uei = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
```

```
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 26:

Query Name - **Stored_XSS**

Severity - **Critical**

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    390. ResultSet rs = stmt.executeQuery(select.toString());
    ...
    392. events = rs2Events(rs);
```

▼

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    994. element = rs.getString("eventUei");
    995. event.uei = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 27:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
994. element = rs.getString("eventUei");
995. event.uei = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 28:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
994. element = rs.getString("eventUei");
995. event.uei = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 29:

Query Name - **Stored_XSS**

Severity - **Critical**

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java

...
263. ResultSet rs = stmt.executeQuery();
...
265. outages = rs2Outages(rs);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    438. element = rs.getString("ipaddr");
    439. outage.ipAddress = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
    ...
    265. outages = rs2Outages(rs);
    ...
    273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
    ...
    75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
    yesterday);
    ...
    97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
    [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
    [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 30:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    438. element = rs.getString("ipaddr");
    439. outage.ipAddress = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      98. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 31:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      438. element = rs.getString("ipaddr");
      439. outage.ipAddress = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
```

```
...
265. outages = rs2Outages(rs);
...
273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 32:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
...
438. element = rs.getString("ipaddr");
439. outage.ipAddress = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
...
265. outages = rs2Outages(rs);
...
273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

Path 33:

Query Name - **Stored_XSS**

Severity - **Critical**

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
    ...
    202. ResultSet rs = stmt.executeQuery();
    ...
    204. outages = rs2Outages(rs);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    438. element = rs.getString("ipaddr");
    439. outage.ipAddress = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
    ...
    204. outages = rs2Outages(rs);
    ...
    212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
    ...
    69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
```

```
...
94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 34:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
438. element = rs.getString("ipaddr");
439. outage.ipAddress = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

▼

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
```

```
...
204. outages = rs2Outages(rs);
...
212. return outages;
```

▼

```
1. <!-- //nodeoutages-box.jsp
```

```
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
95. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 35:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
```

```
SQLException //outagemodel.java
      ...
      438. element = rs.getString("ipaddr");
      439. outage.ipAddress = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      96. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 36:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      438. element = rs.getString("ipaddr");
      439. outage.ipAddress = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      101. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 37:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      438. element = rs.getString("ipaddr");
      439. outage.ipAddress = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      103. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
      [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 38:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      ...
      390. rs = stmt.executeQuery(select.toString());
      ...
      400. node.nodeLabel = rs.getString("nodeLabel");
      ...
      402. nodeList.add(node);
      ...
      414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
      [nodeList.size()]);
      ...
      417. return (nodes);
```

```
1. <!-- //performance-box.jsp
      ...
      61. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
      ...
      86. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 39:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
      ...
      391. rs = stmt.executeQuery(select.toString());
      ...
      402. node.nodeLabel = rs.getString("nodeLabel");
      ...
      404. nodeList.add(node);
      ...
      419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
      [nodeList.size()]);
      ...
      422. return (nodes);
```

```
1. <!-- //response-box.jsp
      ...
      65. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
```

```
...
90. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 40:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
438. element = rs.getString("ipaddr");
439. outage.ipAddress = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

▼

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
```

```
...
330. outages = rs2Outages(rs);
...
338. return outages;
```

▼

```
1. <!-- //serviceoutages-box.jsp
```

```
...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 41:

Query Name - **Stored_XSS**

Severity - **Critical**

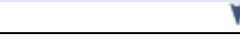
```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
438. element = rs.getString("ipaddr");
439. outage.ipAddress = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
     ...
     330. outages = rs2Outages(rs);
     ...
     338. return outages;
```



```
1. <!-- //serviceoutages-box.jsp
     ...
     83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
     serviceId, yesterday);
     ...
     103. <td <%=(outages[i].getRegainedServiceTime() == null) ?
     "bgcolor=\"red\"" : ""%>><%=(outages[i].getRegainedServiceTime() == null) ?
     "<b>DOWN</b>" : org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getRegainedServiceTime())%></td>
```

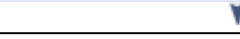
[Table of Contents] [Description]

Path 42:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     438. element = rs.getString("ipaddr");
     439. outage.ipAddress = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```



```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
    ...
    330. outages = rs2Outages(rs);
    ...
    338. return outages;
```



```
1. <!-- //serviceoutages-box.jsp
    ...
    83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
    serviceId, yesterday);
    ...
    104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
    [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 43:

Query Name - **Stored_XSS**

Severity - **Critical**

```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
    ...
    353. ResultSet rs = stmt.executeQuery("select distinct outages.nodeid, max
    (outages.iflostservice) as timeDown, node.nodelabel " + "from outages, node,
    ipinterface, ifservices " + "where ifregainedservice is null " + "and
    node.nodeid=outages.nodeid and ipinterface.nodeid = outages.nodeid and
    ifservices.nodeid=outages.nodeid " + "and ipinterface.ipaddr = outages.ipaddr
    and ifservices.ipaddr = outages.ipaddr " + "and ifservices.serviceid =
    outages.serviceid " + "and node.nodeType != 'D' and ipinterface.ismanaged !=
    'D' and ifservices.status != 'D' " + "group by outages.nodeid, node.nodelabel
    " + "order by timeDown desc;");
    ...
    362. String nodeLabel = rs.getString("nodelabel");
    ...
    364. list.add(new OutageSummary(nodeId, nodeLabel, downDate));
```



```
50. public OutageSummary(int nodeId, String nodeLabel, Date
timeDown) //outagesummary.java
    ...
    56. this.nodeLabel = nodeLabel;
```



```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
    ...
    364. list.add(new OutageSummary(nodeId, nodeLabel, downDate));
```

```
...
370. summaries = (OutageSummary[]) list.toArray(new OutageSummary[list.size
()]);
...
375. return summaries;
```

```
1. <!-- //servicesdown-box.jsp
...
53. OutageSummary[] summaries = this.model.getCurrentOutageSummaries();
...
63. <% OutageSummary summary = summaries[i];
64. String nodeLabel = summary.getNodeLabel();
```

```
69. public String getNodeLabel() //outagesummary.java
70. return (this.nodeLabel);
```

```
1. <!-- //servicesdown-box.jsp
...
64. String nodeLabel = summary.getNodeLabel();
...
68. <td align="left"><a href="element/node.jsp?node=<%=nodeId%>"><nobr><%
=nodeLabel%></nobr></a></td>
```

[Table of Contents] [Description]

Path 44:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
...
390. rs = stmt.executeQuery(select.toString());
...
400. node.nodeLabel = rs.getString("nodeLabel");
...
402. nodeList.add(node);
...
414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
[nodeList.size()]);
...
417. return (nodes);
```

```
1. <!-- //custom_graph1.jsp
...
56. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
...
132. <option value="<%=nodes[i].nodeId%>" SELECTED><%=nodes[i].nodeLabel%
></option>
```

[Table of Contents] [Description]

Path 45:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      ...
      390. rs = stmt.executeQuery(select.toString());
      ...
      400. node.nodeLabel = rs.getString("nodeLabel");
      ...
      402. nodeList.add(node);
      ...
      414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
      [nodeList.size()]);
      ...
      417. return (nodes);
```

```
1. <!-- //custom_graph1.jsp
      ...
      56. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
      ...
      134. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 46:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      99. ResultSet rs = stmt.executeQuery();
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //custom_graph3.jsp
      ...
      169. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 47:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      99. ResultSet rs = stmt.executeQuery();
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //custom_report.jsp
     ...
     159. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 48:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
     ...
     99. ResultSet rs = stmt.executeQuery();
     ...
     102. label = rs.getString("NODELABEL");
     ...
     111. return (label);
```

```
1. <!-- //custom_view.jsp
     ...
     208. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 49:

Query Name - **Stored_XSS**

Severity - **Critical**

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
     ...
     148. ResultSet rs = stmt.executeQuery("SELECT * FROM NODE WHERE NODETYPE !=
     'D' ORDER BY NODELABEL");
     ...
     748. service.m_ipAddr = rs.getString("ipaddr");
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
     ...
     614. node.m_nodeId = rs.getInt("nodeId");
     ...
     639. vector.addElement(node);
     ...
     645. nodes[i] = (Node) vector.elementAt(i);
     ...
     648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
     ...
     150. nodes = rs2Nodes(rs);
     ...
     158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 50:

Query Name - **Stored_XSS**

Severity - **Critical**

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      148. ResultSet rs = stmt.executeQuery("SELECT * FROM NODE WHERE NODETYPE !=
      'D' ORDER BY NODELABEL");
      ...
      748. service.m_ipAddr = rs.getString("ipaddr");
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      614. node.m_nodeId = rs.getInt("nodeId");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 51:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
```

```
115. ResultSet sentToResults = sentTo.executeQuery();
...
121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
    72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 52:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    115. ResultSet sentToResults = sentTo.executeQuery();
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
    72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
```

```
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 53:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    115. ResultSet sentToResults = sentTo.executeQuery();
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
    72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
    notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 54:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
    72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 55:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 56:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
    101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
    =notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 57:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
    101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 58:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      144. <a href="element/service.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>&service=<%=notices[i].getServiceId()%>"><%
      =notices[i].getServiceName()%></a>
```

[Table of Contents] [Description]

Path 59:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      146. <%=notices[i].getServiceName()%>
```

[Table of Contents] [Description]

Path 60:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      153. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeReplied()%> </td>
```

[Table of Contents] [Description]

Path 61:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      99. ResultSet rs = stmt.executeQuery();
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
285. public String getTextDescription() //noticefactory.java
      ...
      288. nodeName = NetworkElementFactory.getNodeLabel(this.nodeId);
      ...
      292. return (TYPE + "=" + nodeName);
```

```
1. <!-- //browser.jsp
      ...
      225. <li><strong><%=filter.getTextDescription()%></strong>
```

[Table of Contents] [Description]

Path 62:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      115. ResultSet sentToResults = sentTo.executeQuery();
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 63:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      115. ResultSet sentToResults = sentTo.executeQuery();
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
=notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 64:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
      (notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 65:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
      101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      127. newSentTo.setMedia(sentToResults.getString(MEDIA));
      ...
      129. sentToList.add(newSentTo);
```

```
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
113. <td width="17%"><%=notice.getTimeReplied()!=null ?
org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
" "%></td>
```

[Table of Contents] [Description]

Path 66:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
127. newSentTo.setMedia(sentToResults.getString(MEDIA));
...
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
" "%></td>
```

[Table of Contents] [Description]

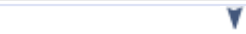
Path 67:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```



```
1. <!-- //detail.jsp
...
126. <a href="element/node.jsp?node=<%=notice.getNodeId()%>"><%
=NetworkElementFactory.getNodeLabel(notice.getNodeId())%></a>
```

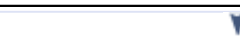
[Table of Contents] [Description]

Path 68:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```



```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
135. <a href="element/interface.jsp?node=<%=notice.getNodeId()%>&intf=<%
=notice.getIpAddress()%>"><%=notice.getIpAddress()%></a>
```

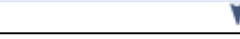
[Table of Contents] [Description]

Path 69:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```



```
1. <!-- //detail.jsp
...
157. <a href="outage/list?filter=node%3D<%=notice.getNodeId()%>">See outages
for <%=NetworkElementFactory.getNodeLabel(notice.getNodeId())%></a>
```

[Table of Contents] [Description]

Path 70:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
    ...
    240. Object element = rs.getString(TXT_MESG);
    241. nbean.m_txtMsg = (String) element;
    ...
    286. vector.addElement(nbean);
    ...
    297. notices[i] = (Notification) vector.elementAt(i);
    ...
    300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
    ...
    405. notices = rs2NotifyBean(conn, rs);
    ...
    417. return (notices);
```

```
1. <!-- //list.jsp
    ...
    58. Notification[] notices = this.model.getOutstandingNotices( username );
    ...
    150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
    =notices[i].getId()%></a></td>
    151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
    [i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 71:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
    ...
    240. Object element = rs.getString(TXT_MESG);
    241. nbean.m_txtMsg = (String) element;
    ...
    286. vector.addElement(nbean);
    ...
    297. notices[i] = (Notification) vector.elementAt(i);
    ...
    300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
    ...
    405. notices = rs2NotifyBean(conn, rs);
    ...
```

```
417. return (notices);
```

```
1. <!-- //list.jsp
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 72:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 73:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
...
```

```
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 74:

Query Name - **Stored_XSS**

Severity - **Critical**

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

...
252. ResultSet rs = stmt.executeQuery();
...
254. Outage[] outages = rs2Outages(rs);
```

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java

...
542. outage.outageId = rs.getInt("outageid");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp

...
64. Outage outage = OutageFactory.getOutage( outageId );
...
```

```
100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 75:

Query Name - **Stored_XSS**

Severity - **Critical**

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
252. ResultSet rs = stmt.executeQuery();
...
254. Outage[] outages = rs2Outages(rs);
```

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
542. outage.outageId = rs.getInt("outageid");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
=outage.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 76:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
```

```
...
111. return (label);
```

```
1. <!-- //results.jsp
...
205. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
=NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 77:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //zoom.jsp
...
135. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
=NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 78:

Query Name - **Stored_XSS**

Severity - **Critical**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
...
450. String ipAddr = rs.getString("ipaddr");
...
454. intfs.add(ipAddr);
...
463. return intfs;
```

```
425. public ArrayList getQueryableInterfacesForNode(int nodeId) throws
SQLException //responsetimemodel.java
426. return this.getQueryableInterfacesForNode(String.valueOf(nodeId));
```

```
1. <!-- //adhoc.jsp
...
67. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
...
71. intfMap.put(this.model.getHumanReadableNameForIfLabel(nodeId, (String)
intfs.get(i)), (String)intfs.get(i));
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
```

```
...
312. descr = ifLabel;
...
341. return descr;
```

```
1. <!-- //adhoc.jsp
```

```
...
71. intfMap.put(this.model.getHumanReadableNameForIfLabel(nodeId, (String)
intfs.get(i)), (String)intfs.get(i));
...
111. <% Iterator iterator = intfMap.keySet().iterator(); %>
...
113. <% String key = (String)iterator.next(); %>
114. <option value="<%=intfMap.get(key)%>"><%=key%></option>
```

[Table of Contents] [Description]

Path 79:

Query Name - **Stored_XSS**

Severity - **Critical**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
```

```
...
129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
...
142. return info;
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
```

```
...
169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
...
176. return info;
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
```

```
...
304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
...
318. parenString += (String) intfMap.get("ipaddr");
...
322. parenString += ((parenString.equals("")) ? speed : (", " + speed));
...
338. descr = descr + ((parenString.equals("")) ? "" : (" (" + parenString +
")"));
...
341. return descr;
```

```
1. <!-- //adhoc2.jsp
```

```
...
133. Node: <%=nodeLabel%>    Interface: <%
=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

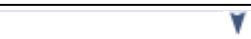
[Table of Contents] [Description]

Path 80:

Query Name - **Stored_XSS**

Severity - **Critical**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
    ...
    447. ResultSet rs = stmt.executeQuery(select.toString());
    ...
    450. String ipAddr = rs.getString("ipaddr");
    ...
    454. intfs.add(ipAddr);
    ...
    463. return intfs;
```



```
1. <!-- //chooseinterface.jsp
    ...
    71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
    ...
    148. <option value="<%=intfs.get(i)%>" <%=(i==0) ? "selected" : ""%>><%
    =intfs.get(i)%></option>
```

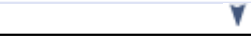
[Table of Contents] [Description]

Path 81:

Query Name - **Stored_XSS**

Severity - **Critical**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
    ...
    447. ResultSet rs = stmt.executeQuery(select.toString());
    ...
    450. String ipAddr = rs.getString("ipaddr");
    ...
    454. intfs.add(ipAddr);
    ...
    463. return intfs;
```



```
1. <!-- //chooseinterface.jsp
    ...
    71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
    ...
    148. <option value="<%=intfs.get(i)%>" <%=(i==0) ? "selected" : ""%>><%
    =intfs.get(i)%></option>
```

[Table of Contents] [Description]

Path 82:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
```

```
...
391. rs = stmt.executeQuery(select.toString());
...
402. node.nodeLabel = rs.getString("nodeLabel");
...
404. nodeList.add(node);
...
419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
[nodeList.size()]);
...
422. return (nodes);
```

```
1. <!-- //index.jsp
```

```
...
60. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
...
153. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

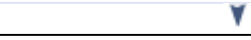
Path 83:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
```

```
...
391. rs = stmt.executeQuery(select.toString());
...
402. node.nodeLabel = rs.getString("nodeLabel");
...
404. nodeList.add(node);
...
419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
[nodeList.size()]);
...
422. return (nodes);
```



```
1. <!-- //index.jsp
```

```
...
60. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
...
169. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 84:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
99. ResultSet rs = stmt.executeQuery();
```

```
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```

```
1. <!-- //results.jsp
...
210. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
=NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 85:

Query Name - **Stored_XSS**

Severity - **Critical**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
...
129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
...
142. return info;
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
...
169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
...
176. return info;
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
...
304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
...
318. parenString += (String) intfMap.get("ipaddr");
...
322. parenString += ((parenString.equals("")) ? speed : (", " + speed));
...
338. descr = descr + ((parenString.equals("")) ? "" : (" (" + parenString +
")"));
...
341. return descr;
```

```
1. <!-- //results.jsp
...
212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 86:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
99. ResultSet rs = stmt.executeQuery();
...
102. label = rs.getString("NODELABEL");
...
111. return (label);
```



```
1. <!-- //category.jsp
```

```
...
84. String nodeLabel = NetworkElementFactory.getNodeLabel((int)node.getNodeid
());
...
89. nodeMap.put( nodeLabel, node );
...
93. nodeMap.put( "nodeId=" + node.getNodeid(), node );
...
97. Set keySet = nodeMap.keySet();
98. Iterator nameIterator = keySet.iterator();
...
142. String nodeLabel = (String)nameIterator.next();
...
160. <td><a href="element/node.jsp?node=<%=node.getNodeid()%>"><%=nodeLabel%
></a></td>
```

[Table of Contents] [Description]

Path 87:

Query Name - **Stored_XSS**

Severity - **Critical**

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
251. ResultSet rs = stmt.executeQuery();
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
```



```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
```

```
...
453. element = rs.getObject("nodeID");
...
455. vul.nodeId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```



```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
```

```
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
175. return (this.nodeLabel);
```

```
1. <!-- //detail.jsp
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

[Table of Contents] [Description]

Path 88:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
453. element = rs.getObject("nodeID");
...
455. vul.nodeId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
```

```
215. return (this.protocol);
```

```
1. <!-- //detail.jsp
     ...
   116. <%=vul.getProtocol()%>
```

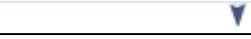
[Table of Contents] [Description]

Path 89:

Query Name - **Stored_XSS**

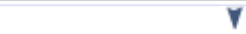
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
     ...
   453. element = rs.getObject("nodeID");
     ...
   455. vul.nodeId = (Integer) element;
     ...
   526. list.add(vul);
     ...
   529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
     ...
   531. return vuls;
```

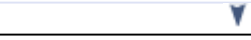


```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
     ...
   253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
     ...
   258. vul = vuls[0];
     ...
   267. return vul;
```

```
1. <!-- //detail.jsp
     ...
    64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
     ...
   124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```



```
143. public Date getCreateTime() //vulnerability.java
     144. return (this.createTime);
```



```
1. <!-- //detail.jsp
     ...
   124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 90:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      453. element = rs.getObject("nodeID");
      ...
      455. vul.nodeId = (Integer) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 91:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      453. element = rs.getObject("nodeID");
      ...
      455. vul.nodeId = (Integer) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 92:

Query Name - **Stored_XSS**

Severity - **Critical**

```
82. private static InetAddress lookupLocalInetAddress() throws RuntimeException,
UnknownHostException //simulatorcontroller.java
      ...
      110. String ip_out = p_output.readLine();
      111. ret = InetAddress.getByName(ip_out);
      ...
      131. return ret;
```

```
15. public SimulatorController(InetAddress controlChannel) throws
RuntimeException //simulatorcontroller.java
      ...
      19. System.out.println(this._localInetAddress = lookupLocalInetAddress());
```

[Table of Contents] [Description]

Path 93:

Query Name - **Stored_XSS**

Severity - **Critical**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java
      ...
      691. services.add(rset.getString(1));
      ...
```

```
703. return services;
```

```
197. public String buildServiceOptions(String rule) //buildrule.jsp
     ...
200. List services = NotificationFactory.getInstance().getServiceNames();
     ...
209. buffer.append("<option selected VALUE='" + services.get(i) + "'>" +
services.get(i) + "</option>");
     ...
213. buffer.append("<option VALUE='" + services.get(i) + "'>" + services.get
(i) + "</option>");
     ...
217. return buffer.toString();
```

```
1. <!-- //buildrule.jsp
     ...
154. Services:<br><select size="10" multiple name="services"><%
=buildServiceOptions(newRule)%></select>
```

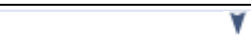
[Table of Contents] [Description]

Path 94:

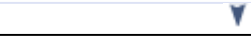
Query Name - **Stored_XSS**

Severity - **Critical**

```
680. public List getServiceNames() throws SQLException //notificationmanager.java
     ...
691. services.add(rset.getString(1));
     ...
703. return services;
```



```
220. public String buildNotServiceOptions(String rule) //buildrule.jsp
     ...
223. List services = NotificationFactory.getInstance().getServiceNames();
     ...
234. buffer.append("<option selected VALUE='" + services.get(i) + "'>" +
services.get(i) + "</option>");
     ...
238. buffer.append("<option VALUE='" + services.get(i) + "'>" + services.get
(i) + "</option>");
     ...
242. return buffer.toString();
```



```
1. <!-- //buildrule.jsp
     ...
160. "NOT" Services:<br><select size="10" multiple name="notServices"><%
=buildNotServiceOptions(newRule)%></select>
```

[Table of Contents] [Description]

Path 95:

Query Name - **Stored_XSS**

Severity - **Critical**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
    129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
    ...
    142. return info;
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
    ...
    169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
    ...
    176. return info;
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
    ...
    304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
    ...
    318. parenString += (String) intfMap.get("ipaddr");
    ...
    322. parenString += ((parenString.equals("")) ? speed : (", " + speed));
    ...
    338. descr = descr + ((parenString.equals("")) ? "" : (" (" + parenString +
    ")"));
    ...
    341. return descr;
```

```
1. <!-- //adhoc2.jsp
    ...
    133. Node: <%=nodeLabel%>    Interface: <%
    =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 96:

Query Name - **Stored_XSS**

Severity - **Critical**

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
    129. info.put(rs.getMetaData().getColumnName(i), rs.getString(i));
    ...
    142. return info;
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
    ...
    169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
    ...
    176. return info;
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
```

```
SQLException //responsetimemodel.java
    ...
304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
...
318. parenString += (String) intfMap.get("ipaddr");
...
322. parenString += ((parenString.equals("")) ? speed : (", " + speed));
...
338. descr = descr + ((parenString.equals("")) ? "" : (" (" + parenString +
")"));
...
341. return descr;
```

```
1. <!-- //results.jsp
    ...
212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

[Table of Contents] [Description]

Path 97:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
283. nodeLabel = rs.getString(1);
...
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
    185. m_nodeLabel = nodeLabel;
```

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
1. <!-- //nodelabel.jsp
    ...
67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

```
194. public String getLabel() //nodelabel.java
    195. return m_nodeLabel;
```

```
1. <!-- //nodelabel.jsp
    ...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
```

```
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 98:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    283. nodeLabel = rs.getString(1);
    ...
    295. return new NodeLabel(nodeLabel, temp[0]);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
    185. m_nodeLabel = nodeLabel;
```

▼

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    295. return new NodeLabel(nodeLabel, temp[0]);
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
    ...
    102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
    for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
    (currentLabel.getSource()))%>)
```

▼

```
203. public char getSource() //nodelabel.java
    204. return m_nodeLabelSource;
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
    for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
    (currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 99:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    284. nodeLabelSource = rs.getString(2);
```

```
...
294. char[] temp = nodeLabelSource.toCharArray();
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
...
186. m_nodeLabelSource = nodeLabelSource;
```

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
1. <!-- //nodelabel.jsp
...
67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

```
194. public String getLabel() //nodelabel.java
195. return m_nodeLabel;
```

```
1. <!-- //nodelabel.jsp
...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 100:

Query Name - **Stored_XSS**

Severity - **Critical**

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
284. nodeLabelSource = rs.getString(2);
...
294. char[] temp = nodeLabelSource.toCharArray();
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
...
186. m_nodeLabelSource = nodeLabelSource;
```

```
265. public static NodeLabel retrieveLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
```

```
...
295. return new NodeLabel(nodeLabel, temp[0]);
```

```
1. <!-- //nodelabel.jsp
...
67. NodeLabel currentLabel = NodeLabel.retrieveLabel( nodeId, myConn );
...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

```
203. public char getSource() //nodelabel.java
204. return m_nodeLabelSource;
```

```
1. <!-- //nodelabel.jsp
...
102. <p><a href="element/node.jsp?node=<%=nodeId%>" title="More information
for this node"><%=currentLabel.getLabel()%></a> (<%=typeMap.get(new Character
(currentLabel.getSource()))%>)
```

[Table of Contents] [Description]

Path 101:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
446. netbiosName = rs.getString(1);
...
458. netbiosName = netbiosName.substring(0, MAX_NODE_LABEL_LENGTH);
...
462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
185. m_nodeLabel = nodeLabel;
```

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
...
462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
463. return nodeLabel;
```

```
1. <!-- //nodelabel.jsp
...
68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
...
126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
(autoLabel.getSource()))%>)</td>
```

```
194. public String getLabel() //nodelabel.java
     195. return m_nodeLabel;
```

```
1. <!-- //nodelabel.jsp
     ...
     126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
     (autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 102:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     446. netbiosName = rs.getString(1);
     ...
     458. netbiosName = netbiosName.substring(0, MAX_NODE_LABEL_LENGTH);
     ...
     462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
     185. m_nodeLabel = nodeLabel;
```

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     462. NodeLabel nodeLabel = new NodeLabel(netbiosName, SOURCE_NETBIOS);
     463. return nodeLabel;
```

```
1. <!-- //nodelabel.jsp
     ...
     68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
     ...
     126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
     (autoLabel.getSource()))%>)</td>
```

```
203. public char getSource() //nodelabel.java
     204. return m_nodeLabelSource;
```

```
1. <!-- //nodelabel.jsp
     ...
     126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
     (autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 103:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    582. ResultSet rs = stmt.executeQuery();
    ...
    586. primarySysName = rs.getString(1);
    ...
    598. primarySysName = primarySysName.substring(0, MAX_NODE_LABEL_LENGTH);
    ...
    602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
    185. m_nodeLabel = nodeLabel;
```

▼

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
    603. return nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
    ...
    126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
    (autoLabel.getSource()))%>)</td>
```

▼

```
194. public String getLabel() //nodelabel.java
    195. return m_nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
    ...
    126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
    (autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 104:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
    ...
    582. ResultSet rs = stmt.executeQuery();
    ...
    586. primarySysName = rs.getString(1);
    ...
    598. primarySysName = primarySysName.substring(0, MAX_NODE_LABEL_LENGTH);
    ...
```

```
602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
     185. m_nodeLabel = nodeLabel;
```

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
     603. return nodeLabel;
```

```
1. <!-- //nodelabel.jsp
     ...
     68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
     ...
     126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
     (autoLabel.getSource()))%>)</td>
```

```
203. public char getSource() //nodelabel.java
     204. return m_nodeLabelSource;
```

```
1. <!-- //nodelabel.jsp
     ...
     126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
     (autoLabel.getSource()))%>)</td>
```

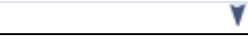
[Table of Contents] [Description]

Path 105:

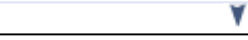
Query Name - **Stored_XSS**

Severity - **Critical**

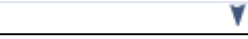
```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     586. primarySysName = rs.getString(1);
     ...
     598. primarySysName = primarySysName.substring(0, MAX_NODE_LABEL_LENGTH);
     ...
     602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
```



```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
     185. m_nodeLabel = nodeLabel;
```



```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
     ...
     602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
     603. return nodeLabel;
```



```
1. <!-- //nodelabel.jsp
      ...
      68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
      ...
      126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
      (autoLabel.getSource()))%>)</td>
```

▼

```
194. public String getLabel() //nodelabel.java
      195. return m_nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
      ...
      126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
      (autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 106:

Query Name - **Stored_XSS**

Severity - **Critical**

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
      586. primarySysName = rs.getString(1);
      ...
      598. primarySysName = primarySysName.substring(0, MAX_NODE_LABEL_LENGTH);
      ...
      602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
```

▼

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
      185. m_nodeLabel = nodeLabel;
```

▼

```
431. public static NodeLabel computeLabel(int nodeID, Connection dbConnection)
throws SQLException //nodelabel.java
      ...
      602. NodeLabel nodeLabel = new NodeLabel(primarySysName, SOURCE_SYSNAME);
      603. return nodeLabel;
```

▼

```
1. <!-- //nodelabel.jsp
      ...
      68. NodeLabel autoLabel = NodeLabel.computeLabel( nodeId, myConn );
      ...
      126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
      (autoLabel.getSource()))%>)</td>
```

▼

```
203. public char getSource() //nodelabel.java
      204. return m_nodeLabelSource;
```

▼

```
1. <!-- //nodelabel.jsp
```

```
...
126. <td><%=autoLabel.getLabel()%> (<%=typeMap.get(new Character
(autoLabel.getSource()))%>)</td>
```

[Table of Contents] [Description]

Path 107:

Query Name - **Stored_XSS**

Severity - **Critical**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
```

```
...
248. asset.nodeLabel = rs.getString("nodelabel");
...
252. vector.addElement(asset);
...
264. assets[i] = (MatchingAsset) vector.elementAt(i);
...
267. return (assets);
```

```
1. <!-- //nodelist.jsp
```

```
...
63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
...
104. <td><a href="asset/modify.jsp?node=<%=assets[i].nodeId%>"><%=assets
[i].nodeLabel%></a></td>
```

[Table of Contents] [Description]

Path 108:

Query Name - **Stored_XSS**

Severity - **Critical**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
```

```
...
248. asset.nodeLabel = rs.getString("nodelabel");
...
252. vector.addElement(asset);
...
264. assets[i] = (MatchingAsset) vector.elementAt(i);
...
267. return (assets);
```

```
1. <!-- //nodelist.jsp
```

```
...
63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
...
105. <td><%=assets[i].matchingValue%></td>
```

[Table of Contents] [Description]

Path 109:

Query Name - **Stored_XSS**

Severity - **Critical**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
     ...
     249. asset.matchingValue = rs.getString(columnName);
     ...
     252. vector.addElement(asset);
     ...
     264. assets[i] = (MatchingAsset) vector.elementAt(i);
     ...
     267. return (assets);
```

```
1. <!-- //nodelist.jsp
     ...
     63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
     ...
     104. <td><a href="asset/modify.jsp?node=<%=assets[i].nodeId%>"><%=assets
     [i].nodeLabel%></a></td>
```

[Table of Contents] [Description]

Path 110:

Query Name - **Stored_XSS**

Severity - **Critical**

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
     ...
     249. asset.matchingValue = rs.getString(columnName);
     ...
     252. vector.addElement(asset);
     ...
     264. assets[i] = (MatchingAsset) vector.elementAt(i);
     ...
     267. return (assets);
```

```
1. <!-- //nodelist.jsp
     ...
     63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
     ...
     105. <td><%=assets[i].matchingValue%></td>
```

[Table of Contents] [Description]

Path 111:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
     ...
     314. asset.setNotifyCategory(rs.getString("notifyCategory"));
```

```
178. public void setNotifyCategory(String notifyCategory) //asset.java
        ...
        180. this.notifyCategory = notifyCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
        ...
        314. asset.setNotifyCategory(rs.getString("notifyCategory"));
        ...
        322. vector.addElement(asset);
        ...
        328. assets[i] = (Asset) vector.elementAt(i);
        ...
        331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
        ...
        61. Asset[] assets = rs2Assets(rs);
        ...
        65. asset = assets[0];
        ...
        74. return (asset);
```

```
1. <!-- //modify.jsp
        ...
        57. Asset asset = this.model.getAsset( nodeId );
        ...
        262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
        =asset.getComments()%></textarea></td>
```

```
606. public String getComments() //asset.java
        607. return (this.comments);
```

```
1. <!-- //modify.jsp
        ...
        262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
        =asset.getComments()%></textarea></td>
```

[Table of Contents] [Description]

Path 112:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
        ...
        314. asset.setNotifyCategory(rs.getString("notifyCategory"));
```

```
178. public void setNotifyCategory(String notifyCategory) //asset.java
        ...
```

```
180. this.notifyCategory = notifyCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java

     ...
     314. asset.setNotifyCategory(rs.getString("notifyCategory"));
     ...
     322. vector.addElement(asset);
     ...
     328. assets[i] = (Asset) vector.elementAt(i);
     ...
     331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java

     ...
     61. Asset[] assets = rs2Assets(rs);
     ...
     65. asset = assets[0];
     ...
     74. return (asset);
```

```
1. <!-- //node.jsp

     ...
     194. Asset asset = this.model.getAsset( nodeId );
     ...
     299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
     ()%></td>
```

```
270. public String getDescription() //asset.java
     271. return (this.description);
```

```
1. <!-- //node.jsp

     ...
     299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
     ()%></td>
```

[Table of Contents] [Description]

Path 113:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java

     ...
     315. asset.setPollerCategory(rs.getString("pollerCategory"));
```

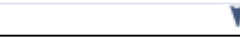
```
190. public void setPollerCategory(String pollerCategory) //asset.java

     ...
     192. this.pollerCategory = pollerCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      315. asset.setPollerCategory(rs.getString("pollerCategory"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```

```
1. <!-- //modify.jsp
      ...
      57. Asset asset = this.model.getAsset( nodeId );
      ...
      262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
      =asset.getComments()%></textarea></td>
```

```
606. public String getComments() //asset.java
      607. return (this.comments);
```



```
1. <!-- //modify.jsp
      ...
      262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
      =asset.getComments()%></textarea></td>
```

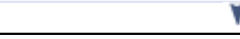
[Table of Contents] [Description]

Path 114:

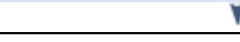
Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      315. asset.setPollerCategory(rs.getString("pollerCategory"));
```



```
190. public void setPollerCategory(String pollerCategory) //asset.java
      ...
      192. this.pollerCategory = pollerCategory;
```



```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
```

```
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

```
1. <!-- //modify.jsp
...
57. Asset asset = this.model.getAsset( nodeId );
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

```
150. public Date getLastModifiedDate() //asset.java
151. return (this.lastModifiedDate);
```

```
1. <!-- //modify.jsp
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 115:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
```

```
190. public void setPollerCategory(String pollerCategory) //asset.java
...
192. this.pollerCategory = pollerCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
...
```

```
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

```
1. <!-- //modify.jsp
...
57. Asset asset = this.model.getAsset( nodeId );
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

```
510. public String getUserLastModified() //asset.java
511. return (this.userLastModified);
```

```
1. <!-- //modify.jsp
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 116:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
```

```
190. public void setPollerCategory(String pollerCategory) //asset.java
...
192. this.pollerCategory = pollerCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
```

```
...
331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

```
1. <!-- //node.jsp
...
194. Asset asset = this.model.getAsset( nodeId );
...
299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
()%></td>
```

```
270. public String getDescription() //asset.java
271. return (this.description);
```

```
1. <!-- //node.jsp
...
299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
()%></td>
```

[Table of Contents] [Description]

Path 117:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
```

```
190. public void setPollerCategory(String pollerCategory) //asset.java
...
192. this.pollerCategory = pollerCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
315. asset.setPollerCategory(rs.getString("pollerCategory"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```

```
1. <!-- //node.jsp
      ...
      194. Asset asset = this.model.getAsset( nodeId );
      ...
      303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
      ></td>
```

```
606. public String getComments() //asset.java
      607. return (this.comments);
```

```
1. <!-- //node.jsp
      ...
      303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
      ></td>
```

[Table of Contents] [Description]

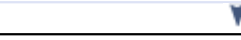
Path 118:

Query Name - **Stored_XSS**

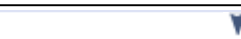
Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```

```
202. public void setThresholdCategory(String thresholdCategory) //asset.java
      ...
      204. this.thresholdCategory = thresholdCategory;
```



```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```



```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
```

```
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

```
1. <!-- //modify.jsp
...
57. Asset asset = this.model.getAsset( nodeId );
...
262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
=asset.getComments()%></textarea></td>
```

```
606. public String getComments() //asset.java
607. return (this.comments);
```

```
1. <!-- //modify.jsp
...
262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
=asset.getComments()%></textarea></td>
```

[Table of Contents] [Description]

Path 119:

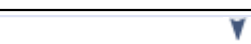
Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```

```
202. public void setThresholdCategory(String thresholdCategory) //asset.java
...
204. this.thresholdCategory = thresholdCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```



```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
```

```
...
74. return (asset);
```

```
1. <!-- //modify.jsp
...
57. Asset asset = this.model.getAsset( nodeId );
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

```
150. public Date getLastModifiedDate() //asset.java
151. return (this.lastModifiedDate);
```

```
1. <!-- //modify.jsp
...
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 120:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```

```
202. public void setThresholdCategory(String thresholdCategory) //asset.java
...
204. this.thresholdCategory = thresholdCategory;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
...
316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

```
1. <!-- //modify.jsp
      ...
      57. Asset asset = this.model.getAsset( nodeId );
      ...
      274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
      =asset.getUserLastModified()%></em>
```



```
510. public String getUserLastModified() //asset.java
      511. return (this.userLastModified);
```



```
1. <!-- //modify.jsp
      ...
      274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
      =asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 121:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```



```
202. public void setThresholdCategory(String thresholdCategory) //asset.java
      ...
      204. this.thresholdCategory = thresholdCategory;
```



```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```



```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```



```
1. <!-- //node.jsp
      ...
```

```
194. Asset asset = this.model.getAsset( nodeId );
...
299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
()%></td>
```



```
270. public String getDescription() //asset.java
     271. return (this.description);
```



```
1. <!-- //node.jsp
     ...
     299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
     ()%></td>
```

[Table of Contents] [Description]

Path 122:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
     ...
     316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
```



```
202. public void setThresholdCategory(String thresholdCategory) //asset.java
     ...
     204. this.thresholdCategory = thresholdCategory;
```



```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
     ...
     316. asset.setThresholdCategory(rs.getString("thresholdCategory"));
     ...
     322. vector.addElement(asset);
     ...
     328. assets[i] = (Asset) vector.elementAt(i);
     ...
     331. return assets;
```



```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
     ...
     61. Asset[] assets = rs2Assets(rs);
     ...
     65. asset = assets[0];
     ...
     74. return (asset);
```



```
1. <!-- //node.jsp
     ...
     194. Asset asset = this.model.getAsset( nodeId );
     ...
     303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
```

```
></td>
```

```
606. public String getComments() //asset.java
      607. return (this.comments);
```

```
1. <!-- //node.jsp
      ...
      303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
      ></td>
```

[Table of Contents] [Description]

Path 123:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
```

```
598. public void setComments(String comments) //asset.java
      ...
      600. this.comments = comments;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```

```
1. <!-- //modify.jsp
      ...
      57. Asset asset = this.model.getAsset( nodeId );
      ...
      262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
      =asset.getComments()%></textarea></td>
```

```
606. public String getComments() //asset.java
      607. return (this.comments);
```

```
1. <!-- //modify.jsp
      ...
      262. <td colspan="6"><textarea name="comments" cols="100" rows="15"><%
      =asset.getComments()%></textarea></td>
```

[Table of Contents] [Description]

Path 124:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
```

```
598. public void setComments(String comments) //asset.java
      ...
      600. this.comments = comments;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```

```
1. <!-- //modify.jsp
      ...
      57. Asset asset = this.model.getAsset( nodeId );
      ...
      274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
      =asset.getUserLastModified()%></em>
```

```
150. public Date getLastModifiedDate() //asset.java
      151. return (this.lastModifiedDate);
```

```
1. <!-- //modify.jsp
       ...
       274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
       =asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 125:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
       ...
       317. asset.setComments(rs.getString("comment"));
```

▼

```
598. public void setComments(String comments) //asset.java
       ...
       600. this.comments = comments;
```

▼

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
       ...
       317. asset.setComments(rs.getString("comment"));
       ...
       322. vector.addElement(asset);
       ...
       328. assets[i] = (Asset) vector.elementAt(i);
       ...
       331. return assets;
```

▼

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
       ...
       61. Asset[] assets = rs2Assets(rs);
       ...
       65. asset = assets[0];
       ...
       74. return (asset);
```

▼

```
1. <!-- //modify.jsp
       ...
       57. Asset asset = this.model.getAsset( nodeId );
       ...
       274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
       =asset.getUserLastModified()%></em>
```

▼

```
510. public String getUserLastModified() //asset.java
       511. return (this.userLastModified);
```

▼

```
1. <!-- //modify.jsp
       ...
```

```
274. <em>Last Modified: <%=asset.getLastModifiedDate()%> by <%
=asset.getUserLastModified()%></em>
```

[Table of Contents] [Description]

Path 126:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
```

```
...
317. asset.setComments(rs.getString("comment"));
```

▼

```
598. public void setComments(String comments) //asset.java
```

```
...
600. this.comments = comments;
```

▼

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
```

```
...
317. asset.setComments(rs.getString("comment"));
...
322. vector.addElement(asset);
...
328. assets[i] = (Asset) vector.elementAt(i);
...
331. return assets;
```

▼

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
```

```
...
61. Asset[] assets = rs2Assets(rs);
...
65. asset = assets[0];
...
74. return (asset);
```

▼

```
1. <!-- //node.jsp
```

```
...
194. Asset asset = this.model.getAsset( nodeId );
...
299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
()%></td>
```

▼

```
270. public String getDescription() //asset.java
```

```
271. return (this.description);
```

▼

```
1. <!-- //node.jsp
```

```
...
299. <td><%=(asset.getDescription() == null) ? " " : asset.getDescription
()%></td>
```

[Table of Contents] [Description]

Path 127:

Query Name - **Stored_XSS**

Severity - **Critical**

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
```

```
598. public void setComments(String comments) //asset.java
      ...
      600. this.comments = comments;
```

```
270. protected static Asset[] rs2Assets(ResultSet rs) throws
SQLException //assetmodel.java
      ...
      317. asset.setComments(rs.getString("comment"));
      ...
      322. vector.addElement(asset);
      ...
      328. assets[i] = (Asset) vector.elementAt(i);
      ...
      331. return assets;
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
      ...
      61. Asset[] assets = rs2Assets(rs);
      ...
      65. asset = assets[0];
      ...
      74. return (asset);
```

```
1. <!-- //node.jsp
      ...
      194. Asset asset = this.model.getAsset( nodeId );
      ...
      303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
      ></td>
```

```
606. public String getComments() //asset.java
      607. return (this.comments);
```

```
1. <!-- //node.jsp
      ...
      303. <td><%=(asset.getComments() == null) ? " " : asset.getComments()%
      ></td>
```

[Table of Contents] [Description]

Path 128:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
268. public String buildTableRows(List nodes, int start, int stop) //delete.jsp
      ...
      277. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
      ());
      ...
      291. row.append(nodelabel);
      292. row.append("</td>\n");
      293. row.append("</tr>\n");
      ...
      296. return row.toString();
```

```
1. <!-- //delete.jsp
      ...
      215. <%=buildTableRows(nodes, 0, midNodeIndex)%>
```

[Table of Contents] [Description]

Path 129:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
268. public String buildTableRows(List nodes, int start, int stop) //delete.jsp
      ...
      277. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
      ());
      ...
      291. row.append(nodelabel);
      292. row.append("</td>\n");
      293. row.append("</tr>\n");
      ...
      296. return row.toString();
```

```
1. <!-- //delete.jsp
      ...
      236. <%=buildTableRows(nodes, midNodeIndex, nodes.size())%>
```

[Table of Contents] [Description]

Path 130:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
170. public String buildTableRows(List nodes, int start, int stop) //snmpmanage.jsp
      ...
      179. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
      ());
      ...
      190. row.append(nodelabel);
      191. row.append("\">");
      192. row.append(nodelabel);
      193. row.append("</a>");
      194. row.append("</td>\n");
      195. row.append("</tr>\n");
      ...
      198. return row.toString();
```

```
1. <!-- //snmpmanage.jsp
      ...
      134. <%=buildTableRows(nodes, 0, midNodeIndex)%>
```

[Table of Contents] [Description]

Path 131:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
170. public String buildTableRows(List nodes, int start, int stop) //snmpmanage.jsp
      ...
      179. String nodelabel = NetworkElementFactory.getNodeLabel(curNode.getNodeID
      ());
      ...
      190. row.append(nodelabel);
      191. row.append("\">");
      192. row.append(nodelabel);
      193. row.append("</a>");
      194. row.append("</td>\n");
      195. row.append("</tr>\n");
      ...
      198. return row.toString();
```

```
1. <!-- //snmpmanage.jsp
       ...
       153. <%=buildTableRows(nodes, midNodeIndex, nodes.size())%>
```

[Table of Contents] [Description]

Path 132:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
       ...
       102. label = rs.getString("NODELABEL");
       ...
       111. return (label);
```

```
1. <!-- //managenode.jsp
       ...
       160. nodeLabel = NetworkElementFactory.getNodeLabel(firstInterface.getNodeid
       ());
       ...
       195. <h2>Node: <%=nodeLabel%></h2>
```

[Table of Contents] [Description]

Path 133:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
       ...
       102. label = rs.getString("NODELABEL");
       ...
       111. return (label);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
       ...
       269. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeID);
       ...
       272. " <td width=\"30%\">" + nodeLabel + "</td> " +
       ...
       297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
       ...
       166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
       [intCount].getIpAddress(), services);
       ...
       203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 134:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
      ...
      269. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeID);
      ...
      272. " <td width=\"30%\">" + nodeLabel + "</td> " +
      ...
      297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
      ...
      166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
      [intCount].getIpAddress(), services);
      ...
      224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 135:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //index.jsp
      ...
      162. <li> <%=asset.getAssetNumber()%>: <a href="asset/modify.jsp?node=<%
      =asset.getNodeId()%>"><%=NetworkElementFactory.getNodeLabel(asset.getNodeId
      ())%></a>
```

[Table of Contents] [Description]

Path 136:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    102. label = rs.getString("NODELABEL");
    ...
    111. return (label);
```

```
1. <!-- //index.jsp
    ...
    169. <li> <%=asset.getAssetNumber()%>: <a href="asset/modify.jsp?node=<%
    =asset.getNodeId()%>"><%=NetworkElementFactory.getNodeLabel(asset.getNodeId
    ())%></a>
```

[Table of Contents] [Description]

Path 137:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    102. label = rs.getString("NODELABEL");
    ...
    111. return (label);
```

```
1. <!-- //interface.jsp
    ...
    257. <td><a href="element/node.jsp?node=<%=intf_db.getNodeId()%>"><%
    =NetworkElementFactory.getNodeLabel(intf_db.getNodeId())%></a></td>
```

[Table of Contents] [Description]

Path 138:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    102. label = rs.getString("NODELABEL");
    ...
    111. return (label);
```

```
1. <!-- //rescan.jsp
    ...
    52. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeId);
    ...
    106. <p>Are you sure you want to rescan the <nobr><%=nodeLabel%></nobr>
```

[Table of Contents] [Description]

Path 139:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //service.jsp
      ...
      154. <td><a href="element/node.jsp?node=<%=service_db.getNodeId()%>"><%
      =NetworkElementFactory.getNodeLabel(service_db.getNodeId())%></a></td>
```

[Table of Contents] [Description]

Path 140:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //custom_graph3.jsp
      ...
      169. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 141:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //custom_report.jsp
      ...
      159. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 142:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //custom_view.jsp
      ...
      208. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```

[Table of Contents] [Description]

Path 143:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
285. public String getTextDescription() //noticefactory.java
      ...
      288. nodeName = NetworkElementFactory.getNodeLabel(this.nodeId);
      ...
      292. return (TYPE + "=" + nodeName);
```

```
1. <!-- //browser.jsp
      ...
      225. <li><strong><%=filter.getTextDescription()%></strong>
```

[Table of Contents] [Description]

Path 144:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //detail.jsp
      ...
      126. <a href="element/node.jsp?node=<%=notice.getNodeId()%>"><%
      =NetworkElementFactory.getNodeLabel(notice.getNodeId())%></a>
```

[Table of Contents] [Description]

Path 145:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //detail.jsp
      ...
      157. <a href="outage/list?filter=node%3D<%=notice.getNodeId()%>">See outages
      for <%=NetworkElementFactory.getNodeLabel(notice.getNodeId())%></a>
```

[Table of Contents] [Description]

Path 146:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
      111. return (label);
```

```
1. <!-- //results.jsp
      ...
      205. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
      =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 147:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      102. label = rs.getString("NODELABEL");
      ...
```

```
111. return (label);
```

```
1. <!-- //zoom.jsp
     ...
     135. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
     =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 148:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
     ...
     102. label = rs.getString("NODELABEL");
     ...
     111. return (label);
```

```
1. <!-- //results.jsp
     ...
     210. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
     =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```

[Table of Contents] [Description]

Path 149:

Query Name - **Stored_XSS**

Severity - **Critical**

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
     ...
     102. label = rs.getString("NODELABEL");
     ...
     111. return (label);
```

```
1. <!-- //category.jsp
     ...
     84. String nodeLabel = NetworkElementFactory.getNodeLabel((int)node.getNodeid
     ());
     ...
     89. nodeMap.put( nodeLabel, node );
     ...
     93. nodeMap.put( "nodeId=" + node.getNodeid(), node );
     ...
     97. Set keySet = nodeMap.keySet();
     98. Iterator nameIterator = keySet.iterator();
     ...
     142. String nodeLabel = (String)nameIterator.next();
     ...
     160. <td><a href="element/node.jsp?node=<%=node.getNodeid()%>"><%=nodeLabel%
     ></a></td>
```

[Table of Contents] [Description]

Path 150:

Query Name - **Stored_XSS**

Severity - **Critical**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
      ...
      181. ResultSet rs = stmt.executeQuery();
      ...
      183. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      614. node.m_nodeId = rs.getInt("nodeId");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
      ...
      183. nodes = rs2Nodes(rs);
      ...
      191. return nodes;
```

```
1. <!-- //nodelist.jsp
      ...
      50. nodes = NetworkElementFactory.getNodesLike( nameParm );
      ...
      100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
      [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 151:

Query Name - **Stored_XSS**

Severity - **Critical**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
      ...
      181. ResultSet rs = stmt.executeQuery();
      ...
      183. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
...
183. nodes = rs2Nodes(rs);
...
191. return nodes;
```

```
1. <!-- //nodelist.jsp
...
50. nodes = NetworkElementFactory.getNodesLike( nameParm );
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 152:

Query Name - **Stored_XSS**

Severity - **Critical**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
...
181. ResultSet rs = stmt.executeQuery();
...
183. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
...
183. nodes = rs2Nodes(rs);
...
191. return nodes;
```

```
1. <!-- //nodelistonly.jsp
```

```
...
52. nodes = NetworkElementFactory.getNodesLike( nameParm );
...
102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 153:

Query Name - **Stored_XSS**

Severity - **Critical**

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
```

```
...
181. ResultSet rs = stmt.executeQuery();
...
183. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
165. public static Node[] getNodesLike(String nodeLabel) throws
SQLException //networkelementfactory.java
```

```
...
183. nodes = rs2Nodes(rs);
...
191. return nodes;
```

```
1. <!-- //nodelistonly.jsp
```

```
...
52. nodes = NetworkElementFactory.getNodesLike( nameParm );
...
111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 154:

Query Name - **Stored_XSS**

Severity - **Critical**

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
```

```
...
208. ResultSet rs = stmt.executeQuery();
```

```
...
210. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java

...
210. nodes = rs2Nodes(rs);
...
218. return nodes;
```

```
1. <!-- //nodelist.jsp

...
53. nodes = NetworkElementFactory.getNodesWithIpLike( ipLikeParm );
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 155:

Query Name - **Stored_XSS**

Severity - **Critical**

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java

...
208. ResultSet rs = stmt.executeQuery();
...
210. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
```

```
...
210. nodes = rs2Nodes(rs);
...
218. return nodes;
```

```
1. <!-- //nodelist.jsp
...
53. nodes = NetworkElementFactory.getNodesWithIpLike( ipLikeParm );
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 156:

Query Name - **Stored_XSS**

Severity - **Critical**

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
...
208. ResultSet rs = stmt.executeQuery();
...
210. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
...
210. nodes = rs2Nodes(rs);
...
218. return nodes;
```

```
1. <!-- //nodelistonly.jsp
...
55. nodes = NetworkElementFactory.getNodesWithIpLike( ipLikeParm );
...
102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 157:

Query Name - **Stored_XSS**

Severity - Critical

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
    ...
    208. ResultSet rs = stmt.executeQuery();
    ...
    210. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    614. node.m_nodeId = rs.getInt("nodeId");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
197. public static Node[] getNodesWithIpLike(String iplike) throws
SQLException //networkelementfactory.java
    ...
    210. nodes = rs2Nodes(rs);
    ...
    218. return nodes;
```

```
1. <!-- //nodelistonly.jsp
    ...
    55. nodes = NetworkElementFactory.getNodesWithIpLike( ipLikeParm );
    ...
    111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
    [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 158:

Query Name - **Stored_XSS**

Severity - Critical

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
    ...
    231. ResultSet rs = stmt.executeQuery();
    ...
    233. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    614. node.m_nodeId = rs.getInt("nodeId");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java

...
233. nodes = rs2Nodes(rs);
...
241. return nodes;
```

```
1. <!-- //nodelist.jsp

...
57. nodes = NetworkElementFactory.getNodesWithService( serviceId );
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 159:

Query Name - **Stored_XSS**

Severity - **Critical**

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java

...
231. ResultSet rs = stmt.executeQuery();
...
233. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java

...
233. nodes = rs2Nodes(rs);
...
241. return nodes;
```

```
1. <!-- //nodelist.jsp

...
57. nodes = NetworkElementFactory.getNodesWithService( serviceId );
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 160:

Query Name - **Stored_XSS**

Severity - **Critical**

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
      ...
      231. ResultSet rs = stmt.executeQuery();
      ...
      233. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      614. node.m_nodeId = rs.getInt("nodeId");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
      ...
      233. nodes = rs2Nodes(rs);
      ...
      241. return nodes;
```

```
1. <!-- //nodelistonly.jsp
      ...
      59. nodes = NetworkElementFactory.getNodesWithService( serviceId );
      ...
      102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
      [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 161:

Query Name - **Stored_XSS**

Severity - **Critical**

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java
      ...
      231. ResultSet rs = stmt.executeQuery();
      ...
      233. nodes = rs2Nodes(rs);
```

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      |
```

```
...
614. node.m_nodeId = rs.getInt("nodeId");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
224. public static Node[] getNodesWithService(int serviceId) throws
SQLException //networkelementfactory.java

...
233. nodes = rs2Nodes(rs);
...
241. return nodes;
```

```
1. <!-- //nodelistonly.jsp

...
59. nodes = NetworkElementFactory.getNodesWithService( serviceId );
...
111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 162:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //deletenode.jsp

...
61. Node node_db = NetworkElementFactory.getNode(nodeId);
...
130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 163:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      615. node.m_dpname = rs.getString("dpName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

[Table of Contents] [Description]

Path 164:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      615. node.m_dpname = rs.getString("dpName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 165:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      615. node.m_dpname = rs.getString("dpName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
      ...
      142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
      (nodeList[j].getId());
      ...
      144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
      91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
      ...
      144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 166:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        615. node.m_dpname = rs.getString("dpName");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
        ...
        123. Node[] nodes = rs2Nodes(rs);
        ...
        127. node = nodes[0];
        ...
        136. return node;
```

```
1. <!-- //modify.jsp
        ...
        58. Node node_db = NetworkElementFactory.getNode( nodeId );
        ...
        96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 167:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        615. node.m_dpname = rs.getString("dpName");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
        ...
        123. Node[] nodes = rs2Nodes(rs);
        ...
        127. node = nodes[0];
        ...
        136. return node;
```

```
1. <!-- //modify.jsp
        ...
        58. Node node_db = NetworkElementFactory.getNode( nodeId );
```

```
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 168:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
```

```
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 169:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 170:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 171:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
```

```
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 172:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 173:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      615. node.m_dpname = rs.getString("dpName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
      ...
      123. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
      408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 174:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      615. node.m_dpname = rs.getString("dpName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp

...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 175:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    615. node.m_dpname = rs.getString("dpName");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
    ...
    222. <h2>Node: <%=node_db.getLabel()%></h2>
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
    ...
    310. <% if( node_db.getNodeSysId() != null ) { %>
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
    node_db.getNodeSysName()%></td>
    ...
    321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
    ()%></td>
    ...
    325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
    node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 176:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    615. node.m_dpname = rs.getString("dpName");
    ...
    639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp

...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 177:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelist.jsp
        ...
        60. nodes = NetworkElementFactory.getAllNodes();
        ...
        100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 178:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        615. node.m_dpname = rs.getString("dpName");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelist.jsp
        ...
        60. nodes = NetworkElementFactory.getAllNodes();
        ...
        119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 179:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
```

```
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelistonly.jsp

...
62. nodes = NetworkElementFactory.getAllNodes();
...
102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 180:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
615. node.m_dpname = rs.getString("dpName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelistonly.jsp

...
62. nodes = NetworkElementFactory.getAllNodes();
...
111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 181:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    615. node.m_dpname = rs.getString("dpName");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

▼

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

▼

```
1. <!-- //parent.jsp
    ...
    40. Node[] nodes = NetworkElementFactory.getAllNodes();
    ...
    82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 182:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    615. node.m_dpname = rs.getString("dpName");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

▼

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

▼

```
1. <!-- //parent.jsp
    ...
```

```
40. Node[] nodes = NetworkElementFactory.getAllNodes();
...
91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 183:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //deletenode.jsp
```

```
...
61. Node node_db = NetworkElementFactory.getNode(nodeId);
...
130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 184:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //deletenode.jsp
...
61. Node node_db = NetworkElementFactory.getNode(nodeId);
...
177. Node: <%=node_db.getLabel()%>
```

[Table of Contents] [Description]

Path 185:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //index.jsp
...
62. Node node_db = NetworkElementFactory.getNode( nodeId );
...
107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 186:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
```

```
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```



```
90. public String getLabel() //node.java
    91. return m_label;
```



```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 187:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java


...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
```

```
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 188:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 189:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
```

```
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 190:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 191:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    631. node.m_nodeSysId = rs.getString("nodeSysOID");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //modify.jsp
    ...
    58. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 192:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    631. node.m_nodeSysId = rs.getString("nodeSysOID");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 193:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 194:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
    408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 195:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    631. node.m_nodeSysId = rs.getString("nodeSysOID");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 196:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 197:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 198:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 199:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
631. node.m_nodeSysId = rs.getString("nodeSysOID");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 200:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      631. node.m_nodeSysId = rs.getString("nodeSysOID");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```



```
1. <!-- //nodelistonly.jsp
      ...
      62. nodes = NetworkElementFactory.getAllNodes();
      ...
      102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
      [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 201:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      631. node.m_nodeSysId = rs.getString("nodeSysOID");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```



```
1. <!-- //nodelistonly.jsp
      ...
      62. nodes = NetworkElementFactory.getAllNodes();
      ...
      111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
      [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 202:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      631. node.m_nodeSysId = rs.getString("nodeSysOID");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 203:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      631. node.m_nodeSysId = rs.getString("nodeSysOID");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 204:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 205:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```



```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

[Table of Contents] [Description]

Path 206:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```



```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 207:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 208:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 209:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 210:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 211:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 212:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //modify.jsp
      ...
      58. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 213:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 214:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 215:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
    408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 216:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    632. node.m_nodeSysName = rs.getString("nodeSysName");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 217:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 218:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 219:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 220:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
632. node.m_nodeSysName = rs.getString("nodeSysName");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 221:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        632. node.m_nodeSysName = rs.getString("nodeSysName");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 222:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        632. node.m_nodeSysName = rs.getString("nodeSysName");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 223:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 224:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      632. node.m_nodeSysName = rs.getString("nodeSysName");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 225:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 226:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

[Table of Contents] [Description]

Path 227:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 228:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
```

```
...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
```

```
91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
```

```
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 229:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 230:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 231:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 232:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 233:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //modify.jsp
      ...
      58. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 234:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 235:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 236:

Query Name - **Stored_XSS**

Severity - **Critical**

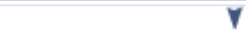
```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```



```
1. <!-- //node.jsp
      ...
      123. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
      408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

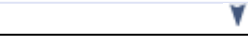
Path 237:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```



```
1. <!-- //node.jsp
      ...
      123. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 238:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 239:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 240:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 241:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 242:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```



```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```



```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 243:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```



```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```



```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 244:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

```
1. <!-- //parent.jsp
    ...
    40. Node[] nodes = NetworkElementFactory.getAllNodes();
    ...
    82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 245:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    633. node.m_nodeSysDescr = rs.getString("nodeSysDescription");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 246:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 247:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

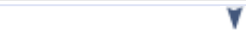
[Table of Contents] [Description]

Path 248:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 249:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 250:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 251:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 252:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 253:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```



```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 254:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //modify.jsp
      ...
      58. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 255:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 256:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 257:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
      ...
      123. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
      408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
      ...
      280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
      this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 258:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
      ...
      123. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 259:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 260:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 261:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 262:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 263:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 264:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 265:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 266:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      634. node.m_nodeSysLocn = rs.getString("nodeSysLocation");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 267:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 268:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

[Table of Contents] [Description]

Path 269:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 270:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

▼

```
90. public String getLabel() //node.java
91. return m_label;
```

▼

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 271:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 272:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 273:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```



```
1. <!-- //modify.jsp

    ...
    58. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 274:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

    ...
    635. node.m_nodeSysContact = rs.getString("nodeSysContact");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```



```
1. <!-- //modify.jsp

    ...
    58. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 275:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //modify.jsp
      ...
      58. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 276:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 277:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 278:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
    408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 279:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    635. node.m_nodeSysContact = rs.getString("nodeSysContact");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 280:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 281:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 282:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 283:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
635. node.m_nodeSysContact = rs.getString("nodeSysContact");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 284:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        635. node.m_nodeSysContact = rs.getString("nodeSysContact");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 285:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        635. node.m_nodeSysContact = rs.getString("nodeSysContact");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 286:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

▼

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 287:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      635. node.m_nodeSysContact = rs.getString("nodeSysContact");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

▼

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 288:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 289:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

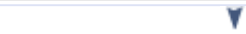
[Table of Contents] [Description]

Path 290:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 291:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
    91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 292:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 293:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 294:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 295:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 296:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //modify.jsp
      ...
      58. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 297:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 298:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 299:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
    408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 300:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    636. node.m_label = rs.getString("nodelabel");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 301:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 302:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 303:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 304:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
636. node.m_label = rs.getString("nodelabel");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 305:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        636. node.m_label = rs.getString("nodelabel");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 306:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        636. node.m_label = rs.getString("nodelabel");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 307:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 308:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      636. node.m_label = rs.getString("nodelabel");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
      ...
      150. nodes = rs2Nodes(rs);
      ...
      158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 309:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      637. node.m_operatingSystem = rs.getString("operatingsystem");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

▼

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      130. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 310:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      637. node.m_operatingSystem = rs.getString("operatingsystem");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //deletenode.jsp
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
      ...
      177. Node: <%=node_db.getLabel()%>
```

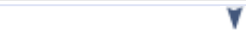
[Table of Contents] [Description]

Path 311:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      637. node.m_operatingSystem = rs.getString("operatingsystem");
      ...
      639. vector.addElement(node);
      ...
      645. nodes[i] = (Node) vector.elementAt(i);
      ...
      648. return (nodes);
```



```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      123. Node[] nodes = rs2Nodes(rs);
      ...
      127. node = nodes[0];
      ...
      136. return node;
```

```
1. <!-- //index.jsp
      ...
      62. Node node_db = NetworkElementFactory.getNode( nodeId );
      ...
      107. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 312:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
142. org.opennms.web.element.Node elementNode=NetworkElementFactory.getNode
(nodeList[j].getId());
...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

```
90. public String getLabel() //node.java
    91. return m_label;
```

```
1. <%@page language="java" contentType="text/html" session="true" //index.jsp

...
144. <%=elementNode==null?"NoNodeFound":elementNode.getLabel()%><BR>
```

[Table of Contents] [Description]

Path 313:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
```

```
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
96. <td> <%=node_db.getNodeSysId()%> </td>
```

[Table of Contents] [Description]

Path 314:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
98. <td> <%=node_db.getNodeSysName()%> </td>
```

[Table of Contents] [Description]

Path 315:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
```

```
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
102. <td> <%=node_db.getNodeSysLocn()%> </td>
```

[Table of Contents] [Description]

Path 316:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java

...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp

...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
104. <td> <%=node_db.getNodeSysContact()%> </td>
```

[Table of Contents] [Description]

Path 317:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //modify.jsp
```

```
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
...
108. <td> <%=node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 318:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
```

[Table of Contents] [Description]

Path 319:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
```

[Table of Contents] [Description]

Path 320:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
```

```
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

```
407. public String getStatusString( char c ) //node.jsp
    408. return( (String)this.statusMap.get( new Character(c) ));
```

```
1. <!-- //node.jsp
    ...
    280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
    this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
```

[Table of Contents] [Description]

Path 321:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    637. node.m_operatingSystem = rs.getString("operatingsystem");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
    ...
    123. Node[] nodes = rs2Nodes(rs);
    ...
    127. node = nodes[0];
    ...
    136. return node;
```

```
1. <!-- //node.jsp
    ...
    123. Node node_db = NetworkElementFactory.getNode( nodeId );
    ...
    317. <td><%=(node_db.getNodeSysName() == null) ? " " :
```

```
node_db.getNodeSysName()%></td>
```

[Table of Contents] [Description]

Path 322:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

▼

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

▼

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
```

[Table of Contents] [Description]

Path 323:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
123. Node[] nodes = rs2Nodes(rs);
...
127. node = nodes[0];
...
136. return node;
```

```
1. <!-- //node.jsp
```

```
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
...
199. <title><%=node_db.getLabel()%> | Node | OpenNMS Web Console</title>
...
222. <h2>Node: <%=node_db.getLabel()%></h2>
...
280. <td><%=(this.getStatusString(node_db.getNodeType())!=null ?
this.getStatusString(node_db.getNodeType()) : "Unknown")%></td>
...
310. <% if( node_db.getNodeSysId() != null ) { %>
...
317. <td><%=(node_db.getNodeSysName() == null) ? " " :
node_db.getNodeSysName()%></td>
...
321. <td><%=(node_db.getNodeSysId() == null) ? " " : node_db.getNodeSysId
()%></td>
...
325. <td><%=(node_db.getNodeSysLocn() == null) ? " " :
node_db.getNodeSysLocn()%></td>
...
329. <td><%=(node_db.getNodeSysContact() == null) ? " " :
node_db.getNodeSysContact()%></td>
...
333. <td valign="top" colspan="3"><%=(node_db.getNodeSysDescr() == null) ?
" " : node_db.getNodeSysDescr()%> </td>
```

[Table of Contents] [Description]

Path 324:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
```

```
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
100. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 325:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java

...
637. node.m_operatingSystem = rs.getString("operatingsystem");
...
639. vector.addElement(node);
...
645. nodes[i] = (Node) vector.elementAt(i);
...
648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java

...
150. nodes = rs2Nodes(rs);
...
158. return nodes;
```

```
1. <!-- //nodelist.jsp

...
60. nodes = NetworkElementFactory.getAllNodes();
...
119. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
[i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 326:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        637. node.m_operatingSystem = rs.getString("operatingsystem");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        102. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 327:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        637. node.m_operatingSystem = rs.getString("operatingsystem");
        ...
        639. vector.addElement(node);
        ...
        645. nodes[i] = (Node) vector.elementAt(i);
        ...
        648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
        ...
        150. nodes = rs2Nodes(rs);
        ...
        158. return nodes;
```

```
1. <!-- //nodelistonly.jsp
        ...
        62. nodes = NetworkElementFactory.getAllNodes();
        ...
        111. <li><a href="element/node.jsp?node=<%=nodes[i].getNodeId()%>"><%=nodes
        [i].getLabel()%></a>
```

[Table of Contents] [Description]

Path 328:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    637. node.m_operatingSystem = rs.getString("operatingsystem");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

```
1. <!-- //parent.jsp
    ...
    40. Node[] nodes = NetworkElementFactory.getAllNodes();
    ...
    82. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 329:

Query Name - **Stored_XSS**

Severity - **Critical**

```
602. protected static Node[] rs2Nodes(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    637. node.m_operatingSystem = rs.getString("operatingsystem");
    ...
    639. vector.addElement(node);
    ...
    645. nodes[i] = (Node) vector.elementAt(i);
    ...
    648. return (nodes);
```

```
142. public static Node[] getAllNodes() throws
SQLException //networkelementfactory.java
    ...
    150. nodes = rs2Nodes(rs);
    ...
    158. return nodes;
```

```
1. <!-- //parent.jsp
      ...
      40. Node[] nodes = NetworkElementFactory.getAllNodes();
      ...
      91. <%= nodes[i].getLabel() %>
```

[Table of Contents] [Description]

Path 330:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      746. service.m_nodeId = rs.getInt("nodeid");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

▼

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
      ...
      485. service = services[0];
      ...
      494. return service;
```

▼

```
1. <!-- //servicedeleted.jsp
      ...
      71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
      serviceId );
      ...
      83. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
      Web Console</title>
```

[Table of Contents] [Description]

Path 331:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      746. service.m_nodeId = rs.getInt("nodeid");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //servicedeleted.jsp
```

```
...
71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
114. <h3>Finished Deleting <%=service_db.getServiceName()%> Service on <%
=ipAddr%>.</h3>
```

[Table of Contents] [Description]

Path 332:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
746. service.m_nodeId = rs.getInt("nodeid");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

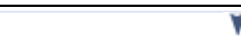
```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
```

```
...
162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
interfaces[intCount].getIpAddress());
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
```

```
services[]) //assignnotices.jsp
    ...
    289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
    ...
    297. return interfaceRow + interfaceSelect + serviceRow;
```



```
1. <!-- //assignnotices.jsp
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
    ...
    203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 333:

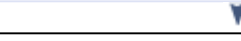
Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    746. service.m_nodeId = rs.getInt("nodeid");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
    517. return getServicesOnInterface(nodeId, ipAddress, false);
```



```
1. <!-- //assignnotices.jsp
    ...
    162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
    interfaces[intCount].getIpAddress());
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
    ...
    289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
```

```
...
297. return interfaceRow + interfaceSelect + serviceRow;
```



```
1. <!-- //assignnotices.jsp
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
...
224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 334:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
746. service.m_nodeId = rs.getInt("nodeid");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```



```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
...
540. services = rs2Services(rs);
...
548. return services;
```



```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```



```
1. <!-- //interface.jsp
...
132. Service[] services = NetworkElementFactory.getServicesOnInterface
( nodeId, ipAddr );
...
352. <td><a href="element/service.jsp?node=<%=services[i].getNodeId()%
>&intf=<%=services[i].getIpAddress()%>&service=<%=services[i].getServiceId()%
>"><%=services[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 335:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
```

```
SQLException //networkelementfactory.java

    ...
    746. service.m_nodeId = rs.getInt("nodeid");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java

    ...
    476. Service[] services = rs2Services(rs);
    ...
    485. service = services[0];
    ...
    494. return service;
```

```
1. <!-- //service.jsp

    ...
    72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
    serviceId );
    ...
    84. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
    Web Console</title>
```

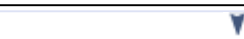
[Table of Contents] [Description]

Path 336:

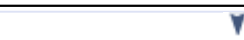
Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

    ...
    746. service.m_nodeId = rs.getInt("nodeid");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```



```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java

    ...
    476. Service[] services = rs2Services(rs);
    ...
    485. service = services[0];
    ...
    494. return service;
```



```
1. <!-- //service.jsp

    ...
    72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
```

```
serviceId );
...
125. <h2><%=service_db.getServiceName()%> service on <%
=service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 337:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
746. service.m_nodeId = rs.getInt("nodeid");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

▼

```
1. <!-- //service.jsp
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
125. <h2><%=service_db.getServiceName()%> service on <%
=service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 338:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
746. service.m_nodeId = rs.getInt("nodeid");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
    517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp
    ...
    172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
    (intf.getNodeId(), intf.getIpAddress());
    ...
    175. Arrays.sort(svcs, this.serviceComparator);
    ...
    178. return svcs;
```

```
1. <!-- //interfaceavailability-box.jsp
    ...
    98. Service[] services = this.getServices(intf);
    ...
    121. <% Service service = services[i]; %>
    ...
    132. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
    >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
    =service.getServiceName()%></a></td>
```

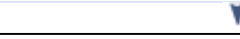
[Table of Contents] [Description]

Path 339:

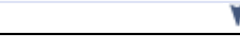
Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    746. service.m_nodeId = rs.getInt("nodeid");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```



```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```



```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
      517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
      ...
      225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
      (intf.getNodeId(), intf.getIpAddress());
      ...
      228. Arrays.sort(svcs, this.serviceComparator);
      ...
      231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp
      ...
      122. <% Service[] svcs = this.getServices(intf); %>
      ...
      130. <% Service service = svcs[j]; %>
      ...
      140. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
      >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
      =service.getServiceName()%></a></td>
```

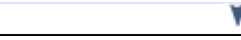
[Table of Contents] [Description]

Path 340:

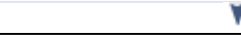
Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      747. service.m_ifIndex = rs.getInt("ifindex");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```



```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```



```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
      517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
```

```
...
225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
228. Arrays.sort(svcs, this.serviceComparator);
...
231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp

...
122. <% Service[] svcs = this.getServices(intf); %>
...
130. <% Service service = svcs[j]; %>
...
140. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

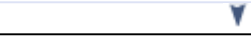
Path 341:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java

...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```



```
1. <!-- //servicedeleted.jsp

...
71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
83. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
Web Console</title>
```

[Table of Contents] [Description]

Path 342:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      748. service.m_ipAddr = rs.getString("ipaddr");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
      ...
      485. service = services[0];
      ...
      494. return service;
```

```
1. <!-- //servicedeleted.jsp
      ...
      71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
      serviceId );
      ...
      114. <h3>Finished Deleting <%=service_db.getServiceName()%> Service on <%
      =ipAddr%>.</h3>
```

[Table of Contents] [Description]

Path 343:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      748. service.m_ipAddr = rs.getString("ipaddr");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
    ...
    162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
    interfaces[intCount].getIpAddress());
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
    ...
    289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
    ...
    297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
    ...
    203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 344:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    748. service.m_ipAddr = rs.getString("ipaddr");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
    517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
```

```
...
162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
interfaces[intCount].getIpAddress());
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
```

```
...
289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
...
297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
```

```
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
...
224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 345:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //interface.jsp
```

```
...
132. Service[] services = NetworkElementFactory.getServicesOnInterface
( nodeId, ipAddr );
...
```

```
352. <td><a href="element/service.jsp?node=<%=services[i].getNodeId()%
>&intf=<%=services[i].getIpAddress()%>&service=<%=services[i].getServiceId()%
>"><%=services[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 346:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //service.jsp
```

```
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
84. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
Web Console</title>
```

[Table of Contents] [Description]

Path 347:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
        ...
        476. Service[] services = rs2Services(rs);
        ...
        485. service = services[0];
        ...
        494. return service;
```

```
1. <!-- //service.jsp
        ...
        72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
        serviceId );
        ...
        125. <h2><%=service_db.getServiceName()%> service on <%
        =service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 348:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
        ...
        748. service.m_ipAddr = rs.getString("ipaddr");
        ...
        768. vector.addElement(service);
        ...
        774. services[i] = (Service) vector.elementAt(i);
        ...
        777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
        ...
        476. Service[] services = rs2Services(rs);
        ...
        485. service = services[0];
        ...
        494. return service;
```

```
1. <!-- //service.jsp
        ...
        72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
        serviceId );
        ...
        125. <h2><%=service_db.getServiceName()%> service on <%
        =service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 349:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      748. service.m_ipAddr = rs.getString("ipaddr");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
      ...
      485. service = services[0];
      ...
      494. return service;
```

```
1. <!-- //service.jsp
      ...
      72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
      serviceId );
      ...
      158. <td><a href="element/interface.jsp?node=<%=service_db.getNodeId()%
      >&intf=<%=service_db.getIpAddress()%>"><%=service_db.getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 350:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      748. service.m_ipAddr = rs.getString("ipaddr");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp

...
172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
175. Arrays.sort(svcs, this.serviceComparator);
...
178. return svcs;
```

```
1. <!-- //interfaceavailability-box.jsp

...
98. Service[] services = this.getServices(intf);
...
121. <% Service service = services[i]; %>
...
127. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 351:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java

...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp


...
172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
```

```
(intf.getNodeId(), intf.getIpAddress());
...
175. Arrays.sort(svcs, this.serviceComparator);
...
178. return svcs;
```

```
1. <!-- //interfaceavailability-box.jsp
...
98. Service[] services = this.getServices(intf);
...
121. <% Service service = services[i]; %>
...
132. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 352:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
...
225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
228. Arrays.sort(svcs, this.serviceComparator);
...
231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp
```

```
...
122. <% Service[] svcs = this.getServices(intf); %>
...
130. <% Service service = svcs[j]; %>
...
135. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 353:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
748. service.m_ipAddr = rs.getString("ipaddr");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

▼

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

▼

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
```

```
...
225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
228. Arrays.sort(svcs, this.serviceComparator);
...
231. return svcs;
```

▼

```
1. <!-- //nodeavailability-box.jsp
```

```
...
122. <% Service[] svcs = this.getServices(intf); %>
...
130. <% Service service = svcs[j]; %>
...
140. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 354:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      755. service.m_serviceName = rs.getString("servicename");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
      ...
      485. service = services[0];
      ...
      494. return service;
```

```
1. <!-- //servicedeleted.jsp
      ...
      71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
      serviceId );
      ...
      83. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
      Web Console</title>
```

[Table of Contents] [Description]

Path 355:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      755. service.m_serviceName = rs.getString("servicename");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
```

```
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //servicedeleted.jsp

...
71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
114. <h3>Finished Deleting <%=service_db.getServiceName()%> Service on <%
=ipAddr%>.</h3>
```

[Table of Contents] [Description]

Path 356:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java

...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp

...
162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
interfaces[intCount].getIpAddress());
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp

...
289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
...
```

```
297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
    ...
    203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 357:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    755. service.m_serviceName = rs.getString("servicename");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
    517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
    ...
    162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
    interfaces[intCount].getIpAddress());
    ...
    166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
    [intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
    ...
    289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
    ...
    297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
```

```
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
...
224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 358:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

▼

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

▼

```
1. <!-- //interface.jsp
```

```
...
132. Service[] services = NetworkElementFactory.getServicesOnInterface
( nodeId, ipAddr );
...
352. <td><a href="element/service.jsp?node=<%=services[i].getNodeId()%
>&intf=<%=services[i].getIpAddress()%>&service=<%=services[i].getServiceId()%
>"><%=services[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 359:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
755. service.m_serviceName = rs.getString("servicename");
...
```

```
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //service.jsp
```

```
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
84. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
Web Console</title>
```

[Table of Contents] [Description]

Path 360:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //service.jsp
```

```
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
125. <h2><%=service_db.getServiceName()%> service on <%
=service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 361:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      755. service.m_serviceName = rs.getString("servicename");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
      ...
      485. service = services[0];
      ...
      494. return service;
```

```
1. <!-- //service.jsp
      ...
      72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
      serviceId );
      ...
      125. <h2><%=service_db.getServiceName()%> service on <%
      =service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 362:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      755. service.m_serviceName = rs.getString("servicename");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
      ...
      476. Service[] services = rs2Services(rs);
```

```
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //service.jsp

...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
158. <td><a href="element/interface.jsp?node=<%=service_db.getNodeId()%
>&intf=<%=service_db.getIpAddress()%>"><%=service_db.getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 363:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java

...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp

...
172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
175. Arrays.sort(svcs, this.serviceComparator);
...
178. return svcs;
```

```
1. <!-- //interfaceavailability-box.jsp


...
98. Service[] services = this.getServices(intf);
```

```
...
121. <% Service service = services[i]; %>
...
127. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 364:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

▼

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

▼

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp
```

```
...
172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
175. Arrays.sort(svcs, this.serviceComparator);
...
178. return svcs;
```

▼

```
1. <!-- //interfaceavailability-box.jsp
```

```
...
98. Service[] services = this.getServices(intf);
...
121. <% Service service = services[i]; %>
...
132. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 365:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      755. service.m_serviceName = rs.getString("servicename");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
      517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
      ...
      225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
      (intf.getNodeId(), intf.getIpAddress());
      ...
      228. Arrays.sort(svcs, this.serviceComparator);
      ...
      231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp
      ...
      122. <% Service[] svcs = this.getServices(intf); %>
      ...
      130. <% Service service = svcs[j]; %>
      ...
      135. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
      >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
      =service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 366:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
755. service.m_serviceName = rs.getString("servicename");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
```

```
...
225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
228. Arrays.sort(svcs, this.serviceComparator);
...
231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp
```

```
...
122. <% Service[] svcs = this.getServices(intf); %>
...
130. <% Service service = svcs[j]; %>
...
140. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 367:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //servicedeleted.jsp
```

```
...
71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
83. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
Web Console</title>
```

[Table of Contents] [Description]

Path 368:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

```
1. <!-- //servicedeleted.jsp
```

```
...
71. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
114. <h3>Finished Deleting <%=service_db.getServiceName()%> Service on <%
=ipAddr%>.</h3>
```

[Table of Contents] [Description]

Path 369:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      761. service.m_notify = rs.getString("notify");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
      517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
      ...
      162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
      interfaces[intCount].getIpAddress());
      ...
      166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
      [intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
      ...
      289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
      ...
      297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
      ...
      166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
      [intCount].getIpAddress(), services);
      ...
      203. <%=serviceRowsEachNode[nodeCount]%>
```

[Table of Contents] [Description]

Path 370:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
1. <!-- //assignnotices.jsp
```

```
...
162. Service services[] = NetworkElementFactory.getServicesOnInterface(nodeID,
interfaces[intCount].getIpAddress());
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
```

```
263. private String interfaceServiceList(int nodeID, String ipAddress, Service
services[]) //assignnotices.jsp
```

```
...
289. " <td width=\"10%\">" + services[i].getServiceName() + "</td> " +
...
297. return interfaceRow + interfaceSelect + serviceRow;
```

```
1. <!-- //assignnotices.jsp
```

```
...
166. serviceRowsEachNode[i] = interfaceServiceList(nodeID, interfaces
[intCount].getIpAddress(), services);
...
224. <%=serviceRowsEachNode[remaining]%>
```

[Table of Contents] [Description]

Path 371:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
```

```
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```



```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java

...
540. services = rs2Services(rs);
...
548. return services;
```



```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```



```
1. <!-- //interface.jsp

...
132. Service[] services = NetworkElementFactory.getServicesOnInterface
( nodeId, ipAddr );
...
352. <td><a href="element/service.jsp?node=<%=services[i].getNodeId()%
>&intf=<%=services[i].getIpAddress()%>&service=<%=services[i].getServiceId()%
>"><%=services[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 372:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```



```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java

...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```



```
1. <!-- //service.jsp


...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
```

```
serviceId );
...
84. <title><%=service_db.getServiceName()%> Service on <%=ipAddr%> | OpenNMS
Web Console</title>
```

[Table of Contents] [Description]

Path 373:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```

▼

```
1. <!-- //service.jsp
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
125. <h2><%=service_db.getServiceName()%> service on <%
=service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 374:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

▼

```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```



```
1. <!-- //service.jsp
```

```
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
125. <h2><%=service_db.getServiceName()%> service on <%
=service_db.getIpAddress()%></h2>
```

[Table of Contents] [Description]

Path 375:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
```

```
...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```



```
457. public static Service getService(int nodeId, String ipAddress, int serviceId)
throws SQLException //networkelementfactory.java
```

```
...
476. Service[] services = rs2Services(rs);
...
485. service = services[0];
...
494. return service;
```



```
1. <!-- //service.jsp
```

```
...
72. Service service_db = NetworkElementFactory.getService( nodeId, ipAddr,
serviceId );
...
158. <td><a href="element/interface.jsp?node=<%=service_db.getNodeId()%
>&intf=<%=service_db.getIpAddress()%>"><%=service_db.getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 376:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      761. service.m_notify = rs.getString("notify");
      ...
      768. vector.addElement(service);
      ...
      774. services[i] = (Service) vector.elementAt(i);
      ...
      777. return services;
```

▼

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
      ...
      540. services = rs2Services(rs);
      ...
      548. return services;
```

▼

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
      517. return getServicesOnInterface(nodeId, ipAddress, false);
```

▼

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp
      ...
      172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
      (intf.getNodeId(), intf.getIpAddress());
      ...
      175. Arrays.sort(svcs, this.serviceComparator);
      ...
      178. return svcs;
```

▼

```
1. <!-- //interfaceavailability-box.jsp
      ...
      98. Service[] services = this.getServices(intf);
      ...
      121. <% Service service = services[i]; %>
      ...
      127. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
      >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
      =service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 377:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
      ...
      761. service.m_notify = rs.getString("notify");
      ...
      768. vector.addElement(service);
```

```
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java

...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
167. public Service[] getServices(Interface intf) throws
java.sql.SQLException //interfaceavailability-box.jsp

...
172. Service[] svcs = NetworkElementFactory.getServicesOnInterface
(intf.getNodeId(), intf.getIpAddress());
...
175. Arrays.sort(svcs, this.serviceComparator);
...
178. return svcs;
```

```
1. <!-- //interfaceavailability-box.jsp

...
98. Service[] services = this.getServices(intf);
...
121. <% Service service = services[i]; %>
...
132. <td align="left" width="30%"><a href="element/service.jsp?node=<%=nodeId%
>&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
=service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 378:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java

...
761. service.m_notify = rs.getString("notify");
...
768. vector.addElement(service);
...
774. services[i] = (Service) vector.elementAt(i);
...
777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
```

```
...
540. services = rs2Services(rs);
...
548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
    517. return getServicesOnInterface(nodeId, ipAddress, false);
```

```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
    ...
    225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
    (intf.getNodeId(), intf.getIpAddress());
    ...
    228. Arrays.sort(svcs, this.serviceComparator);
    ...
    231. return svcs;
```

```
1. <!-- //nodeavailability-box.jsp
    ...
    122. <% Service[] svcs = this.getServices(intf); %>
    ...
    130. <% Service service = svcs[j]; %>
    ...
    135. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
    >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
    =service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 379:

Query Name - **Stored_XSS**

Severity - **Critical**

```
737. protected static Service[] rs2Services(ResultSet rs) throws
SQLException //networkelementfactory.java
    ...
    761. service.m_notify = rs.getString("notify");
    ...
    768. vector.addElement(service);
    ...
    774. services[i] = (Service) vector.elementAt(i);
    ...
    777. return services;
```

```
520. public static Service[] getServicesOnInterface(int nodeId, String ipAddress,
boolean includeDeletions) throws SQLException //networkelementfactory.java
    ...
    540. services = rs2Services(rs);
    ...
    548. return services;
```

```
516. public static Service[] getServicesOnInterface(int nodeId, String ipAddress)
throws SQLException //networkelementfactory.java
```

```
517. return getServicesOnInterface(nodeId, ipAddress, false);
```



```
220. public Service[] getServices(Interface intf) throws
java.sql.SQLException //nodeavailability-box.jsp
    ...
    225. Service[] svcs = NetworkElementFactory.getServicesOnInterface
    (intf.getNodeId(), intf.getIpAddress());
    ...
    228. Arrays.sort(svcs, this.serviceComparator);
    ...
    231. return svcs;
```



```
1. <!-- //nodeavailability-box.jsp
    ...
    122. <% Service[] svcs = this.getServices(intf); %>
    ...
    130. <% Service service = svcs[j]; %>
    ...
    140. <td align="left" width="25%"><a href="element/service.jsp?node=<%=nodeId%
    >&intf=<%=ipAddr%>&service=<%=service.getServiceId()%>"><%
    =service.getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 380:

Query Name - **Stored_XSS**

Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
    ...
    833. ResultSet rs = stmt.executeQuery("SELECT SERVICEID, SERVICENAME FROM
    SERVICE");
    ...
    837. String name = rs.getString("SERVICENAME");
    ...
    839. idMap.put(new Integer(id), name);
    ...
    849. serviceId2NameMap = idMap;
```



```
780. public static String getServiceNameFromId(int serviceId) throws
SQLException //networkelementfactory.java
    ...
    782. createServiceIdNameMaps();
    ...
    785. String serviceName = (String) serviceId2NameMap.get(new Integer
    (serviceId));
    ...
    787. return (serviceName);
```



```
365. public String getTextDescription() //noticefactory.java
    ...
    368. serviceName = NetworkElementFactory.getServiceNameFromId(this.serviceId);
    ...
    372. return (TYPE + "=" + serviceName);
```

```
1. <!-- //browser.jsp
      ...
      225. <li><strong><%=filter.getTextDescription()%></strong>
```

[Table of Contents] [Description]

Path 381:

Query Name - **Stored_XSS**

Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
      ...
      837. String name = rs.getString("SERVICENAME");
      ...
      840. nameMap.put(name, new Integer(id));
      ...
      850. serviceName2IdMap = nameMap;
```

```
818. public static Map getServiceNameToIdMap() throws
SQLException //networkelementfactory.java
      ...
      820. createServiceIdNameMaps();
      ...
      823. return (new HashMap(serviceName2IdMap));
```

```
1. <!-- //index.jsp
      ...
      53. Map serviceNameMap = new TreeMap
      (NetworkElementFactory.getServiceNameToIdMap());
      54. Set serviceNameSet = serviceNameMap.keySet();
      55. Iterator serviceNameIterator = serviceNameSet.iterator();
      ...
      112. <% String name = (String)serviceNameIterator.next(); %>
      113. <option value="<%=serviceNameMap.get(name)%>"><%=name%></option>
```

[Table of Contents] [Description]

Path 382:

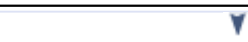
Query Name - **Stored_XSS**

Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java
      ...
      837. String name = rs.getString("SERVICENAME");
      ...
      840. nameMap.put(name, new Integer(id));
      ...
      850. serviceName2IdMap = nameMap;
```

```
818. public static Map getServiceNameToIdMap() throws
SQLException //networkelementfactory.java
      ...
```

```
820. createServiceIdNameMaps();
...
823. return (new HashMap(serviceName2IdMap));
```



```
1. <!-- //advquerypanel.jsp

     ...
     42. Map serviceNameMap = new TreeMap
     (NetworkElementFactory.getServiceNameToIdMap());
     43. Set serviceNameSet = serviceNameMap.keySet();
     44. Iterator serviceNameIterator = serviceNameSet.iterator();
     ...
     102. <% String name = (String)serviceNameIterator.next(); %>
     103. <option value="<%=serviceNameMap.get(name)%>"><%=name%></option>
```

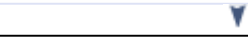
[Table of Contents] [Description]

Path 383:

Query Name - **Stored_XSS**

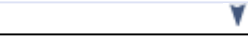
Severity - **Critical**

```
826. protected static void createServiceIdNameMaps() throws
SQLException //networkelementfactory.java

     ...
     837. String name = rs.getString("SERVICENAME");
     ...
     839. idMap.put(new Integer(id), name);
     ...
     849. serviceId2NameMap = idMap;
```



```
780. public static String getServiceNameFromId(int serviceId) throws
SQLException //networkelementfactory.java

     ...
     782. createServiceIdNameMaps();
     ...
     785. String serviceName = (String) serviceId2NameMap.get(new Integer
     (serviceId));
     ...
     787. return (serviceName);
```

```
365. public String getTextDescription() //noticefactory.java

     ...
     368. serviceName = NetworkElementFactory.getServiceNameFromId(this.serviceId);
     ...
     372. return (TYPE + "=" + serviceName);
```



```
1. <!-- //browser.jsp

     ...
     225. <li><strong><%=filter.getTextDescription()%></strong>
```

[Table of Contents] [Description]

Path 384:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        994. element = rs.getString("eventUei");
        995. event.uei = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 385:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        994. element = rs.getString("eventUei");
        995. event.uei = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

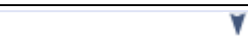
Path 386:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
997. element = rs.getString("eventSnmp");
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 387:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
```

```
997. element = rs.getString("eventSnmp");
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 388:

Query Name - **Stored_XSS**

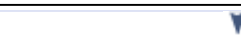
Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
997. element = rs.getString("eventSnmp");
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
```

```
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 389:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
997. element = rs.getString("eventSnmp");
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 390:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      997. element = rs.getString("eventSnmp");
      998. event.snmp = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 391:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      997. element = rs.getString("eventSnmp");
```

```
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
    ()%></a>
```

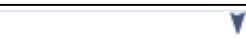
[Table of Contents] [Description]

Path 392:

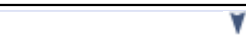
Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    997. element = rs.getString("eventSnmp");
    998. event.snmp = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```



```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 393:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
997. element = rs.getString("eventSnmp");
998. event.snmp = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 394:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1003. element = rs.getString("eventHost");
      1004. event.host = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 395:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1003. element = rs.getString("eventHost");
      1004. event.host = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
```

```
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 396:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1003. element = rs.getString("eventHost");
1004. event.host = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 397:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1003. element = rs.getString("eventHost");
```

```
1004. event.host = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 398:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1003. element = rs.getString("eventHost");
1004. event.host = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 399:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1003. element = rs.getString("eventHost");
1004. event.host = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 400:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1003. element = rs.getString("eventHost");
1004. event.host = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 401:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1003. element = rs.getString("eventHost");
      1004. event.host = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 402:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1006. element = rs.getString("eventSnmpHost");
      1007. event.snmphost = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 403:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1006. element = rs.getString("eventSnmpHost");
1007. event.snmphost = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 404:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1006. element = rs.getString("eventSnmpHost");
1007. event.snmphost = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 405:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1006. element = rs.getString("eventSnmpHost");
1007. event.snmphost = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 406:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1006. element = rs.getString("eventSnmpHost");
1007. event.snmphost = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
```

```
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 407:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1006. element = rs.getString("eventSnmpHost");
    1007. event.snmphost = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
    ()%></a>
```

[Table of Contents] [Description]

Path 408:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1006. element = rs.getString("eventSnmpHost");
    1007. event.snmphost = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

▼

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 409:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1006. element = rs.getString("eventSnmpHost");
    1007. event.snmphost = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 410:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1009. element = rs.getString("eventDpName");
    1010. event.dpName = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 411:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1009. element = rs.getString("eventDpName");
      1010. event.dpName = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 412:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1009. element = rs.getString("eventDpName");
      1010. event.dpName = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
```

```
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 413:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1009. element = rs.getString("eventDpName");
1010. event.dpName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp


...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
```

```
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 414:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1009. element = rs.getString("eventDpName");
1010. event.dpName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 415:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1009. element = rs.getString("eventDpName");
      1010. event.dpName = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

▼

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
      ()%></a>
```

[Table of Contents] [Description]

Path 416:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1009. element = rs.getString("eventDpName");
      1010. event.dpName = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 417:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1009. element = rs.getString("eventDpName");
    1010. event.dpName = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 418:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1012. element = rs.getString("eventParms");
      1013. event.parms = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 419:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1012. element = rs.getString("eventParms");
      1013. event.parms = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
```

```
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp

    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 420:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1012. element = rs.getString("eventParms");
    1013. event.parms = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp

    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
    " "%></td>
```

[Table of Contents] [Description]

Path 421:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1012. element = rs.getString("eventParms");
      1013. event.parms = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 422:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
```

```
1012. element = rs.getString("eventParms");
1013. event.parms = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

     ...
     282. Event[] events = rs2Events(rs);
     ...
     286. event = events[0];
     ...
     295. return event;
```

```
1. <!-- //detail.jsp

     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     69. if (event.getAcknowledgeTime()==null)
     ...
     106. <h3>Event <%=event.getId()%></h3>
     ...
     111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
     =EventUtil.getSeverityLabel(event.getSeverity())%></td>
     ...
     114. <% if( event.getNodeId() > 0 ) { %>
     115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
     =event.getNodeLabel()%></a>
     ...
     121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
     " "%></td>
     ...
     125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
     ())%></td>
     ...
     128. <% if( event.getIpAddress() != null ) { %>
     ...
     132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 423:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

     ...
     1012. element = rs.getString("eventParms");
     1013. event.parms = (String) element;
     ...
     1085. vector.addElement(event);
     ...
     1091. events[i] = (Event) vector.elementAt(i);
     ...
     1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
    ()%></a>
```

[Table of Contents] [Description]

Path 424:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1012. element = rs.getString("eventParms");
    1013. event.parms = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 425:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1012. element = rs.getString("eventParms");
        1013. event.parms = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 426:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1016. element = rs.getObject("nodeID");
...
1020. event.nodeID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 427:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1016. element = rs.getObject("nodeID");
...
1020. event.nodeID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
```

```
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 428:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1016. element = rs.getObject("nodeID");
    ...
    1020. event.nodeID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

▼

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
    " "%></td>
```

[Table of Contents] [Description]

Path 429:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1016. element = rs.getObject("nodeID");
    ...
    1020. event.nodeID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
```

```
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    69. if (event.getAcknowledgeTime()==null)
    ...
    106. <h3>Event <%=event.getId()%></h3>
    ...
    111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
    =EventUtil.getSeverityLabel(event.getSeverity())%></td>
    ...
    114. <% if( event.getNodeId() > 0 ) { %>
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
    ...
    121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
    " "%></td>
    ...
    125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
    ())%></td>
    ...
    128. <% if( event.getIpAddress() != null ) { %>
    ...
    130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
    =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

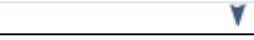
[Table of Contents] [Description]

Path 430:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1016. element = rs.getObject("nodeID");
    ...
    1020. event.nodeID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 431:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1016. element = rs.getObject("nodeID");
...
1020. event.nodeID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
```

```
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
    ()%></a>
```

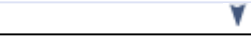
[Table of Contents] [Description]

Path 432:

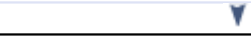
Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1016. element = rs.getObject("nodeID");
    ...
    1020. event.nodeID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```



```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 433:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1016. element = rs.getObject("nodeID");
    ...
    1020. event.nodeID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 434:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1023. element = rs.getString("ipAddr");
    1024. event.ipAddr = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 435:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1023. element = rs.getString("ipAddr");
        1024. event.ipAddr = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
        =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 436:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1023. element = rs.getString("ipAddr");
1024. event.ipAddr = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 437:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1023. element = rs.getString("ipAddr");
1024. event.ipAddr = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 438:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1023. element = rs.getString("ipAddr");
1024. event.ipAddr = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
```

```
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 439:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1023. element = rs.getString("ipAddr");
1024. event.ipAddr = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 440:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1023. element = rs.getString("ipAddr");
      1024. event.ipAddr = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
      [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 441:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1023. element = rs.getString("ipAddr");
      1024. event.ipAddr = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 442:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1026. element = rs.getObject("serviceID");
    1027. event.serviceID = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 443:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1026. element = rs.getObject("serviceID");
      1027. event.serviceID = (Integer) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 444:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1026. element = rs.getObject("serviceID");
      1027. event.serviceID = (Integer) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
```

```
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 445:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1026. element = rs.getObject("serviceID");
1027. event.serviceID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp


...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
```

```
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 446:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1026. element = rs.getObject("serviceID");
1027. event.serviceID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 447:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1026. element = rs.getObject("serviceID");
      1027. event.serviceID = (Integer) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
      ()%></a>
```

[Table of Contents] [Description]

Path 448:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1026. element = rs.getObject("serviceID");
      1027. event.serviceID = (Integer) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 449:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1026. element = rs.getObject("serviceID");
1027. event.serviceID = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
      ...
      108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
      throttle, offset);
      ...
      176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 450:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1029. element = rs.getString("nodeLabel");
      1030. event.nodeLabel = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

▼

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 451:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1029. element = rs.getString("nodeLabel");
      1030. event.nodeLabel = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
```

```
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp

    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 452:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1029. element = rs.getString("nodeLabel");
    1030. event.nodeLabel = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp

    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
    " "%></td>
```

[Table of Contents] [Description]

Path 453:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1029. element = rs.getString("nodeLabel");
      1030. event.nodeLabel = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 454:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
```

```
1029. element = rs.getString("nodeLabel");
1030. event.nodeLabel = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 455:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1029. element = rs.getString("nodeLabel");
1030. event.nodeLabel = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
    ()%></a>
```

[Table of Contents] [Description]

Path 456:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1029. element = rs.getString("nodeLabel");
    1030. event.nodeLabel = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 457:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1029. element = rs.getString("nodeLabel");
        1030. event.nodeLabel = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

▼

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 458:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 459:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
```

```
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 460:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 461:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 462:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
```

```
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 463:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
        ()%></a>
```

[Table of Contents] [Description]

Path 464:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1032. element = rs.getString("serviceName");
        1033. event.serviceName = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

▼

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 465:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java
    ...
1032. element = rs.getString("serviceName");
1033. event.serviceName = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
    ...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

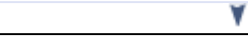
[Table of Contents] [Description]

Path 466:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
```

```
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

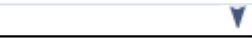
[Table of Contents] [Description]

Path 467:

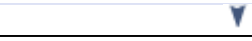
Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 468:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
```

```
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 469:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
```

```
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 470:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

```
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 471:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1038. element = rs.getString("eventDescr");
1039. event.description = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 472:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1038. element = rs.getString("eventDescr");
        1039. event.description = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```



```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```



```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 473:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1038. element = rs.getString("eventDescr");
        1039. event.description = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
```

```
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 474:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1041. element = rs.getString("eventLoggroup");
1042. event.logGroup = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 475:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1041. element = rs.getString("eventLoggroup");
    1042. event.logGroup = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 476:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1041. element = rs.getString("eventLoggroup");
    1042. event.logGroup = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
```

[Table of Contents] [Description]

Path 477:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1041. element = rs.getString("eventLoggroup");
        1042. event.logGroup = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        69. if (event.getAcknowledgeTime()==null)
        ...
        106. <h3>Event <%=event.getId()%></h3>
        ...
        111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
        =EventUtil.getSeverityLabel(event.getSeverity())%></td>
        ...
        114. <% if( event.getNodeId() > 0 ) { %>
        115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
        =event.getNodeLabel()%></a>
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
        ...
        125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
        ())%></td>
        ...
        128. <% if( event.getIpAddress() != null ) { %>
        ...
        130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
        =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 478:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1041. element = rs.getString("eventLoggroup");
      1042. event.logGroup = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 479:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java
    ...
1041. element = rs.getString("eventLoggroup");
1042. event.logGroup = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
    ...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 480:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1041. element = rs.getString("eventLoggroup");
1042. event.logGroup = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
```

```
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 481:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1041. element = rs.getString("eventLoggroup");
    1042. event.logGroup = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
```

```
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 482:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1044. element = rs.getString("eventLogmsg");
1045. event.logMessage = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 483:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1044. element = rs.getString("eventLogmsg");
1045. event.logMessage = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 484:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1044. element = rs.getString("eventLogmsg");
1045. event.logMessage = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 485:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java

    ...
    1044. element = rs.getString("eventLogmsg");
    1045. event.logMessage = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp

    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    69. if (event.getAcknowledgeTime()==null)
    ...
    106. <h3>Event <%=event.getId()%></h3>
    ...
    111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
    =EventUtil.getSeverityLabel(event.getSeverity())%></td>
    ...
    114. <% if( event.getNodeId() > 0 ) { %>
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
    ...
    121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
    " "%></td>
    ...
    125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
    ())%></td>
    ...
    128. <% if( event.getIpAddress() != null ) { %>
    ...
    130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
    =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 486:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1044. element = rs.getString("eventLogmsg");
    1045. event.logMessage = (String) element;
    ...
    1085. vector.addElement(event);
    ...
```

```
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

     ...
     282. Event[] events = rs2Events(rs);
     ...
     286. event = events[0];
     ...
     295. return event;
```

```
1. <!-- //detail.jsp

     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     69. if (event.getAcknowledgeTime()==null)
     ...
     106. <h3>Event <%=event.getId()%></h3>
     ...
     111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
     =EventUtil.getSeverityLabel(event.getSeverity())%></td>
     ...
     114. <% if( event.getNodeId() > 0 ) { %>
     115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
     =event.getNodeLabel()%></a>
     ...
     121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
     " "%></td>
     ...
     125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
     ())%></td>
     ...
     128. <% if( event.getIpAddress() != null ) { %>
     ...
     132. <%=event.getIpAddress()%>
```

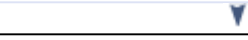
[Table of Contents] [Description]

Path 487:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

     ...
     1044. element = rs.getString("eventLogmsg");
     1045. event.logMessage = (String) element;
     ...
     1085. vector.addElement(event);
     ...
     1091. events[i] = (Event) vector.elementAt(i);
     ...
     1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

     
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 488:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1044. element = rs.getString("eventLogmsg");
1045. event.logMessage = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
```

```
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 489:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1044. element = rs.getString("eventLogmsg");
    1045. event.logMessage = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp
    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 490:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
```

```
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

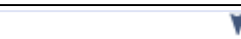
[Table of Contents] [Description]

Path 491:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
```

```
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 492:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 493:

Query Name - **Stored_XSS**

Severity - **Critical**

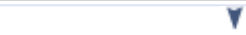
```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
```

```
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```



```
1. <!-- //detail.jsp

      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 494:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

      ...
      1050. element = rs.getString("eventOperInstruct");
      1051. event.operatorInstruction = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        69. if (event.getAcknowledgeTime()==null)
        ...
        106. <h3>Event <%=event.getId()%></h3>
        ...
        111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
        =EventUtil.getSeverityLabel(event.getSeverity())%></td>
        ...
        114. <% if( event.getNodeId() > 0 ) { %>
        115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
        =event.getNodeLabel()%></a>
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
        ...
        125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
        ())%></td>
        ...
        128. <% if( event.getIpAddress() != null ) { %>
        ...
        132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 495:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1050. element = rs.getString("eventOperInstruct");
        1051. event.operatorInstruction = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 496:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 497:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1050. element = rs.getString("eventOperInstruct");
1051. event.operatorInstruction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 498:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1053. element = rs.getString("eventAutoAction");
1054. event.autoAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
```

```
295. return event;
```

```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

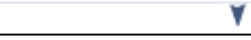
Path 499:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1053. element = rs.getString("eventAutoAction");
     1054. event.autoAction = (String) element;
     ...
     1085. vector.addElement(event);
     ...
     1091. events[i] = (Event) vector.elementAt(i);
     ...
     1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
     ...
     282. Event[] events = rs2Events(rs);
     ...
     286. event = events[0];
     ...
     295. return event;
```



```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
     =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 500:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1053. element = rs.getString("eventAutoAction");
     1054. event.autoAction = (String) element;
     ...
```

```
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 501:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1053. element = rs.getString("eventAutoAction");
1054. event.autoAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
```

```
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 502:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1053. element = rs.getString("eventAutoAction");
1054. event.autoAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
```

```
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 503:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1053. element = rs.getString("eventAutoAction");
1054. event.autoAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 504:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1053. element = rs.getString("eventAutoAction");
        1054. event.autoAction = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 505:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1053. element = rs.getString("eventAutoAction");
        1054. event.autoAction = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
```

```
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 506:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1056. element = rs.getString("eventOperAction");
1057. event.operatorAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 507:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1056. element = rs.getString("eventOperAction");
      1057. event.operatorAction = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 508:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1056. element = rs.getString("eventOperAction");
      1057. event.operatorAction = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
```

[Table of Contents] [Description]

Path 509:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1056. element = rs.getString("eventOperAction");
        1057. event.operatorAction = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

▼

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        69. if (event.getAcknowledgeTime()==null)
        ...
        106. <h3>Event <%=event.getId()%></h3>
        ...
        111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
        =EventUtil.getSeverityLabel(event.getSeverity())%></td>
        ...
        114. <% if( event.getNodeId() > 0 ) { %>
        115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
        =event.getNodeLabel()%></a>
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
        ...
        125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
        ())%></td>
        ...
        128. <% if( event.getIpAddress() != null ) { %>
        ...
        130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
        =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 510:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1056. element = rs.getString("eventOperAction");
      1057. event.operatorAction = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 511:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java
    ...
1056. element = rs.getString("eventOperAction");
1057. event.operatorAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
    ...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 512:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1056. element = rs.getString("eventOperAction");
1057. event.operatorAction = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
```

```
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 513:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1056. element = rs.getString("eventOperAction");
    1057. event.operatorAction = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
```

```
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 514:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 515:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 516:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 517:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java

      ...
      1059. element = rs.getString("eventOperActionMenuText");
      1060. event.operatorActionMenuText = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp

      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 518:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

      ...
      1059. element = rs.getString("eventOperActionMenuText");
      1060. event.operatorActionMenuText = (String) element;
      ...
      1085. vector.addElement(event);
      ...
```

```
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

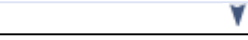
[Table of Contents] [Description]

Path 519:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 520:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
```

```
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 521:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1059. element = rs.getString("eventOperActionMenuText");
1060. event.operatorActionMenuText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 522:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
```

```
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 523:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
```

```
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 524:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

▼

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 525:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

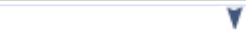
```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
```

```
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 526:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 527:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1062. element = rs.getString("eventNotification");
      1063. event.notification = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 528:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 529:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1062. element = rs.getString("eventNotification");
1063. event.notification = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 530:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1065. element = rs.getString("eventTticket");
1066. event.troubleTicket = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
```

```
295. return event;
```

```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

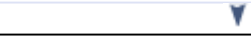
Path 531:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1065. element = rs.getString("eventTticket");
     1066. event.troubleTicket = (String) element;
     ...
     1085. vector.addElement(event);
     ...
     1091. events[i] = (Event) vector.elementAt(i);
     ...
     1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
     ...
     282. Event[] events = rs2Events(rs);
     ...
     286. event = events[0];
     ...
     295. return event;
```



```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
     =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 532:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1065. element = rs.getString("eventTticket");
     1066. event.troubleTicket = (String) element;
     ...
```

```
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 533:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1065. element = rs.getString("eventTticket");
1066. event.troubleTicket = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
```

```
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 534:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1065. element = rs.getString("eventTticket");
    1066. event.troubleTicket = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    69. if (event.getAcknowledgeTime()==null)
    ...
    106. <h3>Event <%=event.getId()%></h3>
    ...
    111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
    =EventUtil.getSeverityLabel(event.getSeverity())%></td>
    ...
    114. <% if( event.getNodeId() > 0 ) { %>
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
    ...
```

```
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 535:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1065. element = rs.getString("eventTticket");
1066. event.troubleTicket = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 536:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1065. element = rs.getString("eventTticket");
        1066. event.troubleTicket = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```

▼

```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 537:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1065. element = rs.getString("eventTticket");
        1066. event.troubleTicket = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
```

```
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 538:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1068. element = rs.getObject("eventTticketState");
1069. event.troubleTicketState = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 539:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1068. element = rs.getObject("eventTticketState");
    1069. event.troubleTicketState = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 540:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1068. element = rs.getObject("eventTticketState");
    1069. event.troubleTicketState = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
```

[Table of Contents] [Description]

Path 541:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1068. element = rs.getObject("eventTticketState");
        1069. event.troubleTicketState = (Integer) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
        ...
        282. Event[] events = rs2Events(rs);
        ...
        286. event = events[0];
        ...
        295. return event;
```

▼

```
1. <!-- //detail.jsp
        ...
        60. Event event = EventFactory.getEvent( eventId );
        ...
        69. if (event.getAcknowledgeTime()==null)
        ...
        106. <h3>Event <%=event.getId()%></h3>
        ...
        111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
        =EventUtil.getSeverityLabel(event.getSeverity())%></td>
        ...
        114. <% if( event.getNodeId() > 0 ) { %>
        115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
        =event.getNodeLabel()%></a>
        ...
        121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
        " "%></td>
        ...
        125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
        ())%></td>
        ...
        128. <% if( event.getIpAddress() != null ) { %>
        ...
        130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
        =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 542:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1068. element = rs.getObject("eventTticketState");
      1069. event.troubleTicketState = (Integer) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 543:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java
    ...
1068. element = rs.getObject("eventTticketState");
1069. event.troubleTicketState = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
    ...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
    ...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 544:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
1068. element = rs.getObject("eventTticketState");
1069. event.troubleTicketState = (Integer) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
    ...
392. events = rs2Events(rs);
...
```

```
400. return events;
```



```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
    174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
    [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 545:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

    ...
    1068. element = rs.getObject("eventTticketState");
    1069. event.troubleTicketState = (Integer) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

    ...
    392. events = rs2Events(rs);
    ...
    400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

    ...
    447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
    offset));
```

```
1. <!-- //eventlist.jsp

    ...
    108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
    throttle, offset);
    ...
```

```
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 546:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 547:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 548:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 549:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
```

```
SQLException //eventfactory.java

      ...
      1071. element = rs.getString("eventForward");
      1072. event.forward = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp

      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 550:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

      ...
      1071. element = rs.getString("eventForward");
      1072. event.forward = (String) element;
      ...
      1085. vector.addElement(event);
      ...
```

```
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp

...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

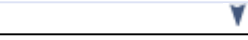
[Table of Contents] [Description]

Path 551:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 552:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
```

```
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 553:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1071. element = rs.getString("eventForward");
1072. event.forward = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 554:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
```

```
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

Path 555:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
```

```
=event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 556:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
```

```
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
```

```
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 557:

Query Name - **Stored_XSS**

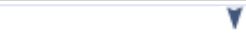
Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
```

```
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```



```
1. <!-- //detail.jsp

      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
      =event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 558:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

      ...
      1074. element = rs.getString("eventMouseOverText");
      1075. event.mouseOverText = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

      ...
      282. Event[] events = rs2Events(rs);
      ...
      286. event = events[0];
      ...
      295. return event;
```

```
1. <!-- //detail.jsp
      ...
      60. Event event = EventFactory.getEvent( eventId );
      ...
      69. if (event.getAcknowledgeTime()==null)
      ...
      106. <h3>Event <%=event.getId()%></h3>
      ...
      111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
      =EventUtil.getSeverityLabel(event.getSeverity())%></td>
      ...
      114. <% if( event.getNodeId() > 0 ) { %>
      115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
      =event.getNodeLabel()%></a>
      ...
      121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
      " "%></td>
      ...
      125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
      ())%></td>
      ...
      128. <% if( event.getIpAddress() != null ) { %>
      ...
      132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 559:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
      ...
      1074. element = rs.getString("eventMouseOverText");
      1075. event.mouseOverText = (String) element;
      ...
      1085. vector.addElement(event);
      ...
      1091. events[i] = (Event) vector.elementAt(i);
      ...
      1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
      ...
      392. events = rs2Events(rs);
      ...
      400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
      ...
      447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
      offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 560:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

▼

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

▼

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

▼

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
[i].getTime())%></td>
```

[Table of Contents] [Description]

Path 561:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1074. element = rs.getString("eventMouseOverText");
1075. event.mouseOverText = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java

...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 562:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java

...
1077. element = rs.getString("eventAckUser");
1078. event.acknowledgeUser = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java

...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
```

```
295. return event;
```

```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     106. <h3>Event <%=event.getId()%></h3>
```

[Table of Contents] [Description]

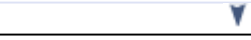
Path 563:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1077. element = rs.getString("eventAckUser");
     1078. event.acknowledgeUser = (String) element;
     ...
     1085. vector.addElement(event);
     ...
     1091. events[i] = (Event) vector.elementAt(i);
     ...
     1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
     ...
     282. Event[] events = rs2Events(rs);
     ...
     286. event = events[0];
     ...
     295. return event;
```



```
1. <!-- //detail.jsp
     ...
     60. Event event = EventFactory.getEvent( eventId );
     ...
     115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
     =event.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 564:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
     ...
     1077. element = rs.getString("eventAckUser");
     1078. event.acknowledgeUser = (String) element;
     ...
```

```
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```

```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
```

[Table of Contents] [Description]

Path 565:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
...
1077. element = rs.getString("eventAckUser");
1078. event.acknowledgeUser = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
...
282. Event[] events = rs2Events(rs);
...
286. event = events[0];
...
295. return event;
```



```
1. <!-- //detail.jsp
...
60. Event event = EventFactory.getEvent( eventId );
...
69. if (event.getAcknowledgeTime()==null)
...
106. <h3>Event <%=event.getId()%></h3>
```

```
...
111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
=EventUtil.getSeverityLabel(event.getSeverity())%></td>
...
114. <% if( event.getNodeId() > 0 ) { %>
115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
=event.getNodeLabel()%></a>
...
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
130. <a href="element/interface.jsp?node=<%=event.getNodeId()%>&intf=<%
=event.getIpAddress()%>"><%=event.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 566:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
    ...
    1077. element = rs.getString("eventAckUser");
    1078. event.acknowledgeUser = (String) element;
    ...
    1085. vector.addElement(event);
    ...
    1091. events[i] = (Event) vector.elementAt(i);
    ...
    1094. return events;
```

▼

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
    ...
    282. Event[] events = rs2Events(rs);
    ...
    286. event = events[0];
    ...
    295. return event;
```

▼

```
1. <!-- //detail.jsp
    ...
    60. Event event = EventFactory.getEvent( eventId );
    ...
    69. if (event.getAcknowledgeTime()==null)
    ...
    106. <h3>Event <%=event.getId()%></h3>
    ...
    111. <td bgcolor="<%=EventUtil.getSeverityColor(event.getSeverity())%>"><%
    =EventUtil.getSeverityLabel(event.getSeverity())%></td>
    ...
    114. <% if( event.getNodeId() > 0 ) { %>
    115. <a href="element/node.jsp?node=<%=event.getNodeId()%>"><%
    =event.getNodeLabel()%></a>
    ...
```

```
121. <td><%=event.getAcknowledgeUser()!=null ? event.getAcknowledgeUser() :
" "%></td>
...
125. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(event.getTime
())%></td>
...
128. <% if( event.getIpAddress() != null ) { %>
...
132. <%=event.getIpAddress()%>
```

[Table of Contents] [Description]

Path 567:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
```

```
...
1077. element = rs.getString("eventAckUser");
1078. event.acknowledgeUser = (String) element;
...
1085. vector.addElement(event);
...
1091. events[i] = (Event) vector.elementAt(i);
...
1094. return events;
```

```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
```

```
...
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
```

```
...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp
```

```
...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
171. <a href="event/detail.jsp?id=<%=events[i].getId()%>"><%=events[i].getId
()%></a>
```

[Table of Contents] [Description]

Path 568:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1077. element = rs.getString("eventAckUser");
        1078. event.acknowledgeUser = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
        392. events = rs2Events(rs);
        ...
        400. return events;
```



```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java
        ...
        447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
        offset));
```



```
1. <!-- //eventlist.jsp
        ...
        108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
        throttle, offset);
        ...
        174. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(events
        [i].getTime())%></td>
```

[Table of Contents] [Description]

Path 569:

Query Name - **Stored_XSS**

Severity - **Critical**

```
984. protected static Event[] rs2Events(ResultSet rs) throws
SQLException //eventfactory.java
        ...
        1077. element = rs.getString("eventAckUser");
        1078. event.acknowledgeUser = (String) element;
        ...
        1085. vector.addElement(event);
        ...
        1091. events[i] = (Event) vector.elementAt(i);
        ...
        1094. return events;
```



```
357. public static Event[] getEvents(SortStyle sortStyle, AcknowledgeType ackType,
Filter[] filters, int limit, int offset) throws SQLException //eventfactory.java
        ...
```

```
392. events = rs2Events(rs);
...
400. return events;
```

```
441. public static Event[] getEventsForNode(int nodeId, SortStyle sortStyle,
AcknowledgeType ackType, int throttle, int offset) throws
SQLException //eventfactory.java

...
447. return (EventFactory.getEvents(sortStyle, ackType, filters, throttle,
offset));
```

```
1. <!-- //eventlist.jsp

...
108. events = EventFactory.getEventsForNode(nodeId, sortStyle, ackType,
throttle, offset);
...
176. <td><%=events[i].getLogMessage()%></td>
```

[Table of Contents] [Description]

Path 570:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java

...
121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java

...
121. newSentTo.setUserId(sentToResults.getString(USERID));
...
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp

...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 571:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 572:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
      72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      121. newSentTo.setUserId(sentToResults.getString(USERID));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 573:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
121. newSentTo.setUserId(sentToResults.getString(USERID));
...
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 574:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
121. newSentTo.setUserId(sentToResults.getString(USERID));
```

```
71. public void setUserId(String userid) //noticesentto.java
72. m_userId = userid;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    121. newSentTo.setUserId(sentToResults.getString(USERID));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
    ...
    60. Notification notice = this.model.getNoticeInfo(noticeID);
    ...
    104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
    =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 575:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
    101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 576:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
    101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 577:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
    101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    127. newSentTo.setMedia(sentToResults.getString(MEDIA));
    ...
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 578:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

▼

```
100. public void setMedia(String media) //noticesentto.java
101. m_media = media;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
127. newSentTo.setMedia(sentToResults.getString(MEDIA));
...
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
=notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 579:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

▼

```
100. public void setMedia(String media) //noticesentto.java
101. m_media = media;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      127. newSentTo.setMedia(sentToResults.getString(MEDIA));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 580:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
      101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      127. newSentTo.setMedia(sentToResults.getString(MEDIA));
      ...
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 581:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
     101. m_media = media;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     127. newSentTo.setMedia(sentToResults.getString(MEDIA));
     ...
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```

```
1. <!-- //detail.jsp
     ...
     60. Notification notice = this.model.getNoticeInfo(noticeID);
     ...
     104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
     =notice.getEventId()%></a>
```

[Table of Contents] [Description]

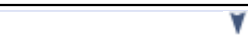
Path 582:

Query Name - **Stored_XSS**

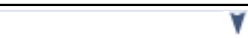
Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     127. newSentTo.setMedia(sentToResults.getString(MEDIA));
```

```
100. public void setMedia(String media) //noticesentto.java
     101. m_media = media;
```



```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     127. newSentTo.setMedia(sentToResults.getString(MEDIA));
     ...
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```



```
1. <!-- //detail.jsp
```

```
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
(notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 583:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

▼

```
114. public void setContactInfo(String contact) //noticesentto.java
115. m_contactInfo = contact;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 584:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

▼

```
114. public void setContactInfo(String contact) //noticesentto.java
115. m_contactInfo = contact;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
```

```
SQLException //notificationmodel.java
      ...
      128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```



```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 585:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```



```
114. public void setContactInfo(String contact) //noticesentto.java
      115. m_contactInfo = contact;
```



```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
      129. sentToList.add(newSentTo);
      ...
      132. nbean.m_sentTo = sentToList;
      ...
      196. return nbean;
```



```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 586:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
```

```
SQLException //notificationmodel.java
    ...
    128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
    115. m_contactInfo = contact;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
    =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 587:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
    115. m_contactInfo = contact;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
    129. sentToList.add(newSentTo);
    ...
    132. nbean.m_sentTo = sentToList;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
```

```
128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 588:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

▼

```
114. public void setContactInfo(String contact) //noticesentto.java
115. m_contactInfo = contact;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
     ...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 589:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

▼

```
114. public void setContactInfo(String contact) //noticesentto.java
115. m_contactInfo = contact;
```

▼

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 590:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
115. m_contactInfo = contact;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
129. sentToList.add(newSentTo);
...
132. nbean.m_sentTo = sentToList;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 591:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
     115. m_contactInfo = contact;
```



```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```



```
1. <!-- //detail.jsp
     ...
     60. Notification notice = this.model.getNoticeInfo(noticeID);
     ...
     104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
     =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 592:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```



```
114. public void setContactInfo(String contact) //noticesentto.java
     115. m_contactInfo = contact;
```



```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```



```
1. <!-- //detail.jsp
     ...
     60. Notification notice = this.model.getNoticeInfo(noticeID);
     ...
     111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
     (notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 593:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
     115. m_contactInfo = contact;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```

```
1. <!-- //detail.jsp
     ...
     60. Notification notice = this.model.getNoticeInfo(noticeID);
     ...
     113. <td width="17%"><%=notice.getTimeReplied()!=null ?
     org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
     " "%></td>
```

[Table of Contents] [Description]

Path 594:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
```

```
114. public void setContactInfo(String contact) //noticesentto.java
     115. m_contactInfo = contact;
```

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
     ...
     128. newSentTo.setContactInfo(sentToResults.getString(CONTACT));
     129. sentToList.add(newSentTo);
     ...
     132. nbean.m_sentTo = sentToList;
     ...
     196. return nbean;
```

```
1. <!-- //detail.jsp
        ...
        60. Notification notice = this.model.getNoticeInfo(noticeID);
        ...
        115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
        " "%></td>
```

[Table of Contents] [Description]

Path 595:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        137. ResultSet rs = pstmt.executeQuery();
        ...
        141. Object element = rs.getString(TXT_MESG);
        142. nbean.m_txtMsg = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 596:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        137. ResultSet rs = pstmt.executeQuery();
        ...
        141. Object element = rs.getString(TXT_MESG);
        142. nbean.m_txtMsg = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 597:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      137. ResultSet rs = pstmt.executeQuery();
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 598:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      137. ResultSet rs = pstmt.executeQuery();
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 599:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      137. ResultSet rs = pstmt.executeQuery();
      ...
      141. Object element = rs.getString(TXT_MESG);
```

```
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 600:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
137. ResultSet rs = pstmt.executeQuery();
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 601:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
137. ResultSet rs = pstmt.executeQuery();
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 602:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      137. ResultSet rs = pstmt.executeQuery();
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 603:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      137. ResultSet rs = pstmt.executeQuery();
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
      =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 604:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
137. ResultSet rs = pstmt.executeQuery();
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
(notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 605:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
137. ResultSet rs = pstmt.executeQuery();
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
113. <td width="17%"><%=notice.getTimeReplied()!=null ?
org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
" "%></td>
```

[Table of Contents] [Description]

Path 606:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
137. ResultSet rs = pstmt.executeQuery();
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
      " "%></td>
```

[Table of Contents] [Description]

Path 607:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 608:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 609:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 610:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 611:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      141. Object element = rs.getString(TXT_MESG);
      142. nbean.m_txtMsg = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
```

```
128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 612:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 613:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 614:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
141. Object element = rs.getString(TXT_MESG);
```

```
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 615:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
=notice.getEventId()%></a>
```

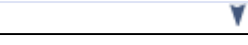
[Table of Contents] [Description]

Path 616:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
141. Object element = rs.getString(TXT_MESG);
142. nbean.m_txtMsg = (String) element;
...
196. return nbean;
```



```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
(notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 617:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    141. Object element = rs.getString(TXT_MESG);
    142. nbean.m_txtMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
    ...
    60. Notification notice = this.model.getNoticeInfo(noticeID);
    ...
    113. <td width="17%"><%=notice.getTimeReplied()!=null ?
    org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
    " "%></td>
```

[Table of Contents] [Description]

Path 618:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    141. Object element = rs.getString(TXT_MESG);
    142. nbean.m_txtMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
    ...
    60. Notification notice = this.model.getNoticeInfo(noticeID);
    ...
    115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
    " "%></td>
```

[Table of Contents] [Description]

Path 619:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    144. element = rs.getString(NUM_MESG);
    145. nbean.m_numMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 620:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      144. element = rs.getString(NUM_MESG);
      145. nbean.m_numMsg = (String) element;
      ...
      196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 621:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      144. element = rs.getString(NUM_MESG);
      145. nbean.m_numMsg = (String) element;
      ...
      196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 622:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        144. element = rs.getString(NUM_MESG);
        145. nbean.m_numMsg = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
        =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 623:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        144. element = rs.getString(NUM_MESG);
        145. nbean.m_numMsg = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 624:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        144. element = rs.getString(NUM_MESG);
        145. nbean.m_numMsg = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
```

```
=notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 625:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
144. element = rs.getString(NUM_MESG);
145. nbean.m_numMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 626:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
144. element = rs.getString(NUM_MESG);
145. nbean.m_numMsg = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
144. <a href="element/service.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>&service=<%=notices[i].getServiceId()%>"><%
=notices[i].getServiceName()%></a>
```

[Table of Contents] [Description]

Path 627:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
```

```
144. element = rs.getString(NUM_MESG);
145. nbean.m_numMsg = (String) element;
...
196. return nbean;
```



```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
146. <%=notices[i].getServiceName()%>
```

[Table of Contents] [Description]

Path 628:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
144. element = rs.getString(NUM_MESG);
145. nbean.m_numMsg = (String) element;
...
196. return nbean;
```



```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
153. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeReplied()%> </td>
```

[Table of Contents] [Description]

Path 629:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
144. element = rs.getString(NUM_MESG);
145. nbean.m_numMsg = (String) element;
...
196. return nbean;
```



```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 630:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    144. element = rs.getString(NUM_MESG);
    145. nbean.m_numMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
    ...
    60. Notification notice = this.model.getNoticeInfo(noticeID);
    ...
    104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
    =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 631:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    144. element = rs.getString(NUM_MESG);
    145. nbean.m_numMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
    ...
    60. Notification notice = this.model.getNoticeInfo(noticeID);
    ...
    111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
    (notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 632:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    144. element = rs.getString(NUM_MESG);
    145. nbean.m_numMsg = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       113. <td width="17%"><%=notice.getTimeReplied()!=null ?
       org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
       " "%></td>
```

[Table of Contents] [Description]

Path 633:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
       ...
       144. element = rs.getString(NUM_MESG);
       145. nbean.m_numMsg = (String) element;
       ...
       196. return nbean;
```

▼

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
       " "%></td>
```

[Table of Contents] [Description]

Path 634:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
       ...
       144. element = rs.getString(NUM_MESG);
       145. nbean.m_numMsg = (String) element;
       ...
       196. return nbean;
```

▼

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       135. <a href="element/interface.jsp?node=<%=notice.getNodeId()%>&intf=<%
       =notice.getIpAddress()%>"><%=notice.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 635:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 636:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 637:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
```

```
...
124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 638:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
159. element = rs.getString(ANS_BY);
160. nbean.m_responder = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
=notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 639:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
159. element = rs.getString(ANS_BY);
160. nbean.m_responder = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
```

```
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 640:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
```

```
...
159. element = rs.getString(ANS_BY);
160. nbean.m_responder = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 641:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
159. element = rs.getString(ANS_BY);
160. nbean.m_responder = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 642:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
159. element = rs.getString(ANS_BY);
160. nbean.m_responder = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
144. <a href="element/service.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
=notices[i].getIpAddress()%>&service=<%=notices[i].getServiceId()%>"><%
=notices[i].getServiceName()%></a>
```

[Table of Contents] [Description]

Path 643:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    146. <%=notices[i].getServiceName()%>
```

[Table of Contents] [Description]

Path 644:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //acknowledge.jsp
    ...
    77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
    ...
    153. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeReplied()%> </td>
```

[Table of Contents] [Description]

Path 645:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
    ...
    159. element = rs.getString(ANS_BY);
    160. nbean.m_responder = (String) element;
    ...
    196. return nbean;
```

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 646:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
       ...
       159. element = rs.getString(ANS_BY);
       160. nbean.m_responder = (String) element;
       ...
       196. return nbean;
```

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
       =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 647:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
       ...
       159. element = rs.getString(ANS_BY);
       160. nbean.m_responder = (String) element;
       ...
       196. return nbean;
```

```
1. <!-- //detail.jsp
       ...
       60. Notification notice = this.model.getNoticeInfo(noticeID);
       ...
       111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
       (notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 648:

Query Name - **Stored_XSS**

Severity

- Critical

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      159. element = rs.getString(ANS_BY);
      160. nbean.m_responder = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      113. <td width="17%"><%=notice.getTimeReplied()!=null ?
      org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
      " "%></td>
```

[Table of Contents] [Description]

Path 649:

Query Name - **Stored_XSS**

Severity - Critical

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      159. element = rs.getString(ANS_BY);
      160. nbean.m_responder = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
      " "%></td>
```

[Table of Contents] [Description]

Path 650:

Query Name - **Stored_XSS**

Severity - Critical

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      159. element = rs.getString(ANS_BY);
      160. nbean.m_responder = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
```

```
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
135. <a href="element/interface.jsp?node=<%=notice.getNodeId()%>&intf=<%
=notice.getIpAddress()%>"><%=notice.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 651:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
165. element = rs.getString(INTERFACE);
166. nbean.m_interfaceID = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 652:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
165. element = rs.getString(INTERFACE);
166. nbean.m_interfaceID = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 653:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
```

```
SQLException //notificationmodel.java

      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp

      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 654:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java

      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp

      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 655:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java

      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp

      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 656:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 657:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 658:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
```

```
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      144. <a href="element/service.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>&service=<%=notices[i].getServiceId()%>"><%
      =notices[i].getServiceName()%></a>
```

[Table of Contents] [Description]

Path 659:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      146. <%=notices[i].getServiceName()%>
```

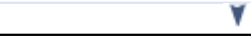
[Table of Contents] [Description]

Path 660:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```



```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      153. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeReplied()%> </td>
```

[Table of Contents] [Description]

Path 661:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 662:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
      =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 663:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
```

```
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
(notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 664:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
165. element = rs.getString(INTERFACE);
166. nbean.m_interfaceID = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
113. <td width="17%"><%=notice.getTimeReplied()!=null ?
org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
" "%></td>
```

[Table of Contents] [Description]

Path 665:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
165. element = rs.getString(INTERFACE);
166. nbean.m_interfaceID = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
" "%></td>
```

[Table of Contents] [Description]

Path 666:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      165. element = rs.getString(INTERFACE);
      166. nbean.m_interfaceID = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      135. <a href="element/interface.jsp?node=<%=notice.getNodeId()%>&intf=<%
      =notice.getIpAddress()%>"><%=notice.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 667:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 668:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 669:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 670:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 671:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

[Table of Contents] [Description]

Path 672:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 673:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
        where serviceid = " + nbean.m_serviceId);
        ...
        178. element = rstmp.getString("servicename");
        ...
        180. nbean.m_serviceName = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //acknowledge.jsp
        ...
        77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
        ...
        136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 674:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
        where serviceid = " + nbean.m_serviceId);
        ...
        178. element = rstmp.getString("servicename");
        ...
        180. nbean.m_serviceName = (String) element;
        ...
        196. return nbean;
```

```
1. <!-- //detail.jsp
        ...
        60. Notification notice = this.model.getNoticeInfo(noticeID);
        ...
        102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 675:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
        ...
        175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
        where serviceid = " + nbean.m_serviceId);
        ...
        178. element = rstmp.getString("servicename");
        ...
```

```
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
=notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 676:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
where serviceid = " + nbean.m_serviceId);
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
(notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 677:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
where serviceid = " + nbean.m_serviceId);
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
```

```
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
113. <td width="17%"><%=notice.getTimeReplied()!=null ?
org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
" "%></td>
```

[Table of Contents] [Description]

Path 678:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
175. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
where serviceid = " + nbean.m_serviceId);
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
" "%></td>
```

[Table of Contents] [Description]

Path 679:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

▼

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
110. <li> Notice #<%=notices[i].getId()%> was successfully acknowledged.
```

[Table of Contents] [Description]

Path 680:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      120. <h4> Notice <%=notices[i].getId()%> Summary </h4>
```

[Table of Contents] [Description]

Path 681:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      124. <td BGCOLOR="#cccccc" width="15%"> <a HREF="notification/detail.jsp?
      notice=<%=notices[i].getId()%>"><%=notices[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 682:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
```

```
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      126. <td BGCOLOR="#cccccc" width="15%"> <a href="event/detail.jsp?id=<%
      =notices[i].getEventId()%>"><%=notices[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 683:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      128. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeSent()%> </td>
```

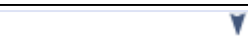
[Table of Contents] [Description]

Path 684:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```



```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      134. <a href="element/interface.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>"><%=notices[i].getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 685:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      136. <%=notices[i].getIpAddress()%>
```

[Table of Contents] [Description]

Path 686:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //acknowledge.jsp
      ...
      77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
      ...
      144. <a href="element/service.jsp?node=<%=notices[i].getNodeId()%>&intf=<%
      =notices[i].getIpAddress()%>&service=<%=notices[i].getServiceId()%>"><%
      =notices[i].getServiceName()%></a>
```

[Table of Contents] [Description]

Path 687:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
```

```
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
146. <%=notices[i].getServiceName()%>
```

[Table of Contents] [Description]

Path 688:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //acknowledge.jsp
...
77. notices[i] = this.model.getNoticeInfo( Integer.parseInt( noticeIds[i] ));
...
153. <td BGCOLOR="#cccccc" width="15%"> <%=notices[i].getTimeReplied()%> </td>
```

[Table of Contents] [Description]

Path 689:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
102. <h3>Notice #<%=notice.getId()%>
```

[Table of Contents] [Description]

Path 690:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      104. from event #<a href="event/detail.jsp?id=<%=notice.getEventId()%>"><%
      =notice.getEventId()%></a>
```

[Table of Contents] [Description]

Path 691:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
      ...
      196. return nbean;
```

```
1. <!-- //detail.jsp
      ...
      60. Notification notice = this.model.getNoticeInfo(noticeID);
      ...
      111. <td width="17%"><%=org.opennms.netmgt.EventConstants.formatToUIString
      (notice.getTimeSent())%></td>
```

[Table of Contents] [Description]

Path 692:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
      ...
      178. element = rstmp.getString("servicename");
      ...
      180. nbean.m_serviceName = (String) element;
```

```
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
113. <td width="17%"><%=notice.getTimeReplied()!=null ?
org.opennms.netmgt.EventConstants.formatToUIString(notice.getTimeReplied()) :
" "%></td>
```

[Table of Contents] [Description]

Path 693:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
115. <td width="17%"><%=notice.getResponder()!=null ? notice.getResponder() :
" "%></td>
```

[Table of Contents] [Description]

Path 694:

Query Name - **Stored_XSS**

Severity - **Critical**

```
106. public Notification getNoticeInfo(int id) throws
SQLException //notificationmodel.java
...
178. element = rstmp.getString("servicename");
...
180. nbean.m_serviceName = (String) element;
...
196. return nbean;
```

```
1. <!-- //detail.jsp
...
60. Notification notice = this.model.getNoticeInfo(noticeID);
...
135. <a href="element/interface.jsp?node=<%=notice.getNodeId()%>&intf=<%
=notice.getIpAddress()%>"><%=notice.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 695:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      243. element = rs.getString(NUM_MESG);
      244. nbean.m_numMsg = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
      =notices[i].getId()%></a></td>
      151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
      [i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 696:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      243. element = rs.getString(NUM_MESG);
      244. nbean.m_numMsg = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
```

```
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 697:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
...
243. element = rs.getString(NUM_MESG);
244. nbean.m_numMsg = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 698:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
...
243. element = rs.getString(NUM_MESG);
244. nbean.m_numMsg = (String) element;
...
```

```
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 699:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java

...
258. element = rs.getString(ANS_BY);
259. nbean.m_responder = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
=notices[i].getId()%></a></td>
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 700:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
258. element = rs.getString(ANS_BY);
259. nbean.m_responder = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 701:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
258. element = rs.getString(ANS_BY);
259. nbean.m_responder = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 702:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      258. element = rs.getString(ANS_BY);
      259. nbean.m_responder = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 703:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      264. element = rs.getString(INTERFACE);
      265. nbean.m_interfaceID = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
      =notices[i].getId()%></a></td>
      151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
      [i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 704:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      264. element = rs.getString(INTERFACE);
      265. nbean.m_interfaceID = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
      [i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 705:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
```

```
SQLException //notificationmodel.java

      ...
      264. element = rs.getString(INTERFACE);
      265. nbean.m_interfaceID = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp

      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 706:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java

      ...
      264. element = rs.getString(INTERFACE);
      265. nbean.m_interfaceID = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```

```
1. <!-- //list.jsp

      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 707:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
    ...
    274. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
    where serviceid = " + nbean.m_serviceId);
    ...
    277. element = rstmp.getString("servicename");
    ...
    279. nbean.m_serviceName = (String) element;
    ...
    286. vector.addElement(nbean);
    ...
    297. notices[i] = (Notification) vector.elementAt(i);
    ...
    300. return notices;
```

▼

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
    ...
    405. notices = rs2NotifyBean(conn, rs);
    ...
    417. return (notices);
```

▼

```
1. <!-- //list.jsp
    ...
    58. Notification[] notices = this.model.getOutstandingNotices( username );
    ...
    150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
    =notices[i].getId()%></a></td>
    151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
    [i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 708:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
    ...
    274. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
    where serviceid = " + nbean.m_serviceId);
    ...
    277. element = rstmp.getString("servicename");
    ...
    279. nbean.m_serviceName = (String) element;
    ...
    286. vector.addElement(nbean);
    ...
    297. notices[i] = (Notification) vector.elementAt(i);
    ...
    300. return notices;
```

▼

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 709:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
274. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
where serviceid = " + nbean.m_serviceId);
...
277. element = rstmp.getString("servicename");
...
279. nbean.m_serviceName = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
152. <td><%=notices[i].getTimeSent()%></td>
```

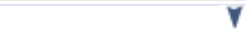
[Table of Contents] [Description]

Path 710:

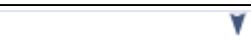
Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      274. ResultSet rstmp = stmttmp.executeQuery("SELECT servicename from service
      where serviceid = " + nbean.m_serviceId);
      ...
      277. element = rstmp.getString("servicename");
      ...
      279. nbean.m_serviceName = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```



```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```



```
1. <!-- //list.jsp
      ...
      58. Notification[] notices = this.model.getOutstandingNotices( username );
      ...
      153. <td><%=notices[i].getTextMessage()%></td>
```

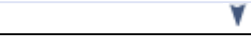
[Table of Contents] [Description]

Path 711:

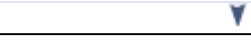
Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
      ...
      277. element = rstmp.getString("servicename");
      ...
      279. nbean.m_serviceName = (String) element;
      ...
      286. vector.addElement(nbean);
      ...
      297. notices[i] = (Notification) vector.elementAt(i);
      ...
      300. return notices;
```



```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
      ...
      405. notices = rs2NotifyBean(conn, rs);
      ...
      417. return (notices);
```



```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
=notices[i].getId()%></a></td>
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 712:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
277. element = rstmp.getString("servicename");
...
279. nbean.m_serviceName = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

▼

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

▼

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 713:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
277. element = rstmp.getString("servicename");
...
279. nbean.m_serviceName = (String) element;
...
286. vector.addElement(nbean);
...
```

```
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```



```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```



```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 714:

Query Name - **Stored_XSS**

Severity - **Critical**

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java

...
277. element = rstmp.getString("servicename");
...
279. nbean.m_serviceName = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```



```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```



```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 715:

Query Name - **Stored_XSS**

Severity - **Critical**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
403. ResultSet rs = pstmt.executeQuery();
...
405. notices = rs2NotifyBean(conn, rs);
```

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
150. <td><a HREF="notification/detail.jsp?notice=<%=notices[i].getId()%>"><%
=notices[i].getId()%></a></td>
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 716:

Query Name - **Stored_XSS**

Severity - **Critical**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
403. ResultSet rs = pstmt.executeQuery();
...
405. notices = rs2NotifyBean(conn, rs);
```

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
```

```
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
151. <td><a href="event/detail.jsp?id=<%=notices[i].getEventId()%>"><%=notices
[i].getEventId()%></a></td>
```

[Table of Contents] [Description]

Path 717:

Query Name - **Stored_XSS**

Severity - **Critical**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
403. ResultSet rs = pstmt.executeQuery();
...
405. notices = rs2NotifyBean(conn, rs);
```

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java

...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java

...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

```
1. <!-- //list.jsp

...
58. Notification[] notices = this.model.getOutstandingNotices( username );
```

```
...
152. <td><%=notices[i].getTimeSent()%></td>
```

[Table of Contents] [Description]

Path 718:

Query Name - **Stored_XSS**

Severity - **Critical**

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
403. ResultSet rs = pstmt.executeQuery();
...
405. notices = rs2NotifyBean(conn, rs);
```

▼

```
230. protected Notification[] rs2NotifyBean(Connection conn, ResultSet rs) throws
SQLException //notificationmodel.java
```

```
...
240. Object element = rs.getString(TXT_MESG);
241. nbean.m_txtMsg = (String) element;
...
286. vector.addElement(nbean);
...
297. notices[i] = (Notification) vector.elementAt(i);
...
300. return notices;
```

▼

```
395. public Notification[] getOutstandingNotices(String name) throws
SQLException //notificationmodel.java
```

```
...
405. notices = rs2NotifyBean(conn, rs);
...
417. return (notices);
```

▼

```
1. <!-- //list.jsp
```

```
...
58. Notification[] notices = this.model.getOutstandingNotices( username );
...
153. <td><%=notices[i].getTextMessage()%></td>
```

[Table of Contents] [Description]

Path 719:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
544. outage.ipAddress = rs.getString("ipaddr");
...
589. list.add(outage);
...
```

```
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

    ...
    254. Outage[] outages = rs2Outages(rs);
    ...
    257. outage = outages[0];
    ...
    266. return outage;
```

```
1. <!-- //detail.jsp

    ...
    64. Outage outage = OutageFactory.getOutage( outageId );
    ...
    100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 720:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java

    ...
    544. outage.ipAddress = rs.getString("ipaddr");
    ...
    589. list.add(outage);
    ...
    592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
    ...
    594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

    ...
    254. Outage[] outages = rs2Outages(rs);
    ...
    257. outage = outages[0];
    ...
    266. return outage;
```

```
1. <!-- //detail.jsp

    ...
    64. Outage outage = OutageFactory.getOutage( outageId );
    ...
    107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
    =outage.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 721:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      544. outage.ipAddress = rs.getString("ipaddr");
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp
      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      114. <td><%=DATE_FORMAT.format(outage.getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 722:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      544. outage.ipAddress = rs.getString("ipaddr");
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
117. <td><a href="event/detail.jsp?id=<%=outage.getLostServiceEventId()%>"><%
=outage.getLostServiceEventId()%></a></td>
```

[Table of Contents] [Description]

Path 723:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
544. outage.ipAddress = rs.getString("ipaddr");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

▼

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
125. <a href="element/interface.jsp?node=<%=outage.getNodeId()%>&intf=<%
=outage.getIpAddress()%>"><%=outage.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 724:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
544. outage.ipAddress = rs.getString("ipaddr");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        127. <%=outage.getIpAddress()%>
```

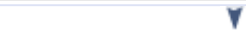
[Table of Contents] [Description]

Path 725:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
        ...
        552. outage.hostname = rs.getString("iphostname"); // from ipinterface
        ...
        589. list.add(outage);
        ...
        592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
        ...
        594. return outages;
```



```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 726:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
552. outage.hostname = rs.getString("iphostname"); // from ipinterface
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
=outage.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 727:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
...
552. outage.hostname = rs.getString("iphostname"); // from ipinterface
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
114. <td><%=DATE_FORMAT.format(outage.getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 728:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      552. outage.hostname = rs.getString("iphostname"); // from ipinterface
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp
      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      117. <td><a href="event/detail.jsp?id=<%=outage.getLostServiceEventId()%>"><%
      =outage.getLostServiceEventId()%></a></td>
```

[Table of Contents] [Description]

Path 729:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      552. outage.hostname = rs.getString("iphostname"); // from ipinterface
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
```

```
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
125. <a href="element/interface.jsp?node=<%=outage.getNodeId()%>&intf=<%
=outage.getIpAddress()%>"><%=outage.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 730:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
...
552. outage.hostname = rs.getString("iphostname"); // from ipinterface
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
127. <%=outage.getIpAddress()%>
```

[Table of Contents] [Description]

Path 731:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
...
556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
...
```

```
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp

...
64. Outage outage = OutageFactory.getOutage( outageId );
...
100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 732:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java

...
556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp

...
64. Outage outage = OutageFactory.getOutage( outageId );
...
107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
=outage.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 733:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

▼

```
1. <!-- //detail.jsp
      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      114. <td><%=DATE_FORMAT.format(outage.getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 734:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

▼

```
1. <!-- //detail.jsp
      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      117. <td><a href="event/detail.jsp?id=<%=outage.getLostServiceEventId()%>"><%
      =outage.getLostServiceEventId()%></a></td>
```

[Table of Contents] [Description]

Path 735:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp
      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      125. <a href="element/interface.jsp?node=<%=outage.getNodeId()%>&intf=<%
      =outage.getIpAddress()%>"><%=outage.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 736:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
      ...
      556. outage.nodeLabel = rs.getString("nodelabel"); // from node table
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        127. <%=outage.getIpAddress()%>
```

[Table of Contents] [Description]

Path 737:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
        ...
        559. outage.serviceName = rs.getString("servicename"); // from service
        ...
        589. list.add(outage);
        ...
        592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
        ...
        594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 738:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
```

```
SQLException //outagefactory.java

      ...
      559. outage.serviceName = rs.getString("servicename"); // from service
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp

      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
      107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
      =outage.getNodeLabel()%></a>
```

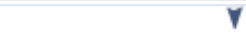
[Table of Contents] [Description]

Path 739:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java

      ...
      559. outage.serviceName = rs.getString("servicename"); // from service
      ...
      589. list.add(outage);
      ...
      592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
      ...
      594. return outages;
```



```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java

      ...
      254. Outage[] outages = rs2Outages(rs);
      ...
      257. outage = outages[0];
      ...
      266. return outage;
```

```
1. <!-- //detail.jsp

      ...
      64. Outage outage = OutageFactory.getOutage( outageId );
      ...
```

```
114. <td><%=DATE_FORMAT.format(outage.getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 740:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
559. outage.serviceName = rs.getString("servicename"); // from service
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
117. <td><a href="event/detail.jsp?id=<%=outage.getLostServiceEventId()%>"><%
=outage.getLostServiceEventId()%></a></td>
```

[Table of Contents] [Description]

Path 741:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
559. outage.serviceName = rs.getString("servicename"); // from service
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
125. <a href="element/interface.jsp?node=<%=outage.getNodeId()%>&intf=<%
=outage.getIpAddress()%>"><%=outage.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 742:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
...
559. outage.serviceName = rs.getString("servicename"); // from service
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
127. <%=outage.getIpAddress()%>
```

[Table of Contents] [Description]

Path 743:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
("answeredby");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
100. <h3>Outage: <%=outage.getId()%></h3>
```

[Table of Contents] [Description]

Path 744:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
...
587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
("answeredby");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
```

```
107. <a href="element/node.jsp?node=<%=outage.getNodeId()%>"><%
=outage.getNodeLabel()%></a>
```

[Table of Contents] [Description]

Path 745:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
("answeredby");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

▼

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
114. <td><%=DATE_FORMAT.format(outage.getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 746:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
("answeredby");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```

▼

```
245. public static Outage getOutage(int outageId) throws
```

```
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        117. <td><a href="event/detail.jsp?id=<%=outage.getLostServiceEventId()%>"><%
        =outage.getLostServiceEventId()%></a></td>
```

[Table of Contents] [Description]

Path 747:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
        ...
        587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
        ("answeredby");
        ...
        589. list.add(outage);
        ...
        592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
        ...
        594. return outages;
```

```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
        ...
        254. Outage[] outages = rs2Outages(rs);
        ...
        257. outage = outages[0];
        ...
        266. return outage;
```

```
1. <!-- //detail.jsp
        ...
        64. Outage outage = OutageFactory.getOutage( outageId );
        ...
        125. <a href="element/interface.jsp?node=<%=outage.getNodeId()%>&intf=<%
        =outage.getIpAddress()%>"><%=outage.getIpAddress()%></a>
```

[Table of Contents] [Description]

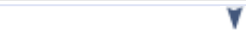
Path 748:

Query Name - **Stored_XSS**

Severity - **Critical**

```
531. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagefactory.java
```

```
...
587. outage.lostServiceNotificationAcknowledgedBy = rs.getString
("answeredby");
...
589. list.add(outage);
...
592. outages = (Outage[]) list.toArray(new Outage[list.size()]);
...
594. return outages;
```



```
245. public static Outage getOutage(int outageId) throws
SQLException //outagefactory.java
```

```
...
254. Outage[] outages = rs2Outages(rs);
...
257. outage = outages[0];
...
266. return outage;
```

```
1. <!-- //detail.jsp
```

```
...
64. Outage outage = OutageFactory.getOutage( outageId );
...
127. <%=outage.getIpAddress()%>
```

[Table of Contents] [Description]

Path 749:

Query Name - **Stored_XSS**

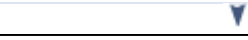
Severity - **Critical**

```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
```

```
...
362. String nodeLabel = rs.getString("nodelabel");
...
364. list.add(new OutageSummary(nodeId, nodeLabel, downDate));
```

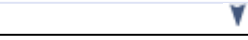
```
50. public OutageSummary(int nodeId, String nodeLabel, Date
timeDown) //outagesummary.java
```

```
...
56. this.nodeLabel = nodeLabel;
```



```
347. public OutageSummary[] getCurrentOutageSummaries() throws
SQLException //outagemodel.java
```

```
...
364. list.add(new OutageSummary(nodeId, nodeLabel, downDate));
...
370. summaries = (OutageSummary[]) list.toArray(new OutageSummary[list.size
()]);
...
375. return summaries;
```



```
1. <!-- //servicesdown-box.jsp
      ...
      53. OutageSummary[] summaries = this.model.getCurrentOutageSummaries();
      ...
      63. <% OutageSummary summary = summaries[i];
      64. String nodeLabel = summary.getNodeLabel();
```



```
69. public String getNodeLabel() //outagesummary.java
      70. return (this.nodeLabel);
```



```
1. <!-- //servicesdown-box.jsp
      ...
      64. String nodeLabel = summary.getNodeLabel();
      ...
      68. <td align="left"><a href="element/node.jsp?node=<%=nodeId%>"><nobr><%
      =nodeLabel%></nobr></a></td>
```

[Table of Contents] [Description]

Path 750:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      438. element = rs.getString("ipaddr");
      439. outage.ipAddress = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
```

```
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 751:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
...
438. element = rs.getString("ipaddr");
439. outage.ipAddress = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

▼

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
...
204. outages = rs2Outages(rs);
...
212. return outages;
```

▼

```
1. <!-- //nodeoutages-box.jsp
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 752:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      444. element = rs.getString("nodeLabel");
      445. outage.nodeLabel = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
      [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 753:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      444. element = rs.getString("nodeLabel");
      445. outage.nodeLabel = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
     ...
     265. outages = rs2Outages(rs);
     ...
     273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
     ...
     75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
     yesterday);
     ...
     98. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 754:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     444. element = rs.getString("nodeLabel");
     445. outage.nodeLabel = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
```

```
...
265. outages = rs2Outages(rs);
...
273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 755:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
...
444. element = rs.getString("nodeLabel");
445. outage.nodeLabel = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
...
265. outages = rs2Outages(rs);
...
273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 756:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      444. element = rs.getString("nodeLabel");
      445. outage.nodeLabel = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 757:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      444. element = rs.getString("nodeLabel");
      445. outage.nodeLabel = (String) element;
```

```
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
     ...
     69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
     ...
     95. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
     [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
     [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 758:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     444. element = rs.getString("nodeLabel");
     445. outage.nodeLabel = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
```

```
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      96. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 759:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      444. element = rs.getString("nodeLabel");
      445. outage.nodeLabel = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
```

```
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
101. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 760:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
444. element = rs.getString("nodeLabel");
445. outage.nodeLabel = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
```

```
...
204. outages = rs2Outages(rs);
...
212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
```

```
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
103. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 761:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    444. element = rs.getString("nodeLabel");
    445. outage.nodeLabel = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
    ...
    330. outages = rs2Outages(rs);
    ...
    338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
    ...
    83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
    serviceId, yesterday);
    ...
    102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
    [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 762:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    444. element = rs.getString("nodeLabel");
    445. outage.nodeLabel = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```



```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
     ...
     330. outages = rs2Outages(rs);
     ...
     338. return outages;
```



```
1. <!-- //serviceoutages-box.jsp
     ...
     83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
     serviceId, yesterday);
     ...
     103. <td <%=(outages[i].getRegainedServiceTime() == null) ?
     "bgcolor=\"red\"" : ""%>><%=(outages[i].getRegainedServiceTime() == null) ?
     "<b>DOWN</b>" : org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 763:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     444. element = rs.getString("nodeLabel");
     445. outage.nodeLabel = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```



```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
```

```
...
330. outages = rs2Outages(rs);
...
338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 764:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
...
265. outages = rs2Outages(rs);
...
273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
```

```
[i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 765:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

▼

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
```

```
...
265. outages = rs2Outages(rs);
...
273. return outages;
```

▼

```
1. <!-- //interfaceoutages-box.jsp
```

```
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
98. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 766:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
     ...
     265. outages = rs2Outages(rs);
     ...
     273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
     ...
     75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
     yesterday);
     ...
     102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 767:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     447. element = rs.getString("iphostname");
     448. outage.hostname = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
     ...
     265. outages = rs2Outages(rs);
     ...
     273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
     ...
     75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
     yesterday);
     ...
     104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
     [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 768:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     447. element = rs.getString("iphostname");
     448. outage.hostname = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 769:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      447. element = rs.getString("iphostname");
      448. outage.hostname = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

▼

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

▼

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      95. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
      [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 770:

Query Name - **Stored_XSS**

Severity

- Critical

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      447. element = rs.getString("iphostname");
      448. outage.hostname = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      96. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 771:

Query Name - **Stored_XSS**

Severity - Critical

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      447. element = rs.getString("iphostname");
      448. outage.hostname = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

▼

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java

     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

▼

```
1. <!-- //nodeoutages-box.jsp

     ...
     69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
     ...
     101. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 772:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java

     ...
     447. element = rs.getString("iphostname");
     448. outage.hostname = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

▼

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java

     ...
```

```
204. outages = rs2Outages(rs);
...
212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp

...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
103. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 773:

Query Name - **Stored_XSS**

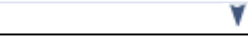
Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java

...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java

...
330. outages = rs2Outages(rs);
...
338. return outages;
```



```
1. <!-- //serviceoutages-box.jsp

...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 774:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
```

```
...
330. outages = rs2Outages(rs);
...
338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
```

```
...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
103. <td <%=(outages[i].getRegainedServiceTime() == null) ?
"bgcolor=\"red\"" : ""%>><%=(outages[i].getRegainedServiceTime() == null) ?
"<b>DOWN</b>" : org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 775:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
447. element = rs.getString("iphostname");
448. outage.hostname = (String) element;
```

```
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
     ...
     330. outages = rs2Outages(rs);
     ...
     338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
     ...
     83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
     serviceId, yesterday);
     ...
     104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
     [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 776:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     450. element = rs.getString("servicename");
     451. outage.serviceName = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
```

```
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```



```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
      [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 777:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      98. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 778:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

▼

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```

▼

```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 779:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
      ...
      265. outages = rs2Outages(rs);
      ...
      273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
      ...
      75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
      yesterday);
      ...
      104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
      [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 780:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
     ...
     69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
     ...
     94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
     [i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 781:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     450. element = rs.getString("servicename");
     451. outage.serviceName = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
```

```
204. outages = rs2Outages(rs);
...
212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp

...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
95. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
[i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
[i].getServiceName()%></a></td>
```

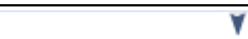
[Table of Contents] [Description]

Path 782:

Query Name - **Stored_XSS**

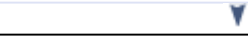
Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java

...
450. element = rs.getString("servicename");
451. outage.serviceName = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

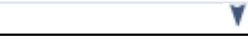


```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```



```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java

...
204. outages = rs2Outages(rs);
...
212. return outages;
```



```
1. <!-- //nodeoutages-box.jsp

...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
96. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 783:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      101. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 784:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
```

```
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
     ...
     69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
     ...
     103. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
     [i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 785:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     450. element = rs.getString("servicename");
     451. outage.serviceName = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
      ...
      330. outages = rs2Outages(rs);
      ...
      338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
      ...
      83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
      serviceId, yesterday);
      ...
      102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 786:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      450. element = rs.getString("servicename");
      451. outage.serviceName = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
      ...
      330. outages = rs2Outages(rs);
      ...
      338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
      ...
      83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
      serviceId, yesterday);
      ...
```

```
103. <td <%=(outages[i].getRegainedServiceTime() == null) ?
"bgcolor=\"red\"" : ""%>><%=(outages[i].getRegainedServiceTime() == null) ?
"<b>DOWN</b>" : org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 787:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
450. element = rs.getString("servicename");
451. outage.serviceName = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

▼

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

▼

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

▼

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
```

```
...
330. outages = rs2Outages(rs);
...
338. return outages;
```

▼

```
1. <!-- //serviceoutages-box.jsp
```

```
...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 788:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    478. element = rs.getString("answeredby");
    479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
    ...
    265. outages = rs2Outages(rs);
    ...
    273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
    ...
    75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
    yesterday);
    ...
    97. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
    [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
    [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 789:

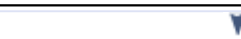
Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    478. element = rs.getString("answeredby");
    479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```



```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
     ...
     265. outages = rs2Outages(rs);
     ...
     273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
     ...
     75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
     yesterday);
     ...
     98. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
     [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 790:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     478. element = rs.getString("answeredby");
     479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
```

```
...
265. outages = rs2Outages(rs);
...
273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 791:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
...
478. element = rs.getString("answeredby");
479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
417. return rs2Outages(rs, true);
```

```
249. public Outage[] getOutagesForInterface(int nodeId, String ipAddr, Date time)
throws SQLException //outagemodel.java
...
265. outages = rs2Outages(rs);
...
273. return outages;
```

```
1. <!-- //interfaceoutages-box.jsp
...
75. Outage[] outages = this.model.getOutagesForInterface(nodeId, ipAddr,
yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 792:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      478. element = rs.getString("answeredby");
      479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      94. <td><a href="element/interface.jsp?node=<%=nodeId%>&intf=<%=outages
      [i].getIpAddress()%>"><%=outages[i].getIpAddress()%></a></td>
```

[Table of Contents] [Description]

Path 793:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      478. element = rs.getString("answeredby");
      479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
```

```
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
     417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
     ...
     204. outages = rs2Outages(rs);
     ...
     212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
     ...
     69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
     ...
     95. <td><a href="element/service.jsp?node=<%=nodeId%>&intf=<%=outages
     [i].getIpAddress()%>&service=<%=outages[i].getServiceId()%>"><%=outages
     [i].getServiceName()%></a></td>
```

[Table of Contents] [Description]

Path 794:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
     ...
     478. element = rs.getString("answeredby");
     479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
     ...
     483. vector.addElement(outage);
     ...
     486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
     ...
     488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
     421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
```

```
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
      ...
      69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
      ...
      96. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
      [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 795:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
      ...
      478. element = rs.getString("answeredby");
      479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
      ...
      483. vector.addElement(outage);
      ...
      486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
      ...
      488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
      421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
      417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
      ...
      204. outages = rs2Outages(rs);
      ...
      212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
```

```
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
101. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
[i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 796:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
```

```
...
478. element = rs.getString("answeredby");
479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
...
483. vector.addElement(outage);
...
486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
...
488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
```

```
421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
```

```
417. return rs2Outages(rs, true);
```

```
189. public Outage[] getOutagesForNode(int nodeId, Date time) throws
SQLException //outagemodel.java
```

```
...
204. outages = rs2Outages(rs);
...
212. return outages;
```

```
1. <!-- //nodeoutages-box.jsp
```

```
...
69. Outage[] outages = this.model.getOutagesForNode(nodeId, yesterday);
...
103. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 797:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    478. element = rs.getString("answeredby");
    479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```

```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```

```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
    ...
    330. outages = rs2Outages(rs);
    ...
    338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
    ...
    83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
    serviceId, yesterday);
    ...
    102. <td><%=org.opennms.netmgt.EventConstants.formatToUIString(outages
    [i].getLostServiceTime())%></td>
```

[Table of Contents] [Description]

Path 798:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    478. element = rs.getString("answeredby");
    479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```

```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
```

```
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```



```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
    ...
    330. outages = rs2Outages(rs);
    ...
    338. return outages;
```



```
1. <!-- //serviceoutages-box.jsp
    ...
    83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
    serviceId, yesterday);
    ...
    103. <td <%=(outages[i].getRegainedServiceTime() == null) ?
    "bgcolor=\"red\"" : ""%>><%=(outages[i].getRegainedServiceTime() == null) ?
    "<b>DOWN</b>" : org.opennms.netmgt.EventConstants.formatToUIString(outages
    [i].getRegainedServiceTime())%></td>
```

[Table of Contents] [Description]

Path 799:

Query Name - **Stored_XSS**

Severity - **Critical**

```
428. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime, boolean includesNotifications) throws
SQLException //outagemodel.java
    ...
    478. element = rs.getString("answeredby");
    479. outage.lostServiceNotificationAcknowledgedBy = (String) element;
    ...
    483. vector.addElement(outage);
    ...
    486. outages = (Outage[]) vector.toArray(new Outage[vector.size()]);
    ...
    488. return outages;
```



```
420. protected static Outage[] rs2Outages(ResultSet rs, boolean
includesRegainedTime) throws SQLException //outagemodel.java
    421. return rs2Outages(rs, includesRegainedTime, false);
```



```
416. protected static Outage[] rs2Outages(ResultSet rs) throws
SQLException //outagemodel.java
    417. return rs2Outages(rs, true);
```



```
313. public Outage[] getOutagesForService(int nodeId, String ipAddr, int serviceId,
Date time) throws SQLException //outagemodel.java
```

```
...
330. outages = rs2Outages(rs);
...
338. return outages;
```

```
1. <!-- //serviceoutages-box.jsp
...
83. Outage[] outages = this.model.getOutagesForService(nodeId, ipAddr,
serviceId, yesterday);
...
104. <td><a href="outage/detail.jsp?id=<%=outages[i].getId()%>"><%=outages
[i].getId()%></a></td>
```

[Table of Contents] [Description]

Path 800:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
...
400. node.nodeLabel = rs.getString("nodeLabel");
...
402. nodeList.add(node);
...
414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
[nodeList.size()]);
...
417. return (nodes);
```

```
1. <!-- //performance-box.jsp
...
61. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
...
86. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 801:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
...
400. node.nodeLabel = rs.getString("nodeLabel");
...
402. nodeList.add(node);
...
414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
[nodeList.size()]);
...
417. return (nodes);
```

```
1. <!-- //custom_graph1.jsp
      ...
      56. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
      ...
      132. <option value="<%=nodes[i].nodeId%>" SELECTED><%=nodes[i].nodeLabel%
      ></option>
```

[Table of Contents] [Description]

Path 802:

Query Name - **Stored_XSS**

Severity - **Critical**

```
363. public QueryableNode[] getQueryableNodes() throws
SQLException //performancemodel.java
      ...
      400. node.nodeLabel = rs.getString("nodeLabel");
      ...
      402. nodeList.add(node);
      ...
      414. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
      [nodeList.size()]);
      ...
      417. return (nodes);
```

▼

```
1. <!-- //custom_graph1.jsp
      ...
      56. PerformanceModel.QueryableNode[] nodes = this.model.getQueryableNodes();
      ...
      134. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 803:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
      ...
      402. node.nodeLabel = rs.getString("nodeLabel");
      ...
      404. nodeList.add(node);
      ...
      419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
      [nodeList.size()]);
      ...
      422. return (nodes);
```

▼

```
1. <!-- //response-box.jsp
      ...
      65. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
      ...
      90. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 804:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
    ...
    402. node.nodeLabel = rs.getString("nodeLabel");
    ...
    404. nodeList.add(node);
    ...
    419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
    [nodeList.size()]);
    ...
    422. return (nodes);
```

```
1. <!-- //index.jsp
    ...
    60. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
    ...
    153. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 805:

Query Name - **Stored_XSS**

Severity - **Critical**

```
365. public QueryableNode[] getQueryableNodes() throws
SQLException //responsetimemodel.java
    ...
    402. node.nodeLabel = rs.getString("nodeLabel");
    ...
    404. nodeList.add(node);
    ...
    419. nodes = (QueryableNode[]) nodeList.toArray(new QueryableNode
    [nodeList.size()]);
    ...
    422. return (nodes);
```

```
1. <!-- //index.jsp
    ...
    60. ResponseTimeModel.QueryableNode[] nodes = this.model.getQueryableNodes();
    ...
    169. <option value="<%=nodes[i].nodeId%>"><%=nodes[i].nodeLabel%></option>
```

[Table of Contents] [Description]

Path 806:

Query Name - **Stored_XSS**

Severity - **Critical**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
```

```
SQLException //responsetimemodel.java

    ...
450. String ipAddr = rs.getString("ipaddr");
    ...
454. intfs.add(ipAddr);
    ...
463. return intfs;
```

```
1. <!-- //chooseinterface.jsp

    ...
71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
    ...
148. <option value="<%=intfs.get(i)%>" <%=(i==0) ? "selected" : ""%>><%
=intfs.get(i)%></option>
```

[Table of Contents] [Description]

Path 807:

Query Name - **Stored_XSS**

Severity - **Critical**

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java

    ...
450. String ipAddr = rs.getString("ipaddr");
    ...
454. intfs.add(ipAddr);
    ...
463. return intfs;
```

```
1. <!-- //chooseinterface.jsp

    ...
71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
    ...
148. <option value="<%=intfs.get(i)%>" <%=(i==0) ? "selected" : ""%>><%
=intfs.get(i)%></option>
```

[Table of Contents] [Description]

Path 808:

Query Name - **Stored_XSS**

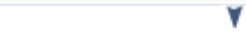
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

    ...
453. element = rs.getObject("nodeID");
    ...
455. vul.nodeId = (Integer) element;
    ...
526. list.add(vul);
    ...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
    ...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```



```
1. <!-- //detail.jsp
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

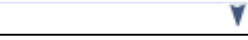
[Table of Contents] [Description]

Path 809:

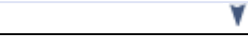
Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      459. element = rs.getString("ipAddr");
      460. vul.ipAddr = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```



```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
```

```
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

▼

```
174. public String getNodeLabel() //vulnerability.java
    175. return (this.nodeLabel);
```

▼

```
1. <!-- //detail.jsp
    ...
    107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
    ()%></a>
```

[Table of Contents] [Description]

Path 810:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
    ...
    459. element = rs.getString("ipAddr");
    460. vul.ipAddr = (String) element;
    ...
    526. list.add(vul);
    ...
    529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
    ...
    531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
    ...
    253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
    ...
    258. vul = vuls[0];
    ...
    267. return vul;
```

▼

```
1. <!-- //detail.jsp
    ...
    64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
    ...
    116. <%=vul.getProtocol()%>
```

▼

```
214. public String getProtocol() //vulnerability.java
    215. return (this.protocol);
```

▼

```
1. <!-- //detail.jsp
    ...
    116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 811:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      459. element = rs.getString("ipAddr");
      460. vul.ipAddr = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

▼

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

▼

```
143. public Date getCreateTime() //vulnerability.java
      144. return (this.createTime);
```

▼

```
1. <!-- //detail.jsp
      ...
      124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 812:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      459. element = rs.getString("ipAddr");
      460. vul.ipAddr = (String) element;
      ...
      526. list.add(vul);
      ...
```

```
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp

...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp

...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 813:

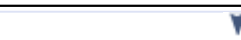
Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

...
459. element = rs.getString("ipAddr");
460. vul.ipAddr = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      132. <%=vul.getIpAddress()%>
```



```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      132. <%=vul.getIpAddress()%>
```

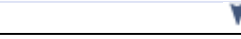
[Table of Contents] [Description]

Path 814:

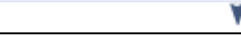
Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      463. element = rs.getObject("serviceID");
      ...
      465. vul.serviceId = (Integer) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

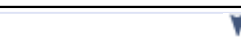


```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```



```
1. <!-- //detail.jsp
```

```
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

[Table of Contents] [Description]

Path 815:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
```

```
...
463. element = rs.getObject("serviceID");
...
465. vul.serviceId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

▼

```
1. <!-- //detail.jsp
```

```
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
116. <%=vul.getProtocol()%>
```

▼

```
214. public String getProtocol() //vulnerability.java
```

```
215. return (this.protocol);
```

▼

```
1. <!-- //detail.jsp
```

```
...
116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 816:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
```

```
...
463. element = rs.getObject("serviceID");
...
465. vul.serviceId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
```

```
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
```

```
144. return (this.createTime);
```

```
1. <!-- //detail.jsp
```

```
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 817:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
```

```
...
463. element = rs.getObject("serviceID");
...
465. vul.serviceId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 818:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
463. element = rs.getObject("serviceID");
...
465. vul.serviceId = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
```

```
132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
...
132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 819:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
469. element = rs.getString("nodeLabel");
470. vul.nodeLabel = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
175. return (this.nodeLabel);
```

```
1. <!-- //detail.jsp
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

[Table of Contents] [Description]

Path 820:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      469. element = rs.getString("nodeLabel");
      470. vul.nodeLabel = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
      215. return (this.protocol);
```

```
1. <!-- //detail.jsp
      ...
      116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 821:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      469. element = rs.getString("nodeLabel");
      470. vul.nodeLabel = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
```

```
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp

...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
144. return (this.createTime);
```

```
1. <!-- //detail.jsp

...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 822:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

...
469. element = rs.getString("nodeLabel");
470. vul.nodeLabel = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
```

```
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

▼

```
179. public String getIpAddress() //vulnerability.java
     180. return (this.ipAddr);
```

▼

```
1. <!-- //detail.jsp
     ...
     130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
     =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 823:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
     ...
     469. element = rs.getString("nodeLabel");
     470. vul.nodeLabel = (String) element;
     ...
     526. list.add(vul);
     ...
     529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
     ...
     531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
     ...
     253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
     ...
     258. vul = vuls[0];
     ...
     267. return vul;
```

▼

```
1. <!-- //detail.jsp
     ...
     64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
     ...
     132. <%=vul.getIpAddress()%>
```

▼

```
179. public String getIpAddress() //vulnerability.java
     180. return (this.ipAddr);
```

▼

```
1. <!-- //detail.jsp
     ...
```

```
132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 824:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
    ...
    473. element = rs.getString("serviceName");
    474. vul.serviceName = (String) element;
    ...
    526. list.add(vul);
    ...
    529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
    ...
    531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
    ...
    253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
    ...
    258. vul = vuls[0];
    ...
    267. return vul;
```

▼

```
1. <!-- //detail.jsp
    ...
    64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
    ...
    107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
    ()%></a>
```

▼

```
174. public String getNodeLabel() //vulnerability.java
    175. return (this.nodeLabel);
```

▼

```
1. <!-- //detail.jsp
    ...
    107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
    ()%></a>
```

[Table of Contents] [Description]

Path 825:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
    ...
```

```
473. element = rs.getString("serviceName");
474. vul.serviceName = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
215. return (this.protocol);
```

```
1. <!-- //detail.jsp
...
116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 826:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
473. element = rs.getString("serviceName");
474. vul.serviceName = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
```

```
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
144. return (this.createTime);
```

```
1. <!-- //detail.jsp
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 827:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
473. element = rs.getString("serviceName");
474. vul.serviceName = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 828:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      473. element = rs.getString("serviceName");
      474. vul.serviceName = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 829:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
```

```
SQLException //vulnerabilityfactory.java

      ...
      496. element = rs.getString("logmsg");
      497. vul.logMessage = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp

      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```

```
1. <!-- //detail.jsp

      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

[Table of Contents] [Description]

Path 830:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

      ...
      496. element = rs.getString("logmsg");
      497. vul.logMessage = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
```

```
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
215. return (this.protocol);
```

```
1. <!-- //detail.jsp
...
116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 831:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
496. element = rs.getString("logmsg");
497. vul.logMessage = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
     144. return (this.createTime);
```



```
1. <!-- //detail.jsp
     ...
     124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 832:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
     ...
     496. element = rs.getString("logmsg");
     497. vul.logMessage = (String) element;
     ...
     526. list.add(vul);
     ...
     529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
     ...
     531. return vuls;
```



```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
     ...
     253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
     ...
     258. vul = vuls[0];
     ...
     267. return vul;
```



```
1. <!-- //detail.jsp
     ...
     64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
     ...
     130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
     =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```



```
179. public String getIpAddress() //vulnerability.java
     180. return (this.ipAddr);
```



```
1. <!-- //detail.jsp
     ...
     130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
     =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 833:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      496. element = rs.getString("logmsg");
      497. vul.logMessage = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
      ...
      132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 834:

Query Name - **Stored_XSS**

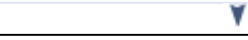
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      509. element = rs.getString("descr");
      510. vul.description = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```



```
1. <!-- //detail.jsp
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

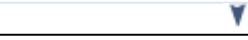
[Table of Contents] [Description]

Path 835:

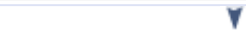
Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      509. element = rs.getString("descr");
      510. vul.description = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```



```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
```

```
...
116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
     215. return (this.protocol);
```

```
1. <!-- //detail.jsp
     ...
     116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 836:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
     ...
     509. element = rs.getString("descr");
     510. vul.description = (String) element;
     ...
     526. list.add(vul);
     ...
     529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
     ...
     531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
     ...
     253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
     ...
     258. vul = vuls[0];
     ...
     267. return vul;
```

```
1. <!-- //detail.jsp
     ...
     64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
     ...
     124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
     144. return (this.createTime);
```

```
1. <!-- //detail.jsp
     ...
     124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 837:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      509. element = rs.getString("descr");
      510. vul.description = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

▼

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

▼

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```

▼

```
1. <!-- //detail.jsp
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 838:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      509. element = rs.getString("descr");
      510. vul.description = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
```

```
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp

...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp

...
132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 839:

Query Name - **Stored_XSS**

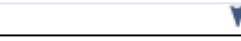
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

...
513. element = rs.getObject("port");
...
515. vul.port = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```



```
1. <!-- //detail.jsp
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

[Table of Contents] [Description]

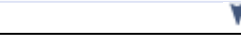
Path 840:

Query Name - **Stored_XSS**

Severity - **Critical**

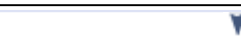
```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      513. element = rs.getObject("port");
      ...
      515. vul.port = (Integer) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
      215. return (this.protocol);
```



```
1. <!-- //detail.jsp
```

```
...
116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 841:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
     ...
     513. element = rs.getObject("port");
     ...
     515. vul.port = (Integer) element;
     ...
     526. list.add(vul);
     ...
     529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
     ...
     531. return vuls;
```

▼

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
     ...
     253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
     ...
     258. vul = vuls[0];
     ...
     267. return vul;
```

▼

```
1. <!-- //detail.jsp
     ...
     64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
     ...
     124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

▼

```
143. public Date getCreateTime() //vulnerability.java
     144. return (this.createTime);
```

▼

```
1. <!-- //detail.jsp
     ...
     124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 842:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
```

```
...
513. element = rs.getObject("port");
...
515. vul.port = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java

...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp

...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp

...
130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
=vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 843:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java

...
513. element = rs.getObject("port");
...
515. vul.port = (Integer) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
```

```
SQLException //vulnerabilityfactory.java
    ...
    253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
    ...
    258. vul = vuls[0];
    ...
    267. return vul;
```

```
1. <!-- //detail.jsp
    ...
    64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
    ...
    132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
    180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
    ...
    132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 844:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
    ...
    519. element = rs.getString("protocol");
    520. vul.protocol = (String) element;
    ...
    526. list.add(vul);
    ...
    529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
    ...
    531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
    ...
    253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
    ...
    258. vul = vuls[0];
    ...
    267. return vul;
```

```
1. <!-- //detail.jsp
    ...
    64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
    ...
    107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
    ()%></a>
```

```
174. public String getNodeLabel() //vulnerability.java
      175. return (this.nodeLabel);
```

```
1. <!-- //detail.jsp
      ...
      107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
      ()%></a>
```

[Table of Contents] [Description]

Path 845:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      519. element = rs.getString("protocol");
      520. vul.protocol = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
      215. return (this.protocol);
```

```
1. <!-- //detail.jsp
      ...
      116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 846:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      519. element = rs.getString("protocol");
      520. vul.protocol = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
      144. return (this.createTime);
```

```
1. <!-- //detail.jsp
      ...
      124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 847:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      519. element = rs.getString("protocol");
      520. vul.protocol = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```



```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```



```
1. <!-- //detail.jsp
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 848:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      519. element = rs.getString("protocol");
      520. vul.protocol = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```



```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```



```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
```

```
...
132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
180. return (this.ipAddr);
```

```
1. <!-- //detail.jsp
...
132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Path 849:

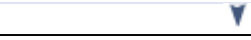
Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
523. element = rs.getString("cveEntry");
524. vul.cveEntry = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```



```
174. public String getNodeLabel() //vulnerability.java
175. return (this.nodeLabel);
```

```
1. <!-- //detail.jsp
...
107. <a href="element/node.jsp?node=<%=vul.getNodeId()%>"><%=vul.getNodeLabel
()%></a>
```

[Table of Contents] [Description]

Path 850:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      523. element = rs.getString("cveEntry");
      524. vul.cveEntry = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      116. <%=vul.getProtocol()%>
```

```
214. public String getProtocol() //vulnerability.java
      215. return (this.protocol);
```

```
1. <!-- //detail.jsp
      ...
      116. <%=vul.getProtocol()%>
```

[Table of Contents] [Description]

Path 851:

Query Name - **Stored_XSS**

Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      523. element = rs.getString("cveEntry");
      524. vul.cveEntry = (String) element;
      ...
      526. list.add(vul);
      ...
```

```
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
...
64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

```
143. public Date getCreateTime() //vulnerability.java
144. return (this.createTime);
```

```
1. <!-- //detail.jsp
...
124. <td valign="top"><%=DATE_FORMAT.format(vul.getCreateTime())%></td>
```

[Table of Contents] [Description]

Path 852:

Query Name - **Stored_XSS**

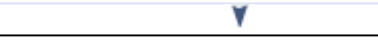
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
...
523. element = rs.getString("cveEntry");
524. vul.cveEntry = (String) element;
...
526. list.add(vul);
...
529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
...
531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
...
253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
...
258. vul = vuls[0];
...
267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```



```
1. <!-- //detail.jsp
      ...
      130. <a href="element/interface.jsp?node=<%=vul.getNodeId()%>&intf=<%
      =vul.getIpAddress()%>"><%=vul.getIpAddress()%></a>
```

[Table of Contents] [Description]

Path 853:

Query Name - **Stored_XSS**

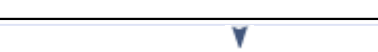
Severity - **Critical**

```
442. protected static Vulnerability[] rs2Vulnerabilities(ResultSet rs) throws
SQLException //vulnerabilityfactory.java
      ...
      523. element = rs.getString("cveEntry");
      524. vul.cveEntry = (String) element;
      ...
      526. list.add(vul);
      ...
      529. vuls = (Vulnerability[]) list.toArray(new Vulnerability[list.size()]);
      ...
      531. return vuls;
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      253. Vulnerability[] vuls = rs2Vulnerabilities(rs);
      ...
      258. vul = vuls[0];
      ...
      267. return vul;
```

```
1. <!-- //detail.jsp
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
      ...
      132. <%=vul.getIpAddress()%>
```

```
179. public String getIpAddress() //vulnerability.java
      180. return (this.ipAddr);
```



```
1. <!-- //detail.jsp
```

```
...
132. <%=vul.getIpAddress()%>
```

[Table of Contents] [Description]

Log Forging

CWE ID	117
Description	Writing unvalidated user input into log files can allow an attacker to forge log entries or inject malicious content into logs.
Likelihood of Exploit	Medium
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid, expected and appropriate input is processed by the system.
Demonstrative Examples	The following web application code attempts to read an integer value from a request object. If the value fails to parse as an integer, then the input is logged with an error message indicating what happened.

```
...
String val = request.getParameter("val");
  try {
    int value = Integer.parseInt(val);
  }
  catch (NumberFormatException) {
    log.info("Failed to parse val = " + val);
  }
...
```

If a user submits the string "twenty-one" for val, the following entry is logged: INFO: Failed to parse val=twenty-one However, if an attacker submits the string "twenty-one%0a%0aINFO:+User+logged+out%3dbadguy", the following entry is logged: INFO: Failed to parse val=twenty-one INFO: User logged out=badguy Clearly, attackers can use this same mechanism to insert arbitrary log entries.

Context Notes	Log forging vulnerabilities occur when: 1. Data enters an application from an untrusted source. 2. The data is written to an application or system log file. Applications typically use log files to store a history of events or transactions for later review, statistics gathering, or debugging. Depending on the nature of the application, the task of reviewing log files may be performed manually on an as-needed basis or automated with a tool that automatically culls logs for important events or trending information. Interpretation of the log files may be hindered or misdirected if an attacker can supply data to the application that is subsequently logged verbatim. In the most

benign case, an attacker may be able to insert false entries into the log file by providing the application with input that includes appropriate characters. If the log file is processed automatically, the attacker can render the file unusable by corrupting the format of the file or injecting unexpected characters. A more subtle attack might involve skewing the log file statistics. Forged or otherwise, corrupted log files can be used to cover an attacker's tracks or even to implicate another party in the commission of a malicious act [22]. In the worst case, an attacker may inject code or other commands into the log file and take advantage of a vulnerability in the log processing utility [17].

References [17] G. Hoglund and G. McGraw. "Exploiting Software: How to Break Code". Addison-Wesley. February 2004.

[22] A. Muffet. "The night the log was forged". <http://doc.novsu.ac.ru/oreilly/tcpip/puis/ch10_05.htm>.

Node Relationships ChildOf - Output Validation (116)

Source Taxonomies 7 Pernicious Kingdoms - Log Forging

Applicable Platforms All

Path 1:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      66. String category = request.getParameter("category");
      ...
      95. report.getReportData(urlReplace, category, "all", null, startMonth,
      startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      167. populateReport(logourl, categoryName, reportFormat, monthFormat,
      startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
      reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
      176. log.debug("CATEGORY " + categoryName);
```

[Table of Contents] [Description]

Path 2:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      66. String category = request.getParameter("category");
      ...
      95. report.getReportData(urlReplace, category, "all", null, startMonth,
      startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      167. populateReport(logourl, categoryName, reportFormat, monthFormat,
      startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
      reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
      155. m_categoryName = categoryName;
      ...
      200. populateDataStructures(cat, report, format, monthFormat, catCount);
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
```

```
85. public static Category getInstance(Class c) //threadcategory.java
      ...
      90. cat = Category.getInstance(prefix + "." + c.getName());
      ...
      95. return cat;
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
      ...
      238. log.debug("Inside populate data Structures" + catIndex);
```

[Table of Contents] [Description]

Path 3:

Query Name - **Log_Forgery**

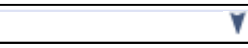
Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      66. String category = request.getParameter("category");
      ...
      95. report.getReportData(urlReplace, category, "all", null, startMonth,
      startDate, startYear);
```

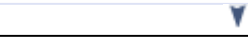
```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      167. populateReport(logourl, categoryName, reportFormat, monthFormat,
      startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
      reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

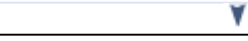
```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
      155. m_categoryName = categoryName;
      ...
      200. populateDataStructures(cat, report, format, monthFormat, catCount);
```



```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
```



```
85. public static Category getInstance(Class c) //threadcategory.java
      ...
      90. cat = Category.getInstance(prefix + "." + c.getName());
      ...
      95. return cat;
```



```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
      ...
      241. log.debug("Inside populate data Structures");
```

Path 4:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      66. String category = request.getParameter("category");
      ...
      95. report.getReportData(urlReplace, category, "all", null, startMonth,
      startDate, startYear);
```



```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      167. populateReport(logourl, categoryName, reportFormat, monthFormat,
      startMonth, startDate, startYear );
```



```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
      reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```



```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
      ...
      155. m_categoryName = categoryName;
      ...
      200. populateDataStructures(cat, report, format, monthFormat, catCount);
```



```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
```



```
85. public static Category getInstance(Class c) //threadcategory.java
      ...
      90. cat = Category.getInstance(prefix + "." + c.getName());
      ...
      95. return cat;
```



```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
      ());
```

```
...
250. log.debug("adding service" + service);
```

[Table of Contents] [Description]

Path 5:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
    66. String category = request.getParameter("category");
    ...
    95. report.getReportData(urlReplace, category, "all", null, startMonth,
    startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    167. populateReport(logourl, categoryName, reportFormat, monthFormat,
    startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
    reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
    155. m_categoryName = categoryName;
    ...
    209. populateDataStructures(cat, report, format, monthFormat, 1);
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
```

```
85. public static Category getInstance(Class c) //threadcategory.java
    ...
    90. cat = Category.getInstance(prefix + "." + c.getName());
    ...
    95. return cat;
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
```

```
236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
());
...
256. log.debug("Nodes " + m_nodes);
```

[Table of Contents] [Description]

Path 6:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
    66. String category = request.getParameter("category");
    ...
    95. report.getReportData(urlReplace, category, "all", null, startMonth,
    startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    167. populateReport(logourl, categoryName, reportFormat, monthFormat,
    startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
    reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
    155. m_categoryName = categoryName;
    ...
    209. populateDataStructures(cat, report, format, monthFormat, 1);
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
```

```
85. public static Category getInstance(Class c) //threadcategory.java
    ...
    90. cat = Category.getInstance(prefix + "." + c.getName());
    ...
    95. return cat;
```

```
235. private void populateDataStructures
```

```
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
    ...
    263. log.debug("Removing node: " + node);
```

[Table of Contents] [Description]

Path 7:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
    66. String category = request.getParameter("category");
    ...
    95. report.getReportData(urlReplace, category, "all", null, startMonth,
    startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    167. populateReport(logourl, categoryName, reportFormat, monthFormat,
    startMonth, startDate, startYear );
```

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
    197. AvailabilityData availData = new AvailabilityData(categoryName, m_report,
    reportFormat, monthFormat, calendar, startMonth, startDate, startYear);
```

```
147. public AvailabilityData(String categoryName, Report report, String format,
String monthFormat, Calendar calendar, String startMonth, String startDate, String
startYear) throws IOException, MarshalException, ValidationException,
Exception //availabilitydata.java
    ...
    155. m_categoryName = categoryName;
    ...
    200. populateDataStructures(cat, report, format, monthFormat, catCount);
```

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    236. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
    ...
    253. populateNodesFromDB(cat, monitoredServices);
```

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
    ...
```

```
533. initialiseConnection();
```

▼

```
425. public void initialiseConnection() throws IOException, MarshalException,
ValidationException, ClassNotFoundException, SQLException //availabilitydata.java
    426. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
```

▼

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
    ...
    533. initialiseConnection();
    ...
    654. closeConnection();
```

▼

```
459. public void closeConnection() //availabilitydata.java
    460. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
```

▼

```
524. private void populateNodesFromDB(org.opennms.netmgt.config.categories.Category
cat, List monitoredServices) throws SQLException, FilterParseException,
Exception //availabilitydata.java
    ...
    654. closeConnection();
```

▼

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
    ...
    253. populateNodesFromDB(cat, monitoredServices);
    ...
    271. topOffenders = getPercentNode();
```

▼

```
476. public TreeMap getPercentNode() //availabilitydata.java
    477. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
```

▼

```
85. public static Category getInstance(Class c) //threadcategory.java
    ...
    90. cat = Category.getInstance(prefix + "." + c.getName());
    ...
    95. return cat;
```

▼

```
476. public TreeMap getPercentNode() //availabilitydata.java
    477. org.apache.log4j.Category log = ThreadCategory.getInstance(this.getClass
    ());
    ...
    486. log.debug("getPercentNode: Start time " + new java.util.Date(startTime));
```

[Table of Contents] [Description]

Path 8:

Query Name - **Log_Forgery**

Severity - **Warning**

```
65. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rawavailabilityservlet.java
      66. String category = request.getParameter("category");
      ...
      95. report.getReportData(urlReplace, category, "all", null, startMonth,
      startDate, startYear);
```

```
158. public void getReportData(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
      ...
      162. log.debug("Category name " + categoryName);
```

[Table of Contents] [Description]

Path 9:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      ...
      136. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
      ...
      144. String response = lineRdr.readLine();
      ...
      148. log.debug("isServer: matching response=" + response);
```

[Table of Contents] [Description]

Path 10:

Query Name - **Log_Forgery**

Severity - **Warning**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      ...
      136. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
      ...
      144. String response = lineRdr.readLine();
      ...
      169. log.debug("isServer: NON-matching response=" + response);
```

[Table of Contents] [Description]

Path 11:

Query Name - **Log_Forgery**

Severity - **Warning**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
```

```
RE regex, StringBuffer bannerResult) //tcpplugin.java
    ...
    137. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
    (socket.getInputStream()));
    ...
    145. String response = lineRdr.readLine();
    ...
    148. log.debug("isServer: matching response=" + response);
```

[Table of Contents] [Description]

Path 12:

Query Name - **Log_Forgery**

Severity - **Warning**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
    ...
    137. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
    (socket.getInputStream()));
    ...
    145. String response = lineRdr.readLine();
    ...
    157. log.debug("isServer: NON-matching response=" + response);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //interface.jsp
    ...
    93. String ipAddr = request.getParameter( "intf" );
    ...
    264. List inPkgs = pollerCfgFactory.getAllPackageMatches(ipAddr);
```

```
634. public synchronized List getAllPackageMatches(String
ipaddr) //pollerconfigfactory.java
    ...
    642. boolean inPkg = interfaceInPackage(ipaddr, pkg);
```

```
472. public synchronized boolean interfaceInPackage(String iface,
org.opennms.netmgt.config.poller.Package pkg) //pollerconfigfactory.java
    ...
    480. filterPassed = ipList.contains(iface);
    ...
    484. log.debug("interfaceInPackage: Interface " + iface + " passed filter for
    package " + pkg.getName() + "?: " + filterPassed);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Log_Forgery**

Severity - **Warning**

```
245. public void run() //client.java
      ...
      253. input = new ObjectInputStream(m_client.getInputStream());
      ...
      271. Message msg = (Message) input.readObject();
      ...
      278. log.debug("Got request... adress = " + msg.getAddress());
```

[Table of Contents] [Description]

Path 15:

Query Name - **Log_Forgery**

Severity - **Warning**

```
233. private Poller(long timeout) throws IOException //poller.java
      ...
      245. m_ins = new ObjectInputStream(m_connection.getInputStream());
```

```
332. static long isServer(InetAddress host, long timeout, int retries) throws
IOException //poller.java
      ...
      401. Poller p = new Poller(timeout);
      ...
      421. resp = (Message) p.m_ins.readObject();
      ...
      431. log.debug("isServer: got a DHCP response from host " + resp.getAddress
      ().getHostAddress() + " with Xid: " + resp.getMessage().getXid());
```

[Table of Contents] [Description]

Path 16:

Query Name - **Log_Forgery**

Severity - **Warning**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
      ...
      201. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
      ...
      221. ENDMULTILINE = new RE(multiLineRC);
      ...
      282. response = lineRdr.readLine();
      ...
      316. log.debug("FtpMonitor.poll: tokenizing respone to check for return code:
      " + response);
```

[Table of Contents] [Description]

Path 17:

Query Name - **Log_Forgery**

Severity - **Warning**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
        ...
        204. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
        205. String line = lineRdr.readLine();
        ...
        211. log.debug("poll: response= " + line);
```

[Table of Contents] [Description]

Path 18:

Query Name - **Log_Forgery**

Severity - **Warning**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
        ...
        217. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (sslSocket.getInputStream()));
        218. String line = lineRdr.readLine();
        ...
        224. log.debug("HttpPlugin.poll: Response = " + line);
```

[Table of Contents] [Description]

Path 19:

Query Name - **Log_Forgery**

Severity - **Warning**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
        ...
        193. BufferedReader rdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
        ...
        199. String banner = rdr.readLine();
        ...
        203. log.debug("ImapMonitor.Poll(): banner: " + banner);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Log_Forgery**

Severity - **Warning**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
        ...
        201. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
        ...
        221. ENDMULTILINE = new RE(multiLineRC);
        ...
        400. this.updateRRD(rrdPath, ipv4Addr, dsName, responseTime, pkg);
```

```
246. public void updateRRD(String repository, InetAddress addr, String dsName, long
value, org.opennms.netmgt.config.poller.Package pkg) //ipv4latencymonitor.java
     247. Category log = ThreadCategory.getInstance(this.getClass());
```

```
85. public static Category getInstance(Class c) //threadcategory.java
     ...
     90. cat = Category.getInstance(prefix + "." + c.getName());
     ...
     95. return cat;
```

```
246. public void updateRRD(String repository, InetAddress addr, String dsName, long
value, org.opennms.netmgt.config.poller.Package pkg) //ipv4latencymonitor.java
     247. Category log = ThreadCategory.getInstance(this.getClass());
     ...
     261. log.error(msg);
```

[Table of Contents] [Description]

Path 21:

Query Name - **Log_Forgery**

Severity - **Warning**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
     ...
     208. BufferedReader rdr = new BufferedReader(new InputStreamReader
     (socket.getInputStream()));
     ...
     214. String banner = rdr.readLine();
     ...
     243. log.debug("poll: banner = " + banner);
```

[Table of Contents] [Description]

Path 22:

Query Name - **Log_Forgery**

Severity - **Warning**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
     ...
     179. BufferedReader rdr = new BufferedReader(new InputStreamReader
     (socket.getInputStream()));
     ...
     185. String response = rdr.readLine();
     ...
     191. log.debug("poll: banner = " + response);
```

[Table of Contents] [Description]

Path 23:

Query Name - **Log_Forgery**

Severity - **Warning**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
      ...
      194. BufferedReader rdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
      ...
      200. String response = rdr.readLine();
      ...
      206. log.debug("poll: banner = " + response);
```

[Table of Contents] [Description]

Path 24:

Query Name - **Log_Forgery**

Severity - **Warning**

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
      ...
      271. String value = request.getParameter(externalValues[i]);
      ...
      276. translationMap.put(RE.simplePatternToFullRegularExpression("{" +
      externalValues[i] + "}"), value);
      ...
      289. translationMap.put(
      ...
      299. Iterator iter = translationMap.keySet().iterator();
      ...
      302. String s1 = (String) iter.next();
      303. String s2 = (String) translationMap.get(s1);
      ...
      307. command = re.subst(command, s2);
      ...
      313. return command;
```

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
      ...
      182. String command = this.createPrefabCommand(request, report, rrds,
      propertiesFile, start, end);
      ...
      193. tempIn = RrdUtils.createGraph(command, workDir);
```

```
295. public static InputStream createGraph(String command, File workDir) throws
IOException, RrdException //rrdutils.java
      296. return getStrategy().createGraph(command, workDir);
```

```
319. public InputStream createGraph(String command, File workDir) throws
IOException, org.opennms.netmgt.rrd.RrdException //jrobinrrdstrategy.java
      ...
      323. String[] commandArray = tokenize(command, " \t", false);
```

```
238. private String[] tokenize(String line, String delims, boolean
processQuoted) //jrobinrrdstrategy.java
      ...
```

```
248. log.debug("tokenize: line=" + line + " delims=" + delims);
```

[Table of Contents] [Description]

Path 25:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //interface.jsp
...
93. String ipAddr = request.getParameter( "intf" );
...
221. <% String ifLabel = IfLabel.getIfLabel(nodeId, ipAddr); %>
```

```
209. public static String getIfLabel(int nodeId, String ipAddr) throws
SQLException //iflabel.java
...
220. stmt.setString(2, ipAddr);
...
222. ResultSet rs = stmt.executeQuery();
...
225. String name = rs.getString("snmpifname");
...
230. label = getIfLabel(name, descr, physAddr);
```

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
...
304. label = AlphaNumeric.parseAndReplace(name, '_');
```

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java
...
67. byte[] bytes = str.getBytes();
...
76. String temp = new String(bytes);
...
83. log.debug("parseAndReplace: original='" + str + "'" + " new='" + temp +
"'");
```

[Table of Contents] [Description]

Path 26:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //custom_graph2.jsp
...
55. String nodeId = request.getParameter("node");
...
64. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
424. public String[] getQueryableInterfacesForNode(String
```

```
nodeId) //performancemodel.java
    ...
    430. File nodeDir = new File(this.rrdDirectory, nodeId);
    ...
    436. File[] intfDirs = nodeDir.listFiles
    (RrdFileConstants.INTERFACE_DIRECTORY_FILTER);
    ...
    442. intfs[i] = intfDirs[i].getName();
    ...
    446. return intfs;
```

```
1. <!-- //custom_graph2.jsp
    ...
    64. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
    ...
    66. Arrays.sort(intfs);
    ...
    142. <%=this.model.getHumanReadableNameForIfLabel(Integer.parseInt(nodeId),
    intfs[i])%></option>
```

```
288. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //performancemodel.java
    ...
    297. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
    ...
    169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
    ...
    92. String desc = ifLabel;
    ...
    104. log.debug("getInterfaceInfoFromIfLabel: desc=" + desc + " mac=" + mac);
```

[Table of Contents] [Description]

Path 27:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //interface.jsp
    ...
    93. String ipAddr = request.getParameter( "intf" );
    ...
    227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
    ifindexString); %>
```

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
    ...
    261. stmt.setString(2, ipAddr);
```

```
262. stmt.setString(3, ifIndex);
...
264. ResultSet rs = stmt.executeQuery();
...
269. String physAddr = rs.getString("snmpphysaddr");
...
272. label = getIfLabel(name, descr, physAddr);
```

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
...
316. physAddr = AlphaNumeric.parseAndTrim(physAddr);
```

```
103. public static String parseAndTrim(String str) //alphanumeric.java
...
107. byte[] bytes = str.getBytes();
...
115. String temp = new String(bytes);
116. temp = temp.trim();
...
118. return temp;
```

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
...
316. physAddr = AlphaNumeric.parseAndTrim(physAddr);
...
321. log.debug("initialize: physical address len is NOT 12, physAddr=" +
physAddr);
```

[Table of Contents] [Description]

Path 28:

Query Name - **Log_Forgery**

Severity - **Warning**

```
74. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //nodelabelchangeservlet.java
...
77. String userLabel = request.getParameter("userlabel");
...
97. newLabel = new NodeLabel(userLabel, NodeLabel.SOURCE_USERDEFINED);
```

```
184. public NodeLabel(String nodeLabel, char nodeLabelSource) //nodelabel.java
185. m_nodeLabel = nodeLabel;
```

```
74. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //nodelabelchangeservlet.java
...
97. newLabel = new NodeLabel(userLabel, NodeLabel.SOURCE_USERDEFINED);
...
102. NodeLabel.assignLabel(nodeId, newLabel);
```

```
314. public static void assignLabel(int nodeID, NodeLabel nodeLabel) throws
SQLException //nodelabel.java
      ...
      318. assignLabel(nodeID, nodeLabel, dbConnection);
```

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
      ...
      354. log.debug("NodeLabel.assignLabel: Node label: " + nodeLabel.getLabel() +
      " source: " + nodeLabel.getSource());
```

```
194. public String getLabel() //nodelabel.java
      195. return m_nodeLabel;
```

```
338. public static void assignLabel(int nodeID, NodeLabel nodeLabel, Connection
dbConnection) throws SQLException //nodelabel.java
      ...
      354. log.debug("NodeLabel.assignLabel: Node label: " + nodeLabel.getLabel() +
      " source: " + nodeLabel.getSource());
```

[Table of Contents] [Description]

Path 29:

Query Name - **Log_Forgery**

Severity - **Warning**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
      ...
      141. String category = request.getParameter("category");
      ...
      179. reportMailer.initialise(filename, username, scriptGenerateReport,
      scriptMailReport, category, "HTML", monthFormat, startMonth, startDate,
      startYear);
```

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
      ...
      115. log.debug("parms " + parms);
```

[Table of Contents] [Description]

Path 30:

Query Name - **Log_Forgery**

Severity - **Warning**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
      ...
      140. String monthFormat = request.getParameter("monthformat");
      ...
```

```
179. reportMailer.initialise(filename, username, scriptGenerateReport,
scriptMailReport, category, "HTML", monthFormat, startMonth, startDate,
startYear);
```

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
    ...
117. log.debug("monthFmt " + monthFmt);
```

[Table of Contents] [Description]

Path 31:

Query Name - **Log_Forgery**

Severity - **Warning**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
    ...
143. String startMonth = request.getParameter("startMonth");
...
179. reportMailer.initialise(filename, username, scriptGenerateReport,
scriptMailReport, category, "HTML", monthFormat, startMonth, startDate,
startYear);
```

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
    ...
118. log.debug("startMonth " + startMonth);
```

[Table of Contents] [Description]

Path 32:

Query Name - **Log_Forgery**

Severity - **Warning**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
    ...
144. String startDate = request.getParameter("startDate");
...
179. reportMailer.initialise(filename, username, scriptGenerateReport,
scriptMailReport, category, "HTML", monthFormat, startMonth, startDate,
startYear);
```

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
    ...
119. log.debug("startDate " + startDate);
```

[Table of Contents] [Description]

Path 33:

Query Name - **Log_Forgery**

Severity - **Warning**

```
137. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //availabilityservlet.java
    ...
    145. String startYear = request.getParameter("startYear");
    ...
    179. reportMailer.initialise(filename, username, scriptGenerateReport,
    scriptMailReport, category, "HTML", monthFormat, startMonth, startDate,
    startYear);
```

```
101. public void initialise(String fileName, String userName, String generateReport,
String mailReport, String parms, String fmt, String monthFmt, String startMonth,
String startDate, String startYear) throws ServletException //reportmailer.java
    ...
    120. log.debug("startYear " + startYear);
```

[Table of Contents] [Description]

Path 34:

Query Name - **Log_Forgery**

Severity - **Warning**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
    ...
    74. String lastIPAddress = request.getParameter("lastIPAddress");
    ...
    81. log.debug("doPost: firstIPAddress=" + firstIPAddress + ", "
```

[Table of Contents] [Description]

Path 35:

Query Name - **Log_Forgery**

Severity - **Warning**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
    82. String sendto = request.getParameter("sendto");
    ...
    87. this.log.debug("To: " + sendto + ", Subject: " + subject + ", message: " +
    msg + ", username: " + username);
```

[Table of Contents] [Description]

Path 36:

Query Name - **Log_Forgery**

Severity - **Warning**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
```

```
...
74. String lastIPAddress = request.getParameter("lastIPAddress");
...
82. + "lastIPAddress=" + lastIPAddress + ", and "
```

[Table of Contents] [Description]

Path 37:

Query Name - **Log_Forgery**

Severity - **Warning**

```
72. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpconfigservlet.java
...
75. String communityString = request.getParameter("communityString");
...
83. + "communityString=" + communityString);
```

[Table of Contents] [Description]

Path 38:

Query Name - **Log_Forgery**

Severity - **Warning**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
...
83. String subject = request.getParameter("subject");
...
87. this.log.debug("To: " + sendto + ", Subject: " + subject + ", message: " +
msg + ", username: " + username);
```

[Table of Contents] [Description]

Path 39:

Query Name - **Log_Forgery**

Severity - **Warning**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
...
84. String msg = request.getParameter("msg");
...
87. this.log.debug("To: " + sendto + ", Subject: " + subject + ", message: " +
msg + ", username: " + username);
```

[Table of Contents] [Description]

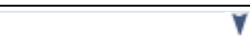
Path 40:

Query Name - **Log_Forgery**

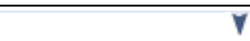
Severity - **Warning**

```
1. <!-- //interface.jsp
```

```
...
94. String ifindexString = request.getParameter( "ifindex" );
...
227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
ifindexString); %>
```



```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java

    ...
    262. stmt.setString(3, ifIndex);
    ...
    264. ResultSet rs = stmt.executeQuery();
    ...
    267. String name = rs.getString("snmpifname");
    ...
    272. label = getIfLabel(name, descr, physAddr);
```



```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java

    ...
    304. label = AlphaNumeric.parseAndReplace(name, '_');
```

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java

    ...
    67. byte[] bytes = str.getBytes();
    ...
    76. String temp = new String(bytes);
    ...
    83. log.debug("parseAndReplace: original='" + str + "'" + " new='" + temp +
    "'");
```

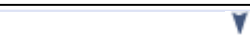
[Table of Contents] [Description]

Path 41:

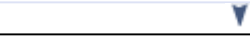
Query Name - **Log_Forgery**

Severity - **Warning**

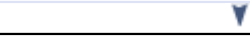
```
1. <!-- //custom_graph3.jsp

    ...
    59. int nodeId = Integer.parseInt(nodeIdString);
    ...
    171. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```



```
288. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //performancemodel.java

    ...
    297. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```



```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java

    ...
    169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```



```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
        ...
        92. String desc = ifLabel;
        ...
        104. log.debug("getInterfaceInfoFromIfLabel: desc=" + desc + " mac=" + mac);
```

[Table of Contents] [Description]

Path 42:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //adhoc2.jsp
        ...
        67. String intf = request.getParameter("intf");
        ...
        133. Node: <%=nodeLabel%>    Interface: <%
        =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
        ...
        304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
        ...
        169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
        ...
        92. String desc = ifLabel;
        ...
        104. log.debug("getInterfaceInfoFromIfLabel: desc=" + desc + " mac=" + mac);
```

[Table of Contents] [Description]

Path 43:

Query Name - **Log_Forgery**

Severity - **Warning**

```
1. <!-- //results.jsp
        ...
        77. String intf = request.getParameter( "intf" );
        ...
        212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
```

```
...
304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```



```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
...
169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
...
92. String desc = ifLabel;
...
104. log.debug("getInterfaceInfoFromIfLabel: desc=" + desc + " mac=" + mac);
```

[Table of Contents] [Description]

Improper Resource Shutdown or Release

CWE ID	404
Description	The program can potentially fail to release or incorrectly release a system resource. A resource is not properly cleared and made available for re-use.
Functional Area	Non-specific
Potential Mitigations	It is good practice to be responsible for freeing all resources you allocate. Memory should be allocated/freed using matching functions such as malloc/free, new/delete, and new[]/delete[].
Demonstrative Examples	The following method never closes the file handle it opens. The Finalize() method for StreamReader eventually calls Close (), but there is no guarantee as to how long it will take before the Finalize() method is invoked. In fact, there is no guarantee that Finalize() will ever be invoked. In a busy environment, this can result in the VM using up all of its available file handles.

```
private void processFile(string fName) {
  StreamWriter sw = new
  StreamWriter(fName);
  string line;
  while ((line = sr.ReadLine()) != null) processLine(line);
}
```

If an exception occurs after establishing the database connection and before the same connection closes, the pool of database connections may become exhausted. If the number of available connections is exceeded, other users cannot access this resource, effectively denying access to the application. Using the following database connection pattern will ensure that all opened connections are closed. The

con.close() call should be the first executable statement in the finally block.

```
try {
  Connection con = DriverManager.getConnection
(some_connection_string)
}
catch ( Exception e ) {
  log( e )
}
finally {
  con.close()
}
```

Under normal conditions the following C# code executes a database query, processes the results returned by the database, and closes the allocated SqlConnection object. But if an exception occurs while executing the SQL or processing the results, the SqlConnection object is not closed. If this happens often enough, the database will run out of available cursors and not be able to execute any more SQL queries.

C# Example:

```
...
SqlConnection conn = new SqlConnection(connString);
SqlCommand cmd = new SqlCommand(queryString);
cmd.Connection = conn; conn.Open();
SqlDataReader rdr = cmd.ExecuteReader();
HarvestResults(rdr);
conn.Connection.Close();
...
```

The following C function does not close the file handle it opens if an error occurs. If the process is long-lived, the process can run out of file handles.

C Example:

```
int decodeFile(char* fName) {
  char buf[BUF_SZ];
  FILE* f = fopen(fName, "r");
  if (!f) {
    printf("cannot open %s\n", fName);
    return DECODE_FAIL;
  }
  else {
    while (fgets(buf, BUF_SZ, f)) {
```

```
      if (!checkChecksum(buf)) {
        return DECODE_FAIL;
      }
      else {
        decodeBlock(buf);
      }
    }
  }
  fclose(f);
  return DECODE_SUCCESS;
}
```

In this example, the program fails to use matching functions such as malloc/free, new/delete, and new[]/delete[] to allocate/deallocate the resource.

C++ Example:

```
class A{
  void foo();
};
void A::foo(){
  int *ptr;
  ptr = (int*)malloc(sizeof(int));
  delete ptr;
}
```

In this example, the program calls the delete[] function on non-heap memory.

C++ Example:

```
class A{
  void foo(bool);
};
void A::foo(bool heap) {
  int localArray[2] = {11,22};
  int *p = localArray;
  if (heap){
    p = new int[2];
  }
  delete[] p;
}
```

Observed Examples

CVE-1999-1127 - Does not shut down named pipe connections if malformed data is sent.

CVE-2001-0830 - Sockets not properly closed when attacker repeatedly connects and disconnects from server.

Context Notes	CVE-2002-1372 - Return values of file/socket operations not checked, allowing resultant consumption of file descriptors. It is important to be consistent with how and where you free memory in a function. If you allocate memory that you intend to free upon completion of the function, you must be sure to free the memory at all exit points for that function including error conditions. Before freeing a pointer, the programmer should make sure that the pointer was previously allocated and that the memory belongs to the programmer. Freeing an unallocated pointer will cause undefined behavior in the program. Can be resultant from improper error handling or insufficient resource tracking. Overlaps memory leaks, asymmetric resource consumption, malformed input errors. Most unreleased resource issues result in general software reliability problems, but if an attacker can intentionally trigger a resource leak, the attacker might be able to launch a denial of service attack by depleting the resource pool. Resource leaks have at least two common causes: - Error conditions and other exceptional circumstances. - Confusion over which part of the program is responsible for releasing the resource.
Node Relationships	ChildOf - Resource Management Errors (399) Peer - Memory Leak (401) Peer - Asymmetric Resource Consumption (Amplification) (405)
Source Taxonomies	PLOVER - Improper resource shutdown or release 7 Pernicious Kingdoms - Unreleased Resource
Applicable Platforms	All

Path 1:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
83. public void run() //processexec.java
      |
      |...
      |85. BufferedReader in = new BufferedReader(new InputStreamReader
      |(m_inputStream));
```

[Table of Contents] [Description]

Path 2:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
245. public void generatePDF(String pdfFileName, OutputStream out, String format)
throws Exception //availabilityreport.java
      |
```

```
...
254. FileReader fileReader = new FileReader(file);
```

[Table of Contents] [Description]

Path 3:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
104. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //citrixplugin.java
...
108. BufferedReader reader = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 4:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
130. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //ftpplugin.java
...
138. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 5:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
196. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //httpplugin.java
...
205. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 6:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
124. protected boolean checkProtocol(Socket socket, ConnectionConfig config) throws
IOException //imapplugin.java
...
129. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 7:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
129. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //msexchangeplugin.java
      ...
      143. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 8:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
118. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //pop3plugin.java
      ...
      133. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 9:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
149. private boolean isServer(InetAddress host, int port, int retries, int
timeout) //smtpplugin.java
      ...
      165. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 10:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
114. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //sshplugin.java
      ...
      136. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 11:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
115. private boolean isServer(InetAddress host, int port, int retries, int timeout,
RE regex, StringBuffer bannerResult) //tcpplugin.java
        ...
        137. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 12:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
174. public void run() //tcpstreamhandler.java
        ...
        281. InputStreamReader in = new InputStreamReader(new BufferedInputStream
        (pipeIn));
```

[Table of Contents] [Description]

Path 13:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
113. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //citrixmonitor.java
        ...
        171. BufferedReader reader = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 14:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
149. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //ftpmonitor.java
        ...
        201. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 15:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
120. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpmonitor.java
        ...
        204. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
        (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 16:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
125. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //httpsmonitor.java
      ...
      217. BufferedReader lineRdr = new BufferedReader(new InputStreamReader
      (sslSocket.getInputStream()));
```

[Table of Contents] [Description]

Path 17:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
143. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //imapmonitor.java
      ...
      193. BufferedReader rdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 18:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
121. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //pop3monitor.java
      ...
      167. BufferedReader rdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 19:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
159. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //smtpmonitor.java
      ...
      208. BufferedReader rdr = new BufferedReader(new InputStreamReader
      (socket.getInputStream()));
```

[Table of Contents] [Description]

Path 20:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
111. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //sshmonitor.java
    ...
179. BufferedReader rdr = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 21:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
117. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //tcpmonitor.java
    ...
194. BufferedReader rdr = new BufferedReader(new InputStreamReader
(socket.getInputStream()));
```

[Table of Contents] [Description]

Path 22:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
379. public InputStream createGraph(String command, File workDir) throws
IOException, RrdException //jnirrdstrategy.java
    ...
393. BufferedReader err = new BufferedReader(new InputStreamReader
(process.getErrorStream()));
```

[Table of Contents] [Description]

Path 23:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
173. public static void main(final String[] args) throws
Exception //rrdstresser.java
    ...
180. BufferedReader rdr = new BufferedReader(new InputStreamReader
(RrdStresser.class.getResourceAsStream(UPDATE_FILE)));
```

[Table of Contents] [Description]

Path 24:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
166. public void run() //streamgobbler.java
    ...
```

```
171. InputStreamReader isr = new InputStreamReader(in);
```

[Table of Contents] [Description]

Path 25:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
166. public void run() //streamgobbler.java
     ...
172. BufferedReader br = new BufferedReader(isr);
```

[Table of Contents] [Description]

Path 26:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
263. protected void mailFileToUser(String mailScript, String filename, String
emailAddr) throws IOException //reportmailer.java
     ...
272. BufferedReader err = new BufferedReader(new InputStreamReader
(process.getErrorStream()));
```

[Table of Contents] [Description]

Path 27:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
236. FileWriter poller_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.POLLER_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 28:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
437. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
".port");
```

[Table of Contents] [Description]

Path 29:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
238. FileWriter poller_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.POLLER_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 30:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
239. FileWriter capsd_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.CAPSD_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 31:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
238. FileWriter poller_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.POLLER_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 32:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
239. FileWriter capsd_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.CAPSD_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 33:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException //pollerconfigservlet.java
    ...
243. FileWriter poller_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.POLLER_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 34:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
244. FileWriter capsd_fileWriter = new FileWriter(ConfigFileConstants.getFile
(ConfigFileConstants.CAPSD_CONFIG_FILE_NAME));
```

[Table of Contents] [Description]

Path 35:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //mailerservlet.java
    ...
114. BufferedReader err = new BufferedReader(new InputStreamReader
(process.getErrorStream()));
```

[Table of Contents] [Description]

Path 36:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
133. protected void loadDTD(String dtdFileName) throws FileNotFoundException,
IOException //xmlwriter.java
    ...
137. BufferedReader originalBuffer = new BufferedReader(new FileReader
(dtdFileName));
```

[Table of Contents] [Description]

Path 37:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
95. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
    ...
111. BufferedReader err = new BufferedReader(new InputStreamReader
(process.getErrorStream()));
```

[Table of Contents] [Description]

Path 38:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
95. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
    ...
    127. BufferedReader in = new BufferedReader(new InputStreamReader
    (process.getInputStream()));
```

[Table of Contents] [Description]

Path 39:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
99. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
    ...
    115. BufferedReader err = new BufferedReader(new InputStreamReader
    (process.getErrorStream()));
```

[Table of Contents] [Description]

Path 40:

Query Name - **DoS_by_Unreleased_Resources**

Severity - **Warning**

```
99. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrddumpservlet.java
    ...
    131. BufferedReader in = new BufferedReader(new InputStreamReader
    (process.getInputStream()));
```

[Table of Contents] [Description]

Hard-Coded Password

CWE ID	259
Description	Hard coded passwords may compromise system security in a way that cannot be easily remedied. It is never a good idea to hardcode a password. Not only does hardcoding a password allow all of the project's developers to view the password, it also makes fixing the problem extremely difficult. Once the code is in production, the password cannot be changed without patching the software. If the account protected by the password is compromised, the owners of the system will be forced to choose between security and availability.
Likelihood of Exploit	Very High

Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Common Consequences	Authentication: If hard-coded passwords are used, it is almost certain that malicious users will gain access through the account in question.
Potential Mitigations	Design (for default accounts): Rather than hard code a default username and password for first time logins, utilize a "first login" mode which requires the user to enter a unique strong password. Design (for front-end to back-end connections): Three solutions are possible, although none are complete. The first suggestion involves the use of generated passwords which are changed automatically and must be entered at given time intervals by a system administrator. These passwords will be held in memory and only be valid for the time intervals. Next, the passwords used should be limited at the back end to only performing actions valid to for the front end, as opposed to having full access. Finally, the messages sent should be tagged and checksummed with time sensitive values so as to prevent replay style attacks.
Demonstrative Examples	The following code uses a hardcoded password to connect to a database: ... DriverManager.getConnection(url, "scott", "tiger"); ... This code will run successfully, but anyone who has access to it will have access to the password. Once the program has shipped, there is no going back from the database user "scott" with a password of "tiger" unless the program is patched. A devious employee with access to this information can use it to break into the system. Even worse, if attackers have access to the bytecode for application, they can use the javap -c command to access the disassembled code, which will contain the values of the passwords used. The result of this operation might look something like the following for the example above: javap -c ConnMngr.class 22: ldc #36; //String jdbc:mysql://ixne.com/rxsql 24: ldc #38; //String scott 26: ldc #17; //String tiger

---

C/C++ Example:

```
int VerifyAdmin(char *password) {
if (strcmp(password, "Mew!")) {
  printf("Incorrect Password!\n");
```

```
  return(0)
}
printf("Entering Diagnostic Mode…\n"); return(1);
}
```

Java Example:

```
int VerifyAdmin(String password) {
  if (passwd.Eqauls("Mew!")) {
    return(0)
  } //Diagnostic Mode return(1);
}
```

	Every instance of this program can be placed into diagnostic mode with the same password. Even worse is the fact that if this program is distributed as a binary-only distribution, it is very difficult to change that password or disable this "functionality."
Context Notes	The use of a hard-coded password has many negative implications -- the most significant of these being a failure of authentication measures under certain circumstances. On many systems, a default administration account exists which is set to a simple default password which is hard-coded into the program or device. This hard-coded password is the same for each device or system of this type and often is not changed or disabled by end users. If a malicious user comes across a device of this kind, it is a simple matter of looking up the default password (which is freely available and public on the internet) and logging in with complete access. In systems which authenticate with a back-end service, hard-coded passwords within closed source or drop-in solution systems require that the back-end service use a password which can be easily discovered. Client-side systems with hard-coded passwords propose even more of a threat, since the extraction of a password from a binary is exceedingly simple.
Node Relationships	ChildOf - Credentials Management (255) Peer - Use of Hard-coded Cryptographic Key (321) Peer - Storing Passwords in a Recoverable Format (257) ChildOf - Weaknesses Examined by SAMATE (630)
Source Taxonomies	7 Pernicious Kingdoms - Password Management: Hard-Coded Password CLASP - Use of hard-coded password
Applicable Platforms	All

Path 1:

Query Name - **Hardcoded_Password**

Severity - **Warning**

```
58. public class EventConstants //eventconstants.java
      ...
646. public final static String PARM_PASSWD = "passwd";
```

[Table of Contents] [Description]

Path 2:

Query Name - **Hardcoded_Password**

Severity - **Warning**

```
61. public final class RadiusAuthPlugin extends
AbstractPlugin //radiusauthplugin.java
      ...
102. public static final String DEFAULT_PASSWORD = "OpenNMS";
```

[Table of Contents] [Description]

Path 3:

Query Name - **Hardcoded_Password**

Severity - **Warning**

```
62. final public class RadiusAuthMonitor extends
IPv4LatencyMonitor //radiusauthmonitor.java
      ...
96. public static final String DEFAULT_PASSWORD = "OpenNMS";
```

[Table of Contents] [Description]

Path 4:

Query Name - **Hardcoded_Password**

Severity - **Warning**

```
74. public class NsclientManager //nsclientmanager.java
      ...
98. private String m_Password = "None";
```

[Table of Contents] [Description]

Path 5:

Query Name - **Hardcoded_Password**

Severity - **Warning**

```
87. class NessusScan implements Runnable //nessusscan.java
      ...
98. private static final String NTP_PASSWORD_PROMPT = "Password : ";
```

[Table of Contents] [Description]

Unsafe URL Redirection

CWE ID	601
Description	An http parameter may contain a URL value and could cause the web application to redirect the request to the specified URL. By modifying the URL value to a malicious site, an attacker may successfully launch a phishing scam and steal user credentials. The fact that the server name in the modified link is identical to the original site helps the attacker by giving his phishing attempts a more reliable appearance.
Observed Examples	- CVE-2005-4206 - Blackboard Learning and Community Portal System in Academic Suite 6.3.1.424, 6.2.3.23, and other versions before 6 allows remote attackers to redirect users to other URLs and conduct phishing attacks via a modified url parameter to frameset.jsp, which loads the URL into a frame and causes it to appear to be part of a valid page.
Context Notes	Phishing is a general term for deceptive attempts to coerce private information from users that will be used for identity theft.
Node Relationships	ChildOf - API Abuse (227)
Source Taxonomies	Anonymous Tool Vendor (under NDA) -

Path 1:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
214. String name1 = request.getParameter("name1");
     ...
225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

▼

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
     ...
282. response.sendRedirect("error.jsp?error=1&name=" + name);
```

[Table of Contents] [Description]

Path 2:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
     ...
215. String protoArray1 = request.getParameter("protArray1");
     ...
225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

▼

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
```

```
...
319. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
".port ");
```

[Table of Contents] [Description]

Path 3:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
215. String protoArray1 = request.getParameter("protArray1");
...
225. addCapsdInfo(name1, port1, user_id, protoArray1, response);
```

▼

```
255. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
357. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
".capsd-class ");
```

[Table of Contents] [Description]

Path 4:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
214. String name1 = request.getParameter("name1");
...
221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

▼

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java
...
371. response.sendRedirect("error.jsp?error=1&name=" + name);
```

[Table of Contents] [Description]

Path 5:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
```

```
215. String protoArray1 = request.getParameter("protArray1");
...
221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java

...
399. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
".monitor ");
```

[Table of Contents] [Description]

Path 6:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java

...
215. String protoArray1 = request.getParameter("protArray1");
...
221. addPollerInfo(check1, name1, port1, user_id, protoArray1, response);
```

```
362. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response) throws ServletException,
IOException //addpollerconfigservlet.java

...
437. response.sendRedirect("error.jsp?error=0&name=" + "service." + protocol +
".port");
```

[Table of Contents] [Description]

Path 7:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
76. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //destinationwizardservlet.java

...
248. String redirectPage = request.getParameter("nextPage");
249. redirectString.append(redirectPage);
...
308. response.sendRedirect(redirectString.toString());
```

[Table of Contents] [Description]

Path 8:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
77. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //notificationwizardservlet.java
```

```
...
180. String redirectPage = request.getParameter("nextPage");
...
198. redirectString.append(redirectPage).append(makeQueryString(params));
...
257. response.sendRedirect(redirectString.toString());
```

[Table of Contents] [Description]

Path 9:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
281. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=1&name=" + name);
```

[Table of Contents] [Description]

Path 10:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
316. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".port ");
```

[Table of Contents] [Description]

Path 11:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
```

```
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
...
353. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".capsd-class ");
```

[Table of Contents] [Description]

Path 12:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
367. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=1&name=" + name);
```

[Table of Contents] [Description]

Path 13:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
392. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".monitor ");
```

[Table of Contents] [Description]

Path 14:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
      ...
      89. StringBuffer buffer = request.getRequestURL();
      ...
      114. return buffer.toString();
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
      ...
      432. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
      error=0&name=" + "service." + protocol + ".port");
```

[Table of Contents] [Description]

Path 15:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
      ...
      257. String redirectPage = request.getParameter("redirect");
      ...
      261. response.sendRedirect(redirectPage);
```

[Table of Contents] [Description]

Path 16:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyassetservlet.java
      64. String nodeIdString = request.getParameter("node");
      ...
      75. int nodeId = Integer.parseInt(nodeIdString);
      ...
      90. response.sendRedirect("modify.jsp?node=" + nodeId);
```

[Table of Contents] [Description]

Path 17:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
65. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //noderescanservlet.java
      ...
      68. String returnUrl = request.getParameter("returnUrl");
      ...
```

```
92. response.sendRedirect(request.getContextPath() + "/" + returnUrl);
```

[Table of Contents] [Description]

Path 18:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
125. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventbyfilterservlet.java
     126. String redirectValue = request.getParameter("redirect");
     ...
     129. return (redirectValue);
```

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
     ...
     115. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 19:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
120. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventservlet.java
     121. String redirectValue = request.getParameter("redirect");
     ...
     124. return (redirectValue);
```

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
     ...
     110. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 20:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
     ...
     432. String[] values = request.getParameterValues(name);
     ...
     439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
     147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
        ...
        439. buffer.append(Util.encode(values[i]));
        ...
        467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
        397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
94. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //eventqueryservlet.java
        ...
        181. queryString = Util.makeQueryString(request, paramAdditions, IGNORE_LIST);
        ...
        186. response.sendRedirect(this.redirectUrl + "?" + queryString);
```

[Table of Contents] [Description]

Path 21:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
74. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //nodelabelchangeservlet.java
        75. String nodeIdString = request.getParameter("node");
        ...
        104. response.sendRedirect(request.getContextPath() + "/element/node.jsp?
        node=" + nodeIdString);
```

[Table of Contents] [Description]

Path 22:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
106. protected String getRedirectString(HttpServletRequest
request) //acknowledgenoticeservlet.java
        107. String redirectValue = request.getParameter("redirect");
        ...
        110. return (redirectValue);
```

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
        ...
        96. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 23:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    114. return buffer.toString();
```

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    126. String queryString = Util.makeQueryString(request);
```

[Table of Contents] [Description]

Path 24:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
    ...
    89. StringBuffer buffer = request.getRequestURL();
    ...
    114. return buffer.toString();
```

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    120. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
    queryString);
```

[Table of Contents] [Description]

Path 25:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    432. String[] values = request.getParameterValues(name);
    ...
    439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
    147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    439. buffer.append(Util.encode(values[i]));
    ...
```

```
467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
    397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
354. public static String makeQueryString(HttpServletRequest request) //util.java
    355. return (makeQueryString(request, new HashMap(), new String[0]));
```

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    126. String queryString = Util.makeQueryString(request);
    ...
    129. response.sendRedirect(this.chooseInterfaceUrl + "?" + queryString);
```

[Table of Contents] [Description]

Path 26:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    432. String[] values = request.getParameterValues(name);
    ...
    439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
    147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    439. buffer.append(Util.encode(values[i]));
    ...
    467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
    397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
365. public static String makeQueryString(HttpServletRequest request, Map
additions) //util.java
    366. return (makeQueryString(request, additions, new String[0]));
```

```
60. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
```

```
...
92. String queryString = Util.makeQueryString(request, additions);
...
94. response.sendRedirect("results.jsp?" + queryString);
```

[Table of Contents] [Description]

Path 27:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
83. public static String calculateUrlBase(HttpServletRequest request) //util.java
...
89. StringBuffer buffer = request.getRequestURL();
...
114. return buffer.toString();
```

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
125. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 28:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
...
432. String[] values = request.getParameterValues(name);
...
439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
...
439. buffer.append(Util.encode(values[i]));
...
467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
354. public static String makeQueryString(HttpServletRequest request) //util.java
```

```
355. return (makeQueryString(request, new HashMap(), new String[0]));
```

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    131. String queryString = Util.makeQueryString(request);
    ...
    134. response.sendRedirect(this.chooseInterfaceUrl + "?" + queryString);
```

[Table of Contents] [Description]

Path 29:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

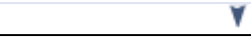
```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    432. String[] values = request.getParameterValues(name);
    ...
    439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
    147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    439. buffer.append(Util.encode(values[i]));
    ...
    467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
    397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
365. public static String makeQueryString(HttpServletRequest request, Map
additions) //util.java
    366. return (makeQueryString(request, additions, new String[0]));
```



```
64. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    95. String queryString = Util.makeQueryString(request, additions);
    ...
    97. response.sendRedirect("results.jsp?" + queryString);
```

[Table of Contents] [Description]

Path 30:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    218. String name1 = request.getParameter("name1");
    ...
    229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
    ...
    281. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=1&name=" + name);
```

[Table of Contents] [Description]

Path 31:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    219. String protoArray1 = request.getParameter("protArray1");
    ...
    229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
    ...
    316. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
    error=0&name=" + "service." + protocol + ".port ");
```

[Table of Contents] [Description]

Path 32:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
    219. String protoArray1 = request.getParameter("protArray1");
    ...
    229. addCapsdInfo(name1, port1, user_id, protoArray1, response, request);
```

```
256. public void addCapsdInfo(String name, String port, String user, String
protocol, HttpServletResponse response, HttpServletRequest request) throws
ServletException, IOException //addpollerconfigservlet.java
    ...
```

```
353. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".capsd-class ");
```

[Table of Contents] [Description]

Path 33:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
218. String name1 = request.getParameter("name1");
...
225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
request);
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
367. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=1&name=" + name);
```

[Table of Contents] [Description]

Path 34:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
...
219. String protoArray1 = request.getParameter("protArray1");
...
225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
request);
```

```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
392. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".monitor ");
```

[Table of Contents] [Description]

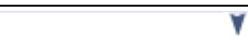
Path 35:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
```

```
...
219. String protoArray1 = request.getParameter("protArray1");
...
225. addPollerInfo(check1, name1, port1, user_id, protoArray1, response,
request);
```



```
358. public void addPollerInfo(String bPolled, String name, String port, String
user, String protocol, HttpServletResponse response, HttpServletRequest request)
throws ServletException, IOException //addpollerconfigservlet.java
...
432. response.sendRedirect(Util.calculateUrlBase(request) + "/admin/error.jsp?
error=0&name=" + "service." + protocol + ".port");
```

[Table of Contents] [Description]

Path 36:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
125. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventbyfilterservlet.java
...
133. String redirectParms = request.getParameter("redirectParms");
...
138. buffer.append(redirectParms);
139. redirectValue = buffer.toString();
...
142. return (redirectValue);
```

```
81. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventbyfilterservlet.java
...
115. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 37:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
120. protected String getRedirectString(HttpServletRequest
request) //acknowledgeeventservlet.java
...
128. String redirectParms = request.getParameter("redirectParms");
...
133. buffer.append(redirectParms);
134. redirectValue = buffer.toString();
...
137. return (redirectValue);
```

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
...
110. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 38:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
106. protected String getRedirectString(HttpServletRequest
request) //acknowledgenoticeservlet.java
      ...
      114. String redirectParms = request.getParameter("redirectParms");
      ...
      119. buffer.append(redirectParms);
      120. redirectValue = buffer.toString();
      ...
      123. return (redirectValue);
```

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
      ...
      96. response.sendRedirect(this.getRedirectString(request));
```

[Table of Contents] [Description]

Path 39:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
      ...
      432. String[] values = request.getParameterValues(name);
      ...
      439. buffer.append(Util.encode(values[i]));
```

```
146. public static String encode(String string) //util.java
      147. return URLEncoder.encode(string);
```

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
      ...
      439. buffer.append(Util.encode(values[i]));
      ...
      467. return (buffer.toString());
```

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
      397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

```
376. public static String makeQueryString(HttpServletRequest request, String[]
ignores) //util.java
      377. return (makeQueryString(request, new HashMap(), ignores));
```

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    101. String queryString = Util.makeQueryString(request, ignores);
    ...
    106. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
    queryString);
```

[Table of Contents] [Description]

Path 40:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
    70. String endUrl = request.getParameter("endUrl");
    ...
    106. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
    queryString);
```

[Table of Contents] [Description]

Path 41:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    432. String[] values = request.getParameterValues(name);
    ...
    439. buffer.append(Util.encode(values[i]));
```

▼

```
146. public static String encode(String string) //util.java
    147. return URLEncoder.encode(string);
```

▼

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
    ...
    439. buffer.append(Util.encode(values[i]));
    ...
    467. return (buffer.toString());
```

▼

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
    397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

▼

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
    ...
```

```
115. String queryString = Util.makeQueryString(request, additions, ignores);
...
120. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 42:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
70. String endUrl = request.getParameter("endUrl");
...
120. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 43:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
...
432. String[] values = request.getParameterValues(name);
...
439. buffer.append(Util.encode(values[i]));
```

▼

```
146. public static String encode(String string) //util.java
147. return URLEncoder.encode(string);
```

▼

```
416. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores, IgnoreType ignoreType) //util.java
...
439. buffer.append(Util.encode(values[i]));
...
467. return (buffer.toString());
```

▼

```
396. public static String makeQueryString(HttpServletRequest request, Map additions,
String[] ignores) //util.java
397. return (makeQueryString(request, additions, ignores, IgnoreType.BOTH));
```

▼

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
...
120. String queryString = Util.makeQueryString(request, additions, ignores);
...
125. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
queryString);
```

[Table of Contents] [Description]

Path 44:

Query Name - **URL_Redirection_Attack**

Severity - **Warning**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
      ...
      75. String endUrl = request.getParameter("endUrl");
      ...
      125. response.sendRedirect(Util.calculateUrlBase(request) + endUrl + "?" +
      queryString);
```

[Table of Contents] [Description]

Object Model Violation: Just One of Equals and Haschode Defined

CWE ID	581
Description	Software fails to maintain equal hashcodes for equal objects.
Common Consequences	Failure to uphold this invariant is likely to cause trouble if objects of this class are stored in a collection. If the objects of the class in question are used as a key in a Hashtable or if they are inserted into a Map or Set, it is critical that equal objects have equal hashcodes.
Context Notes	Java objects are expected to obey a number of invariants related to equality. One of these invariants is that equal objects must have equal hashcodes. In other words, if a.equals(b) == true then a.hashCode() == b.hashCode().
Node Relationships	ChildOf - Failure to Follow Specification (573)
Applicable Platforms	Java

Path 1:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
50. public class Interface extends StandardNamedObject //interface.java
      '
```

[Table of Contents] [Description]

Path 2:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
56. private static class ServiceComparator //interface.java
      '
```

[Table of Contents] [Description]

Path 3:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
55. private static class InterfaceComparator //node.java
      '
```

[Table of Contents] [Description]

Path 4:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
46. public class Service extends StandardNamedObject //service.java
      '
```

[Table of Contents] [Description]

Path 5:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
215. private static final class LightWeightNodeEntry //reparentviasmb.java
      '
```

[Table of Contents] [Description]

Path 6:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
52. public class MibObject //mibobject.java
      '
```

[Table of Contents] [Description]

Path 7:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
52. final class IPPollAddress //ippolladdress.java
      '
```

[Table of Contents] [Description]

Path 8:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
61. public class EventHandlerMBeanProxy implements
```

```
EventHandler //eventhandlermbeanproxy.java
     |
```

[Table of Contents] [Description]

Path 9:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
62. public class EventKey extends LinkedHashMap implements Serializable,
Comparable //eventkey.java
     |
```

[Table of Contents] [Description]

Path 10:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
54. public class EventMaskValueList extends ArrayList //eventmaskvaluelist.java
     |
```

[Table of Contents] [Description]

Path 11:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
50. static class ListenerKeeper //mockeventipcmanager.java
     |
```

[Table of Contents] [Description]

Path 12:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
38. public class MockNotification //mocknotification.java
     |
```

[Table of Contents] [Description]

Path 13:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
48. public class HttpPostInfo extends Object //httppostinfo.java
     |
```

[Table of Contents] [Description]

Path 14:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
62. public class RTCNode extends Object //rtcnode.java
```

[Table of Contents] [Description]

Path 15:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
56. public class IPv4Address extends Object implements
Serializable //ipv4address.java
```

[Table of Contents] [Description]

Path 16:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class AcknowledgedByFilter extends Object implements
Filter //acknowledgedbyfilter.java
```

[Table of Contents] [Description]

Path 17:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
39. public class AfterDateFilter extends Object implements
Filter //afterdatefilter.java
```

[Table of Contents] [Description]

Path 18:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
39. public class BeforeDateFilter extends Object implements
Filter //beforedatefilter.java
```

[Table of Contents] [Description]

Path 19:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
35. public class DescriptionSubstringFilter extends Object implements
Filter //descriptionsubstringfilter.java
```

[Table of Contents] [Description]

Path 20:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class ExactUEIFilter extends Object implements
Filter //exactueifilter.java
```

[Table of Contents] [Description]

Path 21:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class InterfaceFilter extends Object implements
Filter //interfacefilter.java
```

[Table of Contents] [Description]

Path 22:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class IPLikeFilter extends Object implements Filter //iplikefilter.java
```

[Table of Contents] [Description]

Path 23:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
45. public class LogMessageMatchesAnyFilter extends Object implements
Filter //logmessagematchesanyfilter.java
```

[Table of Contents] [Description]

Path 24:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
35. public class LogMessageSubstringFilter extends Object implements
Filter //logmessagesubstringfilter.java
     '
```

[Table of Contents] [Description]

Path 25:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeAcknowledgedByFilter extends Object implements
Filter //negativeacknowledgedbyfilter.java
     '
```

[Table of Contents] [Description]

Path 26:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeExactUEIFilter extends Object implements
Filter //negativeexactueifilter.java
     '
```

[Table of Contents] [Description]

Path 27:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeInterfaceFilter extends Object implements
Filter //negativeinterfacefilter.java
     '
```

[Table of Contents] [Description]

Path 28:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NegativeNodeFilter extends Object implements
Filter //negativenodefilter.java
     '
```

[Table of Contents] [Description]

Path 29:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativePartialUEIFilter extends Object implements
```

```
Filter //negativepartialueifilter.java
     '
```

[Table of Contents] [Description]

Path 30:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NegativeServiceFilter extends Object implements
Filter //negativeservicefilter.java
     '
```

[Table of Contents] [Description]

Path 31:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
41. public class NegativeSeverityFilter extends Object implements
Filter //negativeseverityfilter.java
     '
```

[Table of Contents] [Description]

Path 32:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NodeFilter extends Object implements Filter //nodefilter.java
     '
```

[Table of Contents] [Description]

Path 33:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NodeNameLikeFilter extends Object implements
Filter //nodenamelikefilter.java
     '
```

[Table of Contents] [Description]

Path 34:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class PartialUEIFilter extends Object implements
Filter //partialueifilter.java
     '
```

[Table of Contents] [Description]

Path 35:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class ServiceFilter extends Object implements Filter //servicefilter.java
```

[Table of Contents] [Description]

Path 36:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
38. public class SeverityFilter extends Object implements
Filter //severityfilter.java
```

[Table of Contents] [Description]

Path 37:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
198. public static class UserFilter extends Object implements
Filter //noticefactory.java
```

[Table of Contents] [Description]

Path 38:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
233. public static class ResponderFilter extends Object implements
Filter //noticefactory.java
```

[Table of Contents] [Description]

Path 39:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
268. public static class NodeFilter extends Object implements
Filter //noticefactory.java
```

[Table of Contents] [Description]

Path 40:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
309. public static class InterfaceFilter extends Object implements
Filter //noticefactory.java
```

[Table of Contents] [Description]

Path 41:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
348. public static class ServiceFilter extends Object implements
Filter //noticefactory.java
```

[Table of Contents] [Description]

Path 42:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class InterfaceFilter extends Object implements
Filter //interfacefilter.java
```

[Table of Contents] [Description]

Path 43:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class LostServiceDateAfterFilter extends Object implements
Filter //lostservicedateafterfilter.java
```

[Table of Contents] [Description]

Path 44:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class LostServiceDateBeforeFilter extends Object implements
Filter //lostservicedatebeforefilter.java
```

[Table of Contents] [Description]

Path 45:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeInterfaceFilter extends Object implements
Filter //negativeinterfacefilter.java
```

[Table of Contents] [Description]

Path 46:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NegativeNodeFilter extends Object implements
Filter //negativenodefilter.java
```

[Table of Contents] [Description]

Path 47:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NegativeServiceFilter extends Object implements
Filter //negativeservicefilter.java
```

[Table of Contents] [Description]

Path 48:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NodeFilter extends Object implements Filter //nodefilter.java
```

[Table of Contents] [Description]

Path 49:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class OutageFilter extends Object implements Filter //outagefilter.java
```

[Table of Contents] [Description]

Path 50:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class RegainedServiceDateAfterFilter extends Object implements
Filter //regainedservicedateafterfilter.java
```

[Table of Contents] [Description]

Path 51:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class RegainedServiceDateBeforeFilter extends Object implements
Filter //regainedservicedatebeforefilter.java
```

[Table of Contents] [Description]

Path 52:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class ServiceFilter extends Object implements Filter //servicefilter.java
```

[Table of Contents] [Description]

Path 53:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class CreationDateAfterFilter extends Object implements
Filter //creationdateafterfilter.java
```

[Table of Contents] [Description]

Path 54:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class CreationDateBeforeFilter extends Object implements
Filter //creationdatebeforefilter.java
```

[Table of Contents] [Description]

Path 55:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class InterfaceFilter extends Object implements
Filter //interfacefilter.java
```

[Table of Contents] [Description]

Path 56:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class IPLikeFilter extends Object implements Filter //iplikefilter.java
```

[Table of Contents] [Description]

Path 57:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeInterfaceFilter extends Object implements
Filter //negativeinterfacefilter.java
```

[Table of Contents] [Description]

Path 58:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NegativeNodeFilter extends Object implements
Filter //negativenodefilter.java
```

[Table of Contents] [Description]

Path 59:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
39. public class NegativePortFilter extends Object implements
Filter //negativeportfilter.java
```

[Table of Contents] [Description]

Path 60:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NegativeProtocolFilter extends Object implements
Filter //negativeprotocolfilter.java
```

[Table of Contents] [Description]

Path 61:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
41. public class NegativeSeverityFilter extends Object implements
Filter //negativeseverityfilter.java
```

[Table of Contents] [Description]

Path 62:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class NodeFilter extends Object implements Filter //nodefilter.java
```

[Table of Contents] [Description]

Path 63:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class NodeNameLikeFilter extends Object implements
Filter //nodenamelikefilter.java
```

[Table of Contents] [Description]

Path 64:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class PortFilter extends Object implements Filter //portfilter.java
```

[Table of Contents] [Description]

Path 65:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
36. public class ProtocolFilter extends Object implements
Filter //protocolfilter.java
```

[Table of Contents] [Description]

Path 66:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class ResolvedDateAfterFilter extends Object implements
Filter //resolveddateafterfilter.java
```

[Table of Contents] [Description]

Path 67:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
40. public class ResolvedDateBeforeFilter extends Object implements
Filter //resolveddatebeforefilter.java
```

[Table of Contents] [Description]

Path 68:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
38. public class SeverityFilter extends Object implements
Filter //severityfilter.java
```

[Table of Contents] [Description]

Path 69:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
48. <%! //interfaceavailability-box.jsp
```

[Table of Contents] [Description]

Path 70:

Query Name - **Equals_without_GetHashCode**

Severity - **Warning**

```
49. <%! //nodeavailability-box.jsp
```

[Table of Contents] [Description]

Improper Transaction Handling

CWE ID	10016
Description	If an exception occurred during a database transaction, a rollback should be committed.
Likelihood of Exploit	Low
Common Consequences	Database lock.
Potential Mitigations	Perform rollback in the catch clause.
Applicable Platforms	All

Path 1:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
862. void store() throws SQLException //dbifserviceentry.java
       ...
865. try {
```

[Table of Contents] [Description]

Path 2:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
847. void store() throws SQLException //dbipinterfaceentry.java
       ...
850. try {
```

[Table of Contents] [Description]

Path 3:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
1502. void store() throws SQLException //dbnodeentry.java
        ...
1505. try {
```

[Table of Contents] [Description]

Path 4:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
924. void store() throws SQLException //dbsnmpinterfaceentry.java
       ...
927. try {
```

[Table of Contents] [Description]

Path 5:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
454. public void commit() throws SQLException //legacydbconnectionfactory.java
     ...
456. try {
```

[Table of Contents] [Description]

Path 6:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
672. public void persistEvent(Header eventHeader, Event event) throws
SQLException //eventwriter.java
     ...
689. try {
```

[Table of Contents] [Description]

Path 7:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
250. public void runUpdate(String sql) //vacuumd.java
     ...
257. try {
```

[Table of Contents] [Description]

Path 8:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodeservlet.java
     ...
130. try {
```

[Table of Contents] [Description]

Path 9:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
105. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //managenodesservlet.java
     ...
130. try {
```

[Table of Contents] [Description]

Path 10:

Query Name - **Improper_Transaction_Handling**

Severity - **Warning**

```
87. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //snmpmanagenodesservlet.java
    ...
115. try {
```

[Table of Contents] [Description]

Error Message Information Leaks

CWE ID	209
Description	Server messages need to be parsed before being passed on to the user.
Likelihood of Exploit	High
Common Consequences	Confidentiality: Often this will either reveal sensitive information which may be used for a later attack or private information stored in the server.
Potential Mitigations	Design: When an application detects illegal input, error messages should only provide generic feedback, such as "Illegal characters were detected." Messages should provide few, if any, implementation details. Implementation: Any error should be parsed for dangerous revelations. Build: Debugging information should not make its way into a production release. Handle exceptions internally and do not display errors containing potentially sensitive information to a user. Create default error pages if necessary.
Demonstrative Examples	**Java Example:** try { /.../ } catch (Exception e) { System.out.println(e); } Here you are passing much more data than is needed. Another example is passing the SQL exceptions to a WebUser without filtering.
Context Notes	Error messages should not provide attackers with any implementation details when the application detects an illegal action. This includes indicating exactly what is allowable, or exactly what was illegal about the user input. Such detailed information can help an attacker craft another attack that now will pass through the validation filters. The first thing an attacker may use -- once an attack has

	failed -- to stage the next attack is the error information provided by the server. SQL Injection attacks generally probe the server for information in order to stage a successful attack.
Node Relationships	ChildOf - Information Leak (Information Disclosure) (200) ParentOf - Product-Generated Error Message Information Leak (210) ParentOf - Product-External Error Message Information Leak (211)
Source Taxonomies	CLASP - Accidental leaking of sensitive information through error messages
Applicable Platforms	All

Path 1:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
31. public static void main(String args[]) //parsemib.java
      ...
      118. System.err.println(e.getMessage());
      119. e.printStackTrace();
```

[Table of Contents] [Description]

Path 2:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
83. public void run() //processexec.java
      ...
      94. e.printStackTrace();
```

[Table of Contents] [Description]

Path 3:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
235. private void populateDataStructures
(org.opennms.netmgt.config.categories.Category cat, Report report, String format,
String monthFormat, int catIndex) throws Exception //availabilitydata.java
      ...
      303. e.printStackTrace();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
189. public void populateReport(String logourl, String categoryName, String
```

```
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
200. e.printStackTrace();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
203. e.printStackTrace();
```

[Table of Contents] [Description]

Path 6:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
206. e.printStackTrace();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
189. public void populateReport(String logourl, String categoryName, String
reportFormat, String monthFormat, String startMonth, String startDate, String
startYear) //availabilityreport.java
    ...
209. e.printStackTrace();
```

[Table of Contents] [Description]

Path 8:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
181. private Report buildReport(Calendar calendar, String
calFormat) //availabilityreporttest.java
    ...
194. e.printStackTrace();
```

[Table of Contents] [Description]

Path 9:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
181. private Report buildReport(Calendar calendar, String
calFormat) //availabilityreporttest.java
      ...
197. e.printStackTrace();
```

[Table of Contents] [Description]

Path 10:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
181. private Report buildReport(Calendar calendar, String
calFormat) //availabilityreporttest.java
      ...
200. e.printStackTrace();
```

[Table of Contents] [Description]

Path 11:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
181. private Report buildReport(Calendar calendar, String
calFormat) //availabilityreporttest.java
      ...
203. e.printStackTrace();
```

[Table of Contents] [Description]

Path 12:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      ...
118. ioe.printStackTrace();
```

[Table of Contents] [Description]

Path 13:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
96. public void generateHTML(FileReader xml, OutputStream fotFileName) throws
MarshalException, ValidationException, IOException, Exception //pdfwriter.java
      ...
123. e.printStackTrace();
```

[Table of Contents] [Description]

Path 14:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      ...
191. ioe.printStackTrace();
```

[Table of Contents] [Description]

Path 15:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
131. public void generatePDF(FileReader xml, OutputStream pdfWriter, String
fotFileName) throws MarshalException, ValidationException, IOException,
Exception //pdfwriter.java
      ...
196. e.printStackTrace();
```

[Table of Contents] [Description]

Path 16:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
998. public static void main(String[] args) //dbifserviceentry.java
      ...
1003. t.printStackTrace();
```

[Table of Contents] [Description]

Path 17:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1107. public static void main(String[] args) //dbipinterfaceentry.java
      ...
1116. t.printStackTrace();
```

[Table of Contents] [Description]

Path 18:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
1055. public static void main(String[] args) //dbsnmpinterfaceentry.java
      ...
1061. t.printStackTrace();
```

[Table of Contents] [Description]

Path 19:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
      ...
102. log.debug(getClass().getName() + ": JDBC stored procedure call not
functional: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
103. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 20:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
195. public void onEvent(Event event) //broadcasteventprocessor.java
      ...
208. e.printStackTrace();
```

[Table of Contents] [Description]

Path 21:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
72. public boolean performUpdate( //stringdatasource.java
      ...
108. e.printStackTrace();
```

[Table of Contents] [Description]

Path 22:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
625. private void scheduledOutagesChangeHandler(Category
log) //pollereventprocessor.java
      ...
```

```
630. e.printStackTrace();
```

[Table of Contents] [Description]

Path 23:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
...
221. exp.printStackTrace();
```

[Table of Contents] [Description]

Path 24:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
...
300. log.debug(getClass().getName() + ": JDBC service is not responding on: "
+ ipv4Addr.getCanonicalHostName() + ", " + sqlEx.getSQLState() + ", " +
sqlEx.toString());
301. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 25:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
...
348. log.debug(getClass().getName() + ": JDBC service failed to retrieve
metadata: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
349. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 26:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
...
99. log.debug(getClass().getName() + ": JDBC stored procedure call not
functional: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
100. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 27:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
59. public static void main(String[] args) //checknsc.java
      ...
109. usage("An error occurred creating a new NsclientManager.", e);
```

```
151. private static void usage(String message, Exception e) //checknsc.java
      ...
165. sb.append(e.getStackTrace()).append("\n\n");
      ...
170. System.out.print(sb);
```

[Table of Contents] [Description]

Path 28:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
415. public void close() //nsclientmanager.java
      ...
419. e.printStackTrace();
```

[Table of Contents] [Description]

Path 29:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
829. private void processPendingOperations() //queuingrrdstrategy.java
      ...
854. printLapTime("Error updating file " + fileName + ": " + e.getMessage());
855. e.printStackTrace();
```

[Table of Contents] [Description]

Path 30:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
864. private void processClose(Object rrd) //queuingrrdstrategy.java
      ...
869. printLapTime("Error closing file " + e.getMessage());
870. e.printStackTrace();
```

[Table of Contents] [Description]

Path 31:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
187. public void onEvent(Event event) //broadcasteventprocessor.java
       ...
200. e.printStackTrace();
```

[Table of Contents] [Description]

Path 32:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
383. public static void main(String[] args) throws IOException //execrunner.java
       ...
415. e.printStackTrace();
```

[Table of Contents] [Description]

Path 33:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
166. public void run() //streamgobbler.java
       ...
187. e.printStackTrace();
```

[Table of Contents] [Description]

Path 34:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
       ...
437. String message = "Could not initialize IcmpSocket: " + t.getMessage();
       ...
440. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Path 35:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
       ...
448. String message = "Could not close test IcmpSocket: " + t.getMessage();
       ...
451. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Path 36:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
       ...
459. String message = "Could not initialize RrdUtils: " + t.getMessage();
...
462. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Path 37:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
108. private void initREs() //nessusparser.java
       ...
160. ex.printStackTrace();
```

[Table of Contents] [Description]

Path 38:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
       ...
243. e.printStackTrace();
```

[Table of Contents] [Description]

Path 39:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
203. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
       ...
246. e.printStackTrace();
```

[Table of Contents] [Description]

Path 40:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
244. e.printStackTrace();
```

[Table of Contents] [Description]

Path 41:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
195. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
247. e.printStackTrace();
```

[Table of Contents] [Description]

Path 42:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
244. e.printStackTrace();
```

[Table of Contents] [Description]

Path 43:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
206. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addpollerconfigservlet.java
    ...
247. e.printStackTrace();
```

[Table of Contents] [Description]

Path 44:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
    ...
249. e.printStackTrace();
```

[Table of Contents] [Description]

Path 45:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
194. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //pollerconfigservlet.java
     ...
252. e.printStackTrace();
```

[Table of Contents] [Description]

Path 46:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
510. public synchronized void start() throws
LifecycleException //opennmstomcatrealm.java
     ...
518. e.printStackTrace();
```

[Table of Contents] [Description]

Path 47:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
277. public int execute() //command.java
     ...
340. System.out.println(e);
```

[Table of Contents] [Description]

Path 48:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
97. public void run() //inetechoxchange.java
     ...
117. e.printStackTrace();
```

[Table of Contents] [Description]

Path 49:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
97. public void run() //inetechoxchange.java
     ...
130. ioE.printStackTrace();
```

[Table of Contents] [Description]

Path 50:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
165. public void run() //inetechoxchange.java
       ...
179. e.printStackTrace();
```

[Table of Contents] [Description]

Path 51:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
165. public void run() //inetechoxchange.java
       ...
209. ioE.printStackTrace();
```

[Table of Contents] [Description]

Path 52:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
245. public static void main(String[] args) //inetechoxchange.java
       ...
277. e.printStackTrace();
```

[Table of Contents] [Description]

Path 53:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
245. public static void main(String[] args) //inetechoxchange.java
       ...
287. ie.printStackTrace();
```

[Table of Contents] [Description]

Path 54:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
53. public static void main (String args[]) //trapd.java
       ...
69. e.printStackTrace();
```

[Table of Contents] [Description]

Path 55:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
31. public static void main(String args[]) //parsemib.java
    ...
119. e.printStackTrace();
```

[Table of Contents] [Description]

Path 56:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
    ...
102. log.debug(getClass().getName() + ": JDBC stored procedure call not
functional: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
103. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 57:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
84. public boolean checkStatus( Category log, Connection con, Map
qualifiers ) //jdbcstoredprocedureplugin.java
    ...
103. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 58:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
    ...
300. log.debug(getClass().getName() + ": JDBC service is not responding on: "
+ ipv4Addr.getCanonicalHostName() + ", " + sqlEx.getSQLState() + ", " +
sqlEx.toString());
301. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 59:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
196. public int poll(NetworkInterface iface, Map parameters,
org.opennms.netmgt.config.poller.Package pkg) //jdbcmonitor.java
    ...
301. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 60:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
    ...
348. log.debug(getClass().getName() + ": JDBC service failed to retrieve
metadata: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
349. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 61:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
322. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcmonitor.java
    ...
349. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 62:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
68. public int checkStatus( Category log, Map parameters, Connection
con ) //jdbcstoredproceduremonitor.java
    ...
99. log.debug(getClass().getName() + ": JDBC stored procedure call not
functional: " + sqlEx.getSQLState() + ", " + sqlEx.toString());
100. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 63:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
68. public int checkStatus( Category log, Map parameters, Connection
```

```
con ) //jdbcstoredproceduremonitor.java
      ...
      100. sqlEx.printStackTrace();
```

[Table of Contents] [Description]

Path 64:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
59. public static void main(String[] args) //checknsc.java
      ...
      117. usage("An error occurred initializing the NsclientManager.", e);
```

```
151. private static void usage(String message, Exception e) //checknsc.java
      ...
      165. sb.append(e.getStackTrace()).append("\n\n");
      ...
      170. System.out.print(sb);
```

[Table of Contents] [Description]

Path 65:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
59. public static void main(String[] args) //checknsc.java
      ...
      124. usage("An error occurred creating the parameter object.", e);
```

```
151. private static void usage(String message, Exception e) //checknsc.java
      ...
      165. sb.append(e.getStackTrace()).append("\n\n");
      ...
      170. System.out.print(sb);
```

[Table of Contents] [Description]

Path 66:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
59. public static void main(String[] args) //checknsc.java
      ...
      133. usage("An error occurred processing the command.", e);
```

```
151. private static void usage(String message, Exception e) //checknsc.java
      ...
      165. sb.append(e.getStackTrace()).append("\n\n");
      ...
```

```
170. System.out.print(sb);
```

[Table of Contents] [Description]

Path 67:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
829. private void processPendingOperations() //queuingrrdstrategy.java
     ...
855. e.printStackTrace();
```

[Table of Contents] [Description]

Path 68:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
864. private void processClose(Object rrd) //queuingrrdstrategy.java
     ...
870. e.printStackTrace();
```

[Table of Contents] [Description]

Path 69:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
     ...
440. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Path 70:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
     ...
451. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Path 71:

Query Name - **Verbose_Error_Reporting**

Severity - **Serious**

```
428. public void doTestLoadLibraries() //manager.java
```

```
...
462. t.printStackTrace(System.err);
```

[Table of Contents] [Description]

Resource Injection

CWE ID	99
Description	Allowing user input to control resource identifiers may enable an attacker to access or modify otherwise protected system resources.
Likelihood of Exploit	High
Weakness Ordinality	Primary (Weakness exists independent of other weaknesses)
Causal Nature	Explicit (This is an explicit weakness resulting from behavior of the developer)
Potential Mitigations	Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Demonstrative Examples	The following Java code uses input from an HTTP request to create a file name. The programmer has not considered the possibility that an attacker could provide a file name such as "../../tomcat/conf/server.xml", which causes the application to delete one of its own configuration files.

Java Example:

```
String rName = request.getParameter("reportName");
File rFile = new File("/usr/local/apfr/reports/" + rName);
...
rFile.delete();
```

The following code uses input from the command line to determine which file to open and echo back to the user. If the program runs with privileges and malicious users can create soft links to the file, they can use the program to read the first part of any file on the system.

C++ Example:

```
ifstream ifs(argv[0]);
string s;
ifs >> s;
cout << s;
```

The kind of resource the data affects indicates the kind of content that may be dangerous. For example, data containing special characters like period, slash, and backslash, are risky

	when used in methods that interact with the file system. (Resource injection, when it is related to file system resources, sometimes goes by the name "path manipulation.") Similarly, data that contains URLs and URIs is risky for functions that create remote connections.
Context Notes	A resource injection issue occurs when the following two conditions are met: 1. An attacker can specify the identifier used to access a system resource. For example, an attacker might be able to specify part of the name of a file to be opened or a port number to be used. 2. By specifying the resource, the attacker gains a capability that would not otherwise be permitted. For example, the program may give the attacker the ability to overwrite the specified file, run with a configuration controlled by the attacker, or transmit sensitive information to a third-party server. Note: Resource injection that involves resources stored on the filesystem goes by the name path manipulation and is reported in separate category. See the path manipulation description for further details of this vulnerability.
Node Relationships	ChildOf - Injection (74) CanAlsoBe - Path Manipulation (73) ChildOf - Weaknesses Examined by SAMATE (630)
Source Taxonomies	7 Pernicious Kingdoms - Resource Injection
Applicable Platforms	All

Path 1:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
```

```
212. private java.util.List getList(String array[]) //deletenodesservlet.java
      ...
      217. newList.add(array[i]);
      ...
      221. return newList;
```

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
      ...
      121. File nodeDir = new File(this.snmpRrdDirectory, (String) nodeDataList.get
      (j));
```

[Table of Contents] [Description]

Path 2:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
    ...
    105. java.util.List nodeDataList = getList(request.getParameterValues
    ("nodeData"));
```

```
212. private java.util.List getList(String array[]) //deletenodesservlet.java
    ...
    217. newList.add(array[i]);
    ...
    221. return newList;
```

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
    ...
    105. java.util.List nodeDataList = getList(request.getParameterValues
    ("nodeData"));
    ...
    130. select.append((String) nodeDataList.get(j));
    ...
    138. ResultSet rs = stmt.executeQuery(select.toString());
    ...
    141. String ipAddr = rs.getString("ipaddr");
    ...
    143. File intfDir = new File(this.rtRrdDirectory, ipAddr);
```

[Table of Contents] [Description]

Path 3:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
161. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //rrdgraphservlet.java
    ...
    165. String propertiesFile = request.getParameter("props");
    ...
    182. String command = this.createPrefabCommand(request, report, rrds,
    propertiesFile, start, end);
```

```
213. protected String createPrefabCommand(HttpServletRequest request, String
reportName, String[] rrds, String propertiesFile, String start, String end) throws
ServletException //rrdgraphservlet.java
    ...
    252. fileInputStream = new FileInputStream(this.workDir + File.separator +
    propertiesFile);
```

[Table of Contents] [Description]

Path 4:

Query Name - **Files_Manipulation**

Severity

- **Serious**

```
60. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    62. String nodeIdString = request.getParameter("node");
    ...
    78. queries = this.model.getQueries(nodeIdString);
```

```
135. public PrefabGraph[] getQueries(String nodeId) //performancemodel.java
    ...
    147. List availDataSourceList = this.getDataSourceList(nodeId);
```

```
228. public List getDataSourceList(String nodeId) //performancemodel.java
    ...
    234. File nodeDir = new File(this.rrdDirectory, nodeId);
```

[Table of Contents] [Description]

Path 5:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
60. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    62. String nodeIdString = request.getParameter("node");
    ...
    81. queries = this.model.getQueries(nodeIdString, intf, showNodeQueries);
```

```
170. public PrefabGraph[] getQueries(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
    ...
    183. List availDataSourceList = this.getDataSourceList(nodeId, intf,
    includeNodeQueries);
```

```
254. public List getDataSourceList(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
    ...
    261. File nodeDir = new File(this.rrdDirectory, nodeId);
```

[Table of Contents] [Description]

Path 6:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
60. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    68. String intf = request.getParameter("intf");
```

```
...
81. queries = this.model.getQueries(nodeIdString, intf, showNodeQueries);
```

```
170. public PrefabGraph[] getQueries(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java

...
183. List availDataSourceList = this.getDataSourceList(nodeId, intf,
includeNodeQueries);
```

```
250. public List getDataSourceList(int nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java

251. return this.getDataSourceList(String.valueOf(nodeId), intf,
includeNodeQueries);
```

```
254. public List getDataSourceList(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java

...
262. File intfDir = new File(nodeDir, intf);
```

[Table of Contents] [Description]

Path 7:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
68. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java

69. String nodeId = request.getParameter("node");
...
81. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
424. public String[] getQueryableInterfacesForNode(String
nodeId) //performancemodel.java

...
430. File nodeDir = new File(this.rrdDirectory, nodeId);
```

[Table of Contents] [Description]

Path 8:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //interface.jsp

...
93. String ipAddr = request.getParameter( "intf" );
...
227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
ifindexString); %>
```

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
```

```
...
261. stmt.setString(2, ipAddr);
262. stmt.setString(3, ifIndex);
...
264. ResultSet rs = stmt.executeQuery();
...
268. String descr = rs.getString("snmpifdescr");
...
272. label = getIfLabel(name, descr, physAddr);
```

▼

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
...
306. label = AlphaNumeric.parseAndReplace(descr, '_');
```

▼

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java
...
67. byte[] bytes = str.getBytes();
...
76. String temp = new String(bytes);
...
86. return temp;
```

▼

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
...
306. label = AlphaNumeric.parseAndReplace(descr, '_');
...
326. return label;
```

▼

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
...
272. label = getIfLabel(name, descr, physAddr);
...
289. return label;
```

▼

```
1. <!-- //interface.jsp
...
227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
ifindexString); %>
228. <% if(ifLabel != null && this.perfModel.isQueryableInterface(nodeId,
ifLabel)) { %>
```

▼

```
478. public boolean isQueryableInterface(int nodeId, String
ifLabel) //performancemodel.java
...
487. File intfDir = new File(nodeDir, ifLabel);
```

[Table of Contents] [Description]

Path 9:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
64. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    66. String nodeIdString = request.getParameter("node");
    ...
    81. queries = this.model.getQueries(nodeIdString);
```

```
142. public PrefabGraph[] getQueries(String nodeId) //responsetimemodel.java
    ...
    154. List availDataSourceList = this.getDataSourceList(nodeId);
```

```
235. public List getDataSourceList(String nodeId) //responsetimemodel.java
    ...
    241. File nodeDir = new File(this.rrdDirectory, nodeId);
```

[Table of Contents] [Description]

Path 10:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
64. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addreportstourlservlet.java
    ...
    71. String intf = request.getParameter("intf");
    ...
    84. queries = this.model.getQueries(nodeIdString, intf, showNodeQueries);
```

```
177. public PrefabGraph[] getQueries(String nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
    ...
    190. List availDataSourceList = this.getDataSourceList(nodeId, intf,
    includeNodeQueries);
```

```
257. public List getDataSourceList(int nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
    258. return this.getDataSourceList(String.valueOf(nodeId), intf,
    includeNodeQueries);
```

```
261. public List getDataSourceList(String nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
    ...
    269. File intfDir = new File(this.rrdDirectory, intf);
```

[Table of Contents] [Description]

Path 11:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
      74. String nodeId = request.getParameter("node");
      ...
      89. intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
      ...
      437. select.append(nodeId);
      ...
      447. ResultSet rs = stmt.executeQuery(select.toString());
      ...
      450. String ipAddr = rs.getString("ipaddr");
      451. File intfDir = new File(this.rrdDirectory, ipAddr);
```

[Table of Contents] [Description]

Path 12:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //interface.jsp
      ...
      93. String ipAddr = request.getParameter( "intf" );
      ...
      216. <% if(this.rtModel.isQueryableInterface(ipAddr)) { %>
```

```
482. public boolean isQueryableInterface(String ipAddr) //responsetimemodel.java
      ...
      488. File intfDir = new File(this.rrdDirectory, ipAddr);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //adhoc2.jsp
      ...
      59. String nodeIdString = request.getParameter("node");
      ...
      75. rrdPath = new File(this.model.getRrdDirectory(), nodeIdString);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //adhoc2.jsp
      ...
```

```
59. String nodeIdString = request.getParameter("node");
...
79. File nodeDir = new File(this.model.getRrdDirectory(), nodeIdString);
```

[Table of Contents] [Description]

Path 15:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //adhoc2.jsp
...
65. String intf = request.getParameter("intf");
...
80. rrdPath = new File(nodeDir, intf);
```

[Table of Contents] [Description]

Path 16:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //adhoc2.jsp
...
67. String intf = request.getParameter("intf");
...
80. rrdPath = new File(this.model.getRrdDirectory(), intf);
```

[Table of Contents] [Description]

Path 17:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //custom_graph3.jsp
...
59. int nodeId = Integer.parseInt(nodeIdString);
...
65. graph_options = this.model.getQueries(nodeId, intf, includeNodeQueries);
```

```
166. public PrefabGraph[] getQueries(int nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
167. return this.getQueries(String.valueOf(nodeId), intf, includeNodeQueries);
```

```
170. public PrefabGraph[] getQueries(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
...
183. List availDataSourceList = this.getDataSourceList(nodeId, intf,
includeNodeQueries);
```

```
250. public List getDataSourceList(int nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
```

```
251. return this.getDataSourceList(String.valueOf(nodeId), intf,
includeNodeQueries);
```

```
254. public List getDataSourceList(String nodeId, String intf, boolean
includeNodeQueries) //performancemodel.java
    ...
262. File intfDir = new File(nodeDir, intf);
```

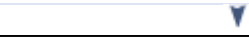
[Table of Contents] [Description]

Path 18:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //custom_graph2.jsp
    ...
55. String nodeId = request.getParameter("node");
...
64. String[] intfs = this.model.getQueryableInterfacesForNode(nodeId);
```



```
424. public String[] getQueryableInterfacesForNode(String
nodeId) //performancemodel.java
    ...
430. File nodeDir = new File(this.rrdDirectory, nodeId);
```

[Table of Contents] [Description]

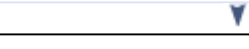
Path 19:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //interface.jsp
    ...
94. String ifindexString = request.getParameter( "ifindex" );
...
227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
ifindexString); %>
```

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
    ...
262. stmt.setString(3, ifIndex);
...
264. ResultSet rs = stmt.executeQuery();
...
268. String descr = rs.getString("snmpifdescr");
...
272. label = getIfLabel(name, descr, physAddr);
```



```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
    ...
```

```
306. label = AlphaNumeric.parseAndReplace(descr, '_');
```

```
62. public static String parseAndReplace(String str, char
replacement) //alphanumeric.java
    ...
    67. byte[] bytes = str.getBytes();
    ...
    76. String temp = new String(bytes);
    ...
    86. return temp;
```

```
292. public static String getIfLabel(String name, String descr, String
physAddr) //iflabel.java
    ...
    306. label = AlphaNumeric.parseAndReplace(descr, '_');
    ...
    326. return label;
```

```
250. public static String getIfLabelfromIfIndex(int nodeId, String ipAddr, String
ifIndex) throws SQLException //iflabel.java
    ...
    272. label = getIfLabel(name, descr, physAddr);
    ...
    289. return label;
```

```
1. <!-- //interface.jsp
    ...
    227. <% String ifLabel = IfLabel.getIfLabelfromIfIndex(nodeId, ipAddr,
    ifindexString); %>
    228. <% if(ifLabel != null && this.perfModel.isQueryableInterface(nodeId,
    ifLabel)) { %>
```

```
478. public boolean isQueryableInterface(int nodeId, String
ifLabel) //performancemodel.java
    ...
    487. File intfDir = new File(nodeDir, ifLabel);
```

[Table of Contents] [Description]

Path 20:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //choosereportanddate.jsp
    ...
    68. String intf = request.getParameter("intf");
    ...
    77. graphs = this.model.getQueries(nodeId, intf, includeNodeQueries);
```

```
173. public PrefabGraph[] getQueries(int nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
    174. return this.getQueries(String.valueOf(nodeId), intf, includeNodeQueries);
```

```
177. public PrefabGraph[] getQueries(String nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
        ...
        190. List availDataSourceList = this.getDataSourceList(nodeId, intf,
        includeNodeQueries);
```

```
257. public List getDataSourceList(int nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
        258. return this.getDataSourceList(String.valueOf(nodeId), intf,
        includeNodeQueries);
```

```
261. public List getDataSourceList(String nodeId, String intf, boolean
includeNodeQueries) //responsetimemodel.java
        ...
        269. File intfDir = new File(this.rrdDirectory, intf);
```

[Table of Contents] [Description]

Path 21:

Query Name - **Files_Manipulation**

Severity - **Serious**

```
1. <!-- //chooseinterface.jsp
        ...
        60. String nodeId = request.getParameter("node");
        ...
        71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
        ...
        437. select.append(nodeId);
        ...
        447. ResultSet rs = stmt.executeQuery(select.toString());
        ...
        450. String ipAddr = rs.getString("ipaddr");
        451. File intfDir = new File(this.rrdDirectory, ipAddr);
```

[Table of Contents] [Description]

	Cross-Site Request Forgery (CSRF)
CWE ID	352
Description	The web product does not, or can not, sufficiently verify whether a well-formed, valid, consistent request was intentionally provided by the user who submitted the request. Note: CSRF is multi-channel: 1. Attacker-to-victim (injection; external or internal channel) 2. Victim-to-server (activation; internal channel)
Alternate Terms	Session Riding Cross Site Reference Forgery

	XSRF
Observed Examples	CVE-2004-1703 -
	CVE-2004-1995 -
	CVE-2004-1967 -
	CVE-2004-1842 -
	CVE-2005-1947 -
	CVE-2005-2059 -
	CVE-2005-1674 - CSRF
Context Notes	Could be resultant from XSS, although XSS is not necessarily required.
References	Peter W. "Cross-Site Request Forgeries (Re: The Dangers of Allowing Users to Post Images)". Bugtraq. <http://marc.info/?l=bugtraq&m=99263135911884&w=2>. Robert Auger. "CSRF - The Cross-Site Request Forgery (CSRF/XSRF) FAQ". <http://www.cgisecurity.com/articles/csrf-faq.shtml>.
Node Relationships	ChildOf - Insufficient Verification of Data (345) ChildOf - Weaknesses in OWASP Top Ten (629) ChildOf - Weaknesses Used by NVD (635)
Source Taxonomies	PLOVER - Cross-Site Request Forgery (CSRF)
Applicable Platforms	All

Path 1:

Query Name - **XSRF**

Severity - **Serious**

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
      ...
      84. String[] eventIdStrings = request.getParameterValues("event");
      ...
      98. eventIds[i] = Integer.parseInt(eventIdStrings[i]);
      ...
      103. EventFactory.acknowledge(eventIds, request.getRemoteUser());
```

```
775. public static void acknowledge(int[] eventIds, String user) throws
SQLException //eventfactory.java
      776. acknowledge(eventIds, user, new Date());
```

```
782. public static void acknowledge(int[] eventIds, String user, Date time) throws
SQLException //eventfactory.java

      ...
      790. update.append(eventIds[0]);
      ...
      803. PreparedStatement stmt = conn.prepareStatement(update.toString());
      ...
      843. PreparedStatement stmt = conn.prepareStatement(update.toString());
```

```
...
847. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 2:

Query Name - **XSRF**

Severity - **Serious**

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
...
84. String[] eventIdStrings = request.getParameterValues("event");
...
98. eventIds[i] = Integer.parseInt(eventIdStrings[i]);
...
103. EventFactory.acknowledge(eventIds, request.getRemoteUser());
```

```
775. public static void acknowledge(int[] eventIds, String user) throws
SQLException //eventfactory.java
776. acknowledge(eventIds, user, new Date());
```

```
827. public static void acknowledge(Filter[] filters, String user, Date time) throws
SQLException //eventfactory.java
...
837. update.append(filters[i].getSql());
...
927. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 3:

Query Name - **XSRF**

Severity - **Serious**

```
82. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgeeventservlet.java
...
84. String[] eventIdStrings = request.getParameterValues("event");
...
98. eventIds[i] = Integer.parseInt(eventIdStrings[i]);
...
105. EventFactory.unacknowledge(eventIds);
```

```
906. public static void unacknowledge(int[] eventIds) throws
SQLException //eventfactory.java
...
914. update.append(eventIds[0]);
...
926. PreparedStatement stmt = conn.prepareStatement(update.toString());
927. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 4:

Query Name - **XSRF**

Severity - **Serious**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparentservlet.java
    ...
66. String childID = request.getParameter("childID");
    ...
76. stmt.setInt(1, new Integer(parentID).intValue());
77. stmt.setInt(2, new Integer(childID).intValue());
78. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 5:

Query Name - **XSRF**

Severity - **Serious**

```
78. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //acknowledgenoticeservlet.java
    ...
80. String[] noticeIdStrings = request.getParameterValues("notices");
    ...
89. noticeIds[i] = Integer.parseInt(noticeIdStrings[i]);
    ...
94. NoticeFactory.acknowledge(noticeIds, request.getRemoteUser());
```

▼

```
830. public static void acknowledge(int[] noticeIds, String user) throws
SQLException //noticefactory.java
    831. acknowledge(noticeIds, user, new Date());
```

▼

```
837. public static void acknowledge(int[] noticeIds, String user, Date time) throws
SQLException //noticefactory.java
    ...
845. update.append(noticeIds[0]);
    ...
853. update.append(" AND RESPONDTIME IS NULL");
    ...
858. PreparedStatement stmt = conn.prepareStatement(update.toString());
    ...
862. stmt.executeUpdate();
```

[Table of Contents] [Description]

Path 6:

Query Name - **XSRF**

Severity - **Serious**

```
1. <!-- //acknowledge.jsp
    ...
49. String[] noticeIds = request.getParameterValues( "notices" );
    ...
69. this.model.acknowledged( username, Integer.parseInt( noticeIds[i] ));
```

▼

```
424. public void acknowledged(String name, int noticeId) throws
SQLException //notificationmodel.java

      ...
      435. pstmt.setInt(3, noticeId);
      436. pstmt.execute();
```

[Table of Contents] [Description]

Path 7:

Query Name - **XSRF**

Severity - **Serious**

```
63. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //modifyparentservlet.java

      ...
      66. String childID = request.getParameter("childID");
      ...
      77. stmt.setInt(2, new Integer(childID).intValue());
      78. stmt.executeUpdate();
```

[Table of Contents] [Description]

Web Parameter Tampering

CWE ID	472
Description	A web product does not properly protect assumed-immutable values from modification in hidden form fields, parameters, cookies, or URLs, which lead to modification of critical data.
Alternate Terms	Assumed-Immutable Parameter Tampering
Potential Mitigations	Do not pass hidden fields or cookies (apart from a session cookie) to the server. And, do not use parameters that are not from immediate user input. Assume all input is malicious. Use an appropriate combination of black lists and white lists to ensure only valid and expected input is processed by the system.
Observed Examples	CVE-2002-0108 - Forum product allows spoofed messages of other users via hidden form fields for name and e-mail address. CVE-2000-0253 - Shopping cart allows price modification via hidden form field. CVE-2000-0254 - Shopping cart allows price modification via hidden form field. CVE-2000-0926 - Shopping cart allows price modification via hidden form field. CVE-2000-0101 - Shopping cart allows price modification via hidden form field. CVE-2000-0102 - Shopping cart allows price modification via

hidden form field.

CVE-2000-0758 - Allows admin access by modifying value of form field.

CVE-2002-1880 - Read messages by modifying message ID parameter.

CVE-2000-1234 - Send email to arbitrary users by modifying email parameter.

CVE-2005-1652 - Authentication bypass by setting a parameter.

CVE-2005-1784 - Product does not check authorization for configuration change admnin script, leading to password theft via modified e-mail address field.

CVE-2005-2314 - Logic error leads to password disclosure.

CVE-2005-1682 - Modification of message number parameter allows attackers to read other people's messages.

Context Notes

This is a primary weakness for many other weaknesses and functional consequences, including XSS, SQL injection, path disclosure, and file inclusion. For example, custom cookies commonly store session data or persistent data across sessions. This kind of session data is normally involved in security related decisions on the server side, such as user authentication and access control. Thus, the cookies might contain sensitive data such as user credentials and privileges. This is a dangerous practice, as it can often lead to improper reliance on the value of the client-provided cookie by the server side application. Without appropriate protection mechanisms, the client can easily tamper with cookies. Reliance on the cookies without detailed validation can lead to problems such as SQL injection. If you use cookie values for security related decisions on the server side, manipulating the cookies might lead to violations of security policies such as authentication bypassing, user impersonation and privilege escalation. In addition, storing sensitive data in the cookie without appropriate protection can also lead to disclosure of sensitive user data, especially data stored in persistent cookies.

Hidden fields should not be trusted as secure parameters. An attacker can intercept and alter hidden fields in a post to the server as easily as user input fields. An attacker can simply parse the HTML for the substring < input type "hidden" or even just "hidden". Hidden field values displayed later in the session, such as on the following page, can open a site up to cross-site scripting attacks.

Node Relationships

This is a technology-specific MAID problem.

ChildOf - Modification of Assumed-Immutable Data (MAID) (471)

ChildOf - Weaknesses in OWASP Top Ten (629)
ParentOf - Unchecked Array Indexing (129)

Source Taxonomies PLOVER - Web Parameter Tampering

Applicable Platforms All

Path 1:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //custom_graph2.jsp
      ...
      55. String nodeId = request.getParameter("node");
      ...
      142. <%=this.model.getHumanReadableNameForIfLabel(Integer.parseInt(nodeId),
      intfs[i])%></option>
```

▼

```
288. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //performancemodel.java
      ...
      297. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

▼

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
      ...
      169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

▼

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
      ...
      106. String query = "SELECT * FROM snmpinterface WHERE nodeid = '" +
      String.valueOf(nodeId) + "'";
      ...
      109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 2:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
```

▼

```
212. private java.util.List getList(String array[]) //deletenodesservlet.java
      ...
      217. newList.add(array[i]);
      ...
      221. return newList;
```

▼

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
        ...
        105. java.util.List nodeDataList = getList(request.getParameterValues
        ("nodeData"));
        ...
        130. select.append((String) nodeDataList.get(j));
        ...
        138. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 3:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //modify.jsp
        ...
        49. String nodeIdString = request.getParameter( "node" );
        ...
        55. int nodeId = Integer.parseInt( nodeIdString );
        ...
        57. Asset asset = this.model.getAsset( nodeId );
```

```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java
        ...
        59. ResultSet rs = stmt.executeQuery("SELECT * FROM ASSETS WHERE NODEID=" +
        nodeId);
```

[Table of Contents] [Description]

Path 4:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //updatesnmp.jsp
        ...
        68. String nodeIdString = request.getParameter("node");
        ...
        79. int nodeId = Integer.parseInt(nodeIdString);
        80. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeId);
```

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
        ...
        98. stmt.setInt(1, nodeId);
        99. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 5:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //deletenode.jsp
      ...
      45. String nodeIdString = request.getParameter("node");
      ...
      52. nodeId = Integer.parseInt(nodeIdString);
      ...
      61. Node node_db = NetworkElementFactory.getNode(nodeId);
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
      ...
      120. stmt.setInt(1, nodeId);
      121. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 6:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //detail.jsp
      ...
      45. String eventIdString = request.getParameter( "id" );
      ...
      54. eventId = Integer.parseInt( eventIdString );
      ...
      60. Event event = EventFactory.getEvent( eventId );
```

```
273. public static Event getEvent(int eventId) throws
SQLException //eventfactory.java
      ...
      279. stmt.setInt(1, eventId);
      280. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 7:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
      74. String nodeId = request.getParameter("node");
      ...
      89. intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
      ...
      437. select.append(nodeId);
      ...
      447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 8:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //detail.jsp
      ...
      49. String vulIdString = request.getParameter("id");
      ...
      58. vulId = Integer.parseInt( vulIdString );
      ...
      64. Vulnerability vul = VulnerabilityFactory.getVulnerability( vulId );
```

```
244. public static Vulnerability getVulnerability(int vulId) throws
SQLException //vulnerabilityfactory.java
      ...
      250. stmt.setInt(1, vulId);
      251. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 9:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //custom_graph3.jsp
      ...
      48. String nodeIdString = request.getParameter("node");
      ...
      59. int nodeId = Integer.parseInt(nodeIdString);
      ...
      171. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
288. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //performancemodel.java
      ...
      297. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
      ...
      169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
      ...
      106. String query = "SELECT * FROM snmpinterface WHERE nodeid = '" +
      String.valueOf(nodeId) + "'";
      ...
      109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 10:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //adhoc.jsp
       ...
       59. String nodeIdString = request.getParameter("node");
       ...
       64. int nodeId = Integer.parseInt(nodeIdString);
       ...
       71. intfMap.put(this.model.getHumanReadableNameForIfLabel(nodeId, (String)
       intfs.get(i)), (String)intfs.get(i));
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
       ...
       304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
       ...
       169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```

```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java
       ...
       106. String query = "SELECT * FROM snmpinterface WHERE nodeid = '" +
       String.valueOf(nodeId) + "'";
       ...
       109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 11:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //adhoc2.jsp
       ...
       61. String nodeIdString = request.getParameter("node");
       ...
       72. int nodeId = Integer.parseInt(nodeIdString);
       ...
       133. Node: <%=nodeLabel%>    Interface: <%
       =this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```

```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java
       ...
       304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```

```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java
```

```
...
169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```



```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java

...
106. String query = "SELECT * FROM snmpinterface WHERE nodeid = '" +
String.valueOf(nodeId) + "'";
...
109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 12:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //results.jsp

...
70. String nodeIdString = request.getParameter( "node" );
...
74. int nodeId = Integer.parseInt(nodeIdString);
...
212. Interface: <%=this.model.getHumanReadableNameForIfLabel(nodeId, intf)%>
```



```
295. public String getHumanReadableNameForIfLabel(int nodeId, String ifLabel) throws
SQLException //responsetimemodel.java

...
304. Map intfMap = IfLabel.getInterfaceInfoFromIfLabel(nodeId, ifLabel);
```



```
164. public static Map getInterfaceInfoFromIfLabel(int nodeId, String ifLabel)
throws SQLException //iflabel.java

...
169. info = getInterfaceInfoFromIfLabel(conn, nodeId, ifLabel);
```



```
86. public static Map getInterfaceInfoFromIfLabel(Connection conn, int nodeId,
String ifLabel) throws SQLException //iflabel.java

...
106. String query = "SELECT * FROM snmpinterface WHERE nodeid = '" +
String.valueOf(nodeId) + "'";
...
109. ResultSet rs = stmt.executeQuery(query);
```

[Table of Contents] [Description]

Path 13:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //node.jsp

...
114. String nodeIdString = request.getParameter( "node" );
```

```
...
120. int nodeId = Integer.parseInt( nodeIdString );
...
194. Asset asset = this.model.getAsset( nodeId );
```



```
53. public Asset getAsset(int nodeId) throws SQLException //assetmodel.java

    ...
    59. ResultSet rs = stmt.executeQuery("SELECT * FROM ASSETS WHERE NODEID=" +
    nodeId);
```

[Table of Contents] [Description]

Path 14:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //rescan.jsp

    ...
    44. String nodeIdString = request.getParameter("node");
    ...
    51. int nodeId = Integer.parseInt(nodeIdString);
    52. String nodeLabel = NetworkElementFactory.getNodeLabel(nodeId);
```



```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

    ...
    98. stmt.setInt(1, nodeId);
    99. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 15:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //custom_graph3.jsp

    ...
    48. String nodeIdString = request.getParameter("node");
    ...
    59. int nodeId = Integer.parseInt(nodeIdString);
    ...
    169. <%=NetworkElementFactory.getNodeLabel(nodeId)%></a><br>
```



```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java

    ...
    98. stmt.setInt(1, nodeId);
    99. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 16:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //results.jsp
       ...
       68. String nodeIdString = request.getParameter( "node" );
       ...
       72. int nodeId = Integer.parseInt(nodeIdString);
       ...
       205. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
       =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```



```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
       ...
       98. stmt.setInt(1, nodeId);
       99. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 17:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //zoom.jsp
       ...
       68. String nodeIdString = request.getParameter( "node" );
       ...
       72. int nodeId = Integer.parseInt(nodeIdString);
       ...
       135. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
       =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```



```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
       ...
       98. stmt.setInt(1, nodeId);
       99. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 18:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //results.jsp
       ...
       70. String nodeIdString = request.getParameter( "node" );
       ...
       74. int nodeId = Integer.parseInt(nodeIdString);
       ...
       210. Node: <a href="element/node.jsp?node=<%=nodeId%>"><%
       =NetworkElementFactory.getNodeLabel(nodeId)%></a><br/>
```



```
92. public static String getNodeLabel(int nodeId) throws
```

```
SQLException //networkelementfactory.java
        ...
98.     stmt.setInt(1, nodeId);
99.     ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 19:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //category.jsp
        ...
66.     String categoryName = request.getParameter("category");
        ...
72.     Category category = this.model.getCategory(categoryName);
```

```
94. public Category getCategory(String categoryName) //categorymodel.java
        ...
99.     return (Category) this.categories.get(categoryName);
```

```
1. <!-- //category.jsp
        ...
72.     Category category = this.model.getCategory(categoryName);
        ...
80.     Enumeration nodeEnum = category.enumerateNode();
        ...
83.     Node node = (Node)nodeEnum.nextElement();
84.     String nodeLabel = NetworkElementFactory.getNodeLabel((int)node.getNodeid
        ());
```

```
92. public static String getNodeLabel(int nodeId) throws
SQLException //networkelementfactory.java
        ...
98.     stmt.setInt(1, nodeId);
99.     ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 20:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //index.jsp
        ...
47.     String nodeIdString = request.getParameter( "node" );
        ...
53.     nodeId = Integer.parseInt( nodeIdString );
        ...
62.     Node node_db = NetworkElementFactory.getNode( nodeId );
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
```

```
...
120. stmt.setInt(1, nodeId);
121. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 21:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //modify.jsp
...
49. String nodeIdString = request.getParameter( "node" );
...
55. int nodeId = Integer.parseInt( nodeIdString );
...
58. Node node_db = NetworkElementFactory.getNode( nodeId );
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
120. stmt.setInt(1, nodeId);
121. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 22:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //node.jsp
...
114. String nodeIdString = request.getParameter( "node" );
...
120. int nodeId = Integer.parseInt( nodeIdString );
...
123. Node node_db = NetworkElementFactory.getNode( nodeId );
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
120. stmt.setInt(1, nodeId);
121. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 23:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //nodeavailability-box.jsp
...
```

```
75. String nodeIdString = request.getParameter( "node" );
...
81. int nodeId = Integer.parseInt( nodeIdString );
...
84. Node node_db = NetworkElementFactory.getNode( nodeId );
```

```
114. public static Node getNode(int nodeId) throws
SQLException //networkelementfactory.java
...
120. stmt.setInt(1, nodeId);
121. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 24:

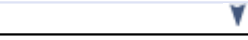
Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //node.jsp
...
114. String nodeIdString = request.getParameter( "node" );
...
120. int nodeId = Integer.parseInt( nodeIdString );
...
245. <% if(this.rtModel.isQueryableNode(nodeId)) { %>
```

```
466. public boolean isQueryableNode(int nodeId) throws
SQLException //responsetimemodel.java
467. return this.isQueryableNode(String.valueOf(nodeId));
```

```
470. public boolean isQueryableNode(String nodeId) throws
SQLException //responsetimemodel.java
...
475. ArrayList intfList = getQueryableInterfacesForNode(nodeId);
```



```
425. public ArrayList getQueryableInterfacesForNode(int nodeId) throws
SQLException //responsetimemodel.java
426. return this.getQueryableInterfacesForNode(String.valueOf(nodeId));
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
...
437. select.append(nodeId);
...
447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 25:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //adhoc.jsp
      ...
      59. String nodeIdString = request.getParameter("node");
      ...
      64. int nodeId = Integer.parseInt(nodeIdString);
      ...
      67. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
425. public ArrayList getQueryableInterfacesForNode(int nodeId) throws
SQLException //responsetimemodel.java
      426. return this.getQueryableInterfacesForNode(String.valueOf(nodeId));
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
      ...
      437. select.append(nodeId);
      ...
      447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 26:

Query Name - **Parameter_Tampering**

Severity - **Serious**

```
1. <!-- //chooseinterface.jsp
      ...
      60. String nodeId = request.getParameter("node");
      ...
      71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
      ...
      437. select.append(nodeId);
      ...
      447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

SQL Injection

CWE ID	89
Description	SQL injection attacks are another instantiation of injection attack, in which SQL commands are injected into data-plane input in order to effect the execution of predefined SQL commands.
Likelihood of Exploit	Very High
Common Consequences	Confidentiality: Since SQL databases generally hold sensitive data, loss of confidentiality is a frequent problem with SQL injection vulnerabilities. Authentication: If poor SQL commands are used to check

user names and passwords, it may be possible to connect to a system as another user with no previous knowledge of the password.

Authorization: If authorization information is held in a SQL database, it may be possible to change this information through the successful exploitation of a SQL injection vulnerability.

Integrity: Just as it may be possible to read sensitive information, it is also possible to make changes or even delete this information with a SQL injection attack.

Potential Mitigations

Requirements specification: A non-SQL style database which is not subject to this flaw may be chosen.

Design: Follow the principle of least privilege when creating user accounts to a SQL database. Users should only have the minimum privileges necessary to use their account. If the requirements of the system indicate that a user can read and modify their own data, then limit their privileges so they cannot read/write others' data.

Design: Duplicate any filtering done on the client-side on the server side.

Implementation: Implement SQL strings using prepared statements that bind variables. Prepared statements that do not bind variables can be vulnerable to attack.

Implementation: Use vigorous white-list style checking on any user input that may be used in a SQL command. Rather than escape meta-characters, it is safest to disallow them entirely. Reason: Later use of data that have been entered in the database may neglect to escape meta-characters before use. Narrowly define the set of safe characters based on the expected value of the parameter in the request.

Demonstrative Examples

The following code dynamically constructs and executes a SQL query that searches for items matching a specified name. The query restricts the items displayed to those where owner matches the user name of the currently-authenticated user.

C# Example:

```
...
string userName = ctx.getAuthenticatedUserName();
string query = "SELECT * FROM items WHERE owner = '"
+ userName + "' AND itemname = '" + ItemName.Text +
"'";
sda = new SqlDataAdapter(query, conn);
DataTable dt = new DataTable();
sda.Fill(dt);
```

...

The query that this code intends to execute follows: SELECT * FROM items WHERE owner = <userName> AND itemname = <itemName>; However, because the query is constructed dynamically by concatenating a constant base query string and a user input string, the query only behaves correctly if itemName does not contain a single-quote character. If an attacker with the user name wiley enters the string "name' OR 'a'='a" for itemName, then the query becomes the following: SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name' OR 'a'='a'; The addition of the OR 'a'='a' condition causes the where clause to always evaluate to true, so the query becomes logically equivalent to the much simpler query: SELECT * FROM items; This simplification of the query allows the attacker to bypass the requirement that the query only return items owned by the authenticated user; the query now returns all entries stored in the items table, regardless of their specified owner.

This example examines the effects of a different malicious value passed to the query constructed and executed in the above example. If an attacker with the user name hacker enters the string "hacker'); DELETE FROM items; --" for itemName, then the query becomes the following two queries:

SQL Example:

```
SELECT * FROM items WHERE owner = 'wiley' AND
itemname = 'name';
DELETE FROM items;
--'
```

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database. Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [19]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in

this way, the general attack could still be made effective using a trick similar to the one shown in Example 1. If an attacker enters the string "name'); DELETE FROM items; SELECT * FROM items WHERE owner = 'wiley' AND itemname = 'name'; DELETE FROM items; SELECT * FROM items WHERE 'a'='a'; One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, blacklisting is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers can: - Target fields that are not quoted - Find ways to bypass the need for certain escaped meta-characters - Use stored procedures to hide the injected meta-characters Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks. Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against many others. For example, the following PL/SQL procedure is vulnerable to the same SQL injection attack shown in the first example. procedure get_item (itm_cv IN OUT ItmCurTyp, usr in varchar2, itm in varchar2) is open itm_cv for ' SELECT * FROM items WHERE ' || 'owner = '|| usr || ' AND itemname = ' || itm || '; end get_item; Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL injection attacks.

MS SQL has a built in function that enables shell command execution. An SQL injection in such a context could be disastrous. For example, a query of the form:

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='$user_input' ORDER BY PRICE

Where $user_input is taken from the user and unfiltered.

If the user provides the string:

' exec master..xp_cmdshell 'vol' --

The query will take the following form: "

SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' exec master..xp_cmdshell 'vol' --' ORDER BY PRICE

Now, this query can be broken down into: [1] a first SQL query: SELECT ITEM,PRICE FROM PRODUCT WHERE ITEM_CATEGORY='' [2] a second SQL query, which executes a shell command: exec master..xp_cmdshell 'vol' [3] an MS SQL comment: --' ORDER BY PRICE As can be seen, the malicious input changes the semantics of the query into a query, a shell command execution and a comment.

Observed Examples

CVE-2004-0366 -

CVE-2004-0343 -

CVE-2003-0779 -

CVE-2003-0500 -

CVE-2003-0377 -

Context Notes

SQL injection has become a common issue with database-driven web sites. The flaw is easily detected, and easily exploited, and as such, any site or software package with even a minimal user base is likely to be subject to an attempted attack of this kind. Essentially, the attack is accomplished by placing a meta character into data input to then place SQL commands in the control plane, which did not exist there before. This flaw depends on the fact that SQL makes no real distinction between the control and data planes.

If successful, SQL Injection attacks can give an attacker access to backend database contents, the ability to remotely execute system commands, or in some circumstances the means to take control of the Windows server hosting the database.

Dynamically generating queries that include user input can lead to SQL injection attacks. An attacker can insert SQL commands or modifiers in the user input that can cause the query to behave in an unsafe manner.

Constructing a dynamic SQL statement with user input may allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.

	Factors: resultant to special character mismanagement, MAID, or blacklist/whitelist problems. Can be primary to authentication errors.
References	M. Howard and D. LeBlanc. "Writing Secure Code". 2nd Edition. Microsoft. 2003.
Node Relationships	ChildOf - Injection (74) ChildOf - Weaknesses in OWASP Top Ten (629) ChildOf - Weaknesses Examined by SAMATE (630) ChildOf - Weaknesses Used by NVD (635) ParentOf - SQL Injection: Hibernate (564)
Source Taxonomies	PLOVER - SQL injection 7 Pernicious Kingdoms - SQL Injection CLASP - SQL injection
Applicable Platforms	All

Path 1:

Query Name - **Blind_SQL_Injections**

Severity - **Serious**

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
```

```
212. private java.util.List getList(String array[]) //deletenodesservlet.java
      ...
      217. newList.add(array[i]);
      ...
      221. return newList;
```

```
100. public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //deletenodesservlet.java
      ...
      105. java.util.List nodeDataList = getList(request.getParameterValues
      ("nodeData"));
      ...
      130. select.append((String) nodeDataList.get(j));
      ...
      138. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 2:

Query Name - **Blind_SQL_Injections**

Severity - **Serious**

```
73. public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException //addinterfacetourlservlet.java
      74. String nodeId = request.getParameter("node");
      ...
```

```
89. intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
    ...
    437. select.append(nodeId);
    ...
    447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Path 3:

Query Name - **Blind_SQL_Injections**

Severity - **Serious**

```
1. <!-- //nodelist.jsp
    ...
    52. String column = request.getParameter( "column" );
    ...
    63. AssetModel.MatchingAsset[] assets = model.searchAssets( column, search );
```

```
220. public static MatchingAsset[] searchAssets(String columnName, String
searchText) throws SQLException //assetmodel.java
    ...
    238. PreparedStatement stmt = conn.prepareStatement("SELECT ASSETS.NODEID,
    NODE.NODELABEL, ASSETS." + columnName + " FROM ASSETS, NODE WHERE LOWER
    (ASSETS." + columnName + ") LIKE ? AND ASSETS.NODEID=NODE.NODEID ORDER BY
    NODE.NODELABEL");
    ...
    241. ResultSet rs = stmt.executeQuery();
```

[Table of Contents] [Description]

Path 4:

Query Name - **Blind_SQL_Injections**

Severity - **Serious**

```
1. <!-- //chooseinterface.jsp
    ...
    60. String nodeId = request.getParameter("node");
    ...
    71. ArrayList intfs = this.model.getQueryableInterfacesForNode(nodeId);
```

```
429. public ArrayList getQueryableInterfacesForNode(String nodeId) throws
SQLException //responsetimemodel.java
    ...
    437. select.append(nodeId);
    ...
    447. ResultSet rs = stmt.executeQuery(select.toString());
```

[Table of Contents] [Description]

Data Leaking Between Users

CWE ID	488
Description	Data can "bleed" from one session to another through member variables of singleton objects, such as Servlets, and objects from a shared pool.
Demonstrative Examples	The following Servlet stores the value of a request parameter in a member field and then later echoes the parameter value to the response output stream.

```
public class GuestBook extends HttpServlet {

  String name;

  protected void doPost (HttpServletRequest req,
  HttpServletResponse res) {
    name = req.getParameter("name");
    ...
    out.println(name + ", thanks for visiting!");
  }
}
```

While this code will work perfectly in a single-user environment, if two users access the Servlet at approximately the same time, it is possible for the two request handler threads to interleave in the following way: Thread 1: assign "Dick" to name Thread 2: assign "Jane" to name Thread 1: print "Jane, thanks for visiting!" Thread 2: print "Jane, thanks for visiting!" Thereby showing the first user the second user's name.

Context Notes	Many Servlet developers do not understand that, unless a Servlet implements the SingleThreadModel interface, the Servlet is a singleton; there is only one instance of the Servlet, and that single instance is used and re-used to handle multiple requests that are processed simultaneously by different threads. A common result of this misunderstanding is that developers use Servlet member fields in such a way that one user may inadvertently see another user's data. In other words, storing user data in Servlet member fields introduces a data access race condition.
Node Relationships	ChildOf - Encapsulation (485)
Source Taxonomies	7 Pernicious Kingdoms - Data Leaking Between Users
Applicable Platforms	All

Path 1:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
75. public class AddPollerConfigServlet extends
HttpServlet //addpollerconfigservlet.java
```

[Table of Contents] [Description]

Path 2:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
77. public class PollerConfigServlet extends HttpServlet //pollerconfigservlet.java
      |
```

[Table of Contents] [Description]

Path 3:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
77. public class DeleteNodesServlet extends HttpServlet //deletenodesservlet.java
      |
```

[Table of Contents] [Description]

Path 4:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
76. public class AddPollerConfigServlet extends
HttpServlet //addpollerconfigservlet.java
      |
```

[Table of Contents] [Description]

Path 5:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
76. public class PollerConfigServlet extends HttpServlet //pollerconfigservlet.java
      |
```

[Table of Contents] [Description]

Path 6:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
51. public class ExportAssetsServlet extends HttpServlet //exportassetsservlet.java
      |
```

[Table of Contents] [Description]

Path 7:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
63. public class ImportAssetsServlet extends HttpServlet //importassetsservlet.java
```

[Table of Contents] [Description]

Path 8:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
56. public class ModifyAssetServlet extends HttpServlet //modifyassetservlet.java
```

[Table of Contents] [Description]

Path 9:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
60. public class AvailabilityServlet extends HttpServlet //availabilityservlet.java
```

[Table of Contents] [Description]

Path 10:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
47. public class RTCDebugServlet extends HttpServlet //rtcdebugservlet.java
```

[Table of Contents] [Description]

Path 11:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
51. public class RTCPostServlet extends HttpServlet //rtcpostservlet.java
```

[Table of Contents] [Description]

Path 12:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
54. public class NodeRescanServlet extends HttpServlet //noderescanservlet.java
```

[Table of Contents] [Description]

Path 13:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
57. public class AcknowledgeEventByFilterServlet extends
HttpServlet //acknowledgeeventbyfilterservlet.java
```

[Table of Contents] [Description]

Path 14:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
55. public class AcknowledgeEventServlet extends
HttpServlet //acknowledgeeventservlet.java
```

[Table of Contents] [Description]

Path 15:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
53. public class DeleteEventServlet extends HttpServlet //deleteeventservlet.java
```

[Table of Contents] [Description]

Path 16:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
68. public class RRDAdHocGraphServlet extends
HttpServlet //rrdadhocgraphservlet.java
```

[Table of Contents] [Description]

Path 17:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
86. public class RRDGraphServlet extends HttpServlet //rrdgraphservlet.java
```

[Table of Contents] [Description]

Path 18:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
57. public class MailerServlet extends HttpServlet //mailerservlet.java
      |
```

[Table of Contents] [Description]

Path 19:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
62. public class NodeLabelChangeServlet extends
HttpServlet //nodelabelchangeservlet.java
      |
```

[Table of Contents] [Description]

Path 20:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
55. public class AcknowledgeNoticeServlet extends
HttpServlet //acknowledgenoticeservlet.java
      |
```

[Table of Contents] [Description]

Path 21:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
48. public class AddInterfaceToURLServlet extends
HttpServlet //addinterfacetourlservlet.java
      |
```

[Table of Contents] [Description]

Path 22:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
49. public class AddReportsToUrlServlet extends
HttpServlet //addreportstourlservlet.java
      |
```

[Table of Contents] [Description]

Path 23:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
64. public class RRDDumpServlet extends HttpServlet //rrddumpservlet.java
     |
```

[Table of Contents] [Description]

Path 24:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
51. public class RRDListServlet extends HttpServlet //rrdlistservlet.java
     |
```

[Table of Contents] [Description]

Path 25:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
53. public class AddInterfaceToURLServlet extends
HttpServlet //addinterfacetourlservlet.java
     |
```

[Table of Contents] [Description]

Path 26:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
53. public class AddReportsToUrlServlet extends
HttpServlet //addreportstourlservlet.java
     |
```

[Table of Contents] [Description]

Path 27:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
68. public class RRDDumpServlet extends HttpServlet //rrddumpservlet.java
     |
```

[Table of Contents] [Description]

Path 28:

Query Name - **Singleton_HTTPServlet**

Severity - **Serious**

```
55. public class RRDListServlet extends HttpServlet //rrdlistservlet.java
     |
```

[Table of Contents] [Description]

Copyright © 2008 Checkmarx
Description materials copyrighted by the National Cyber Security Division of the U.S. Department of Homeland Security and the MITRE Corporation.