A Logic Based Model For Error Management in Network Forensics Analysis

Anoop Singhal, C. Liu and Duminda Wijesekara
anoop.singhal@nist.gov, cliu6@gmu.edu, dwijesek@gmu.edu

National Institute of Standards and Technology, Gaithersburg MD 20899 USA
Department of Computer Science, George Mason University, Fairfax VA 22030 USA

Outline

- Objectives
- The architecture of the logic based model
- An example network
- Implementation
  - Using rules to correlate evidence for attack scenario reconstruction
  - Using expert knowledge and anti-forensic databases for hypothesis testing
- Experimental result
  - using the implementation on the example network
- Future work

Objectives

- Develop a formal model and a software tool to reason with digital evidence in the presence of errors for forensic purposes.
  - Reconstruct attack scenarios by using evidence including IDS alerts and system logs
  - Provide the explanation when evidence is missing or destroyed

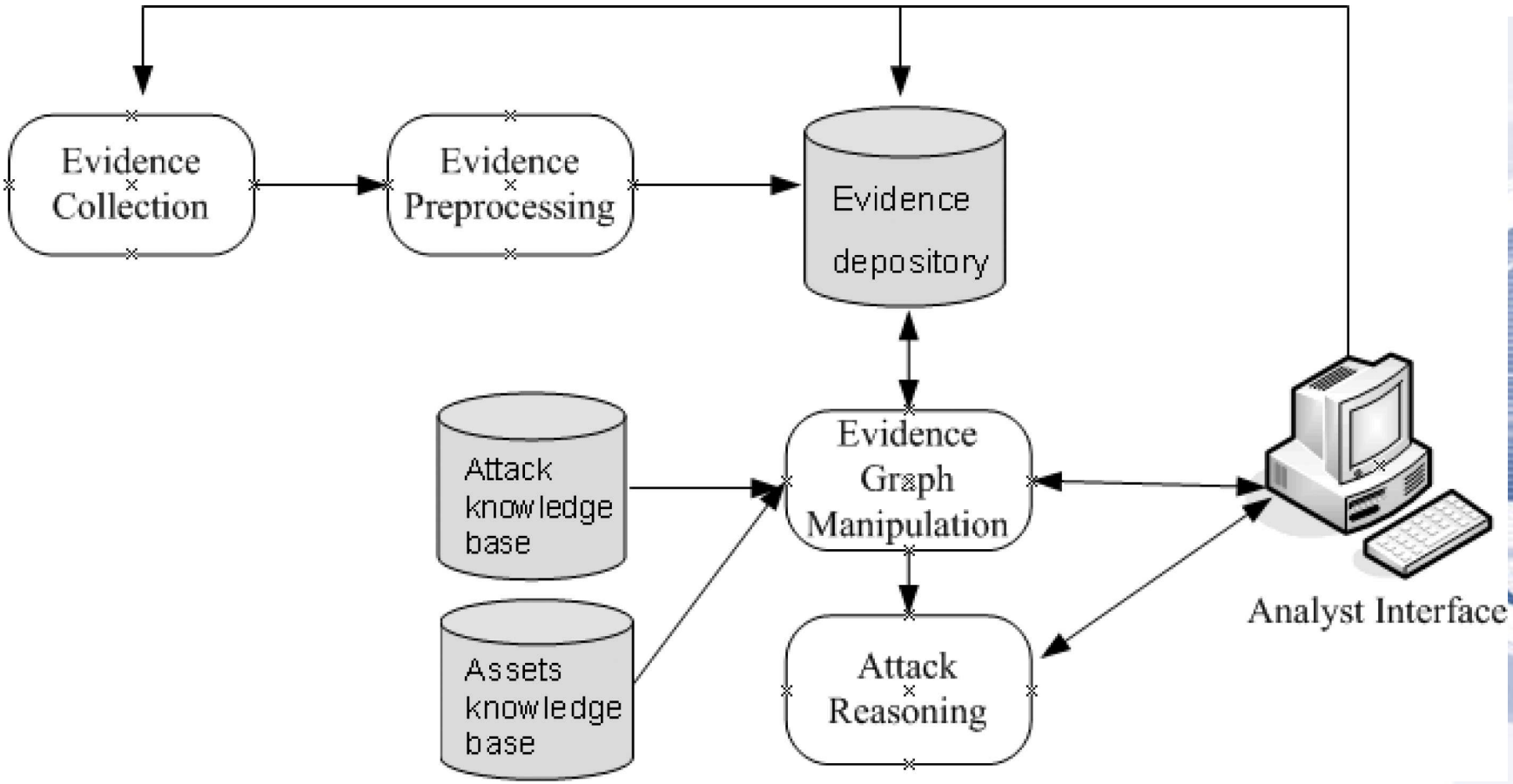
Sources of Error

- Evidence can be a false positive as IDS did not generate the right alert
- Security Events can be very large
- Evidence can be deleted
- Some evidence can be missing due to storage limitation

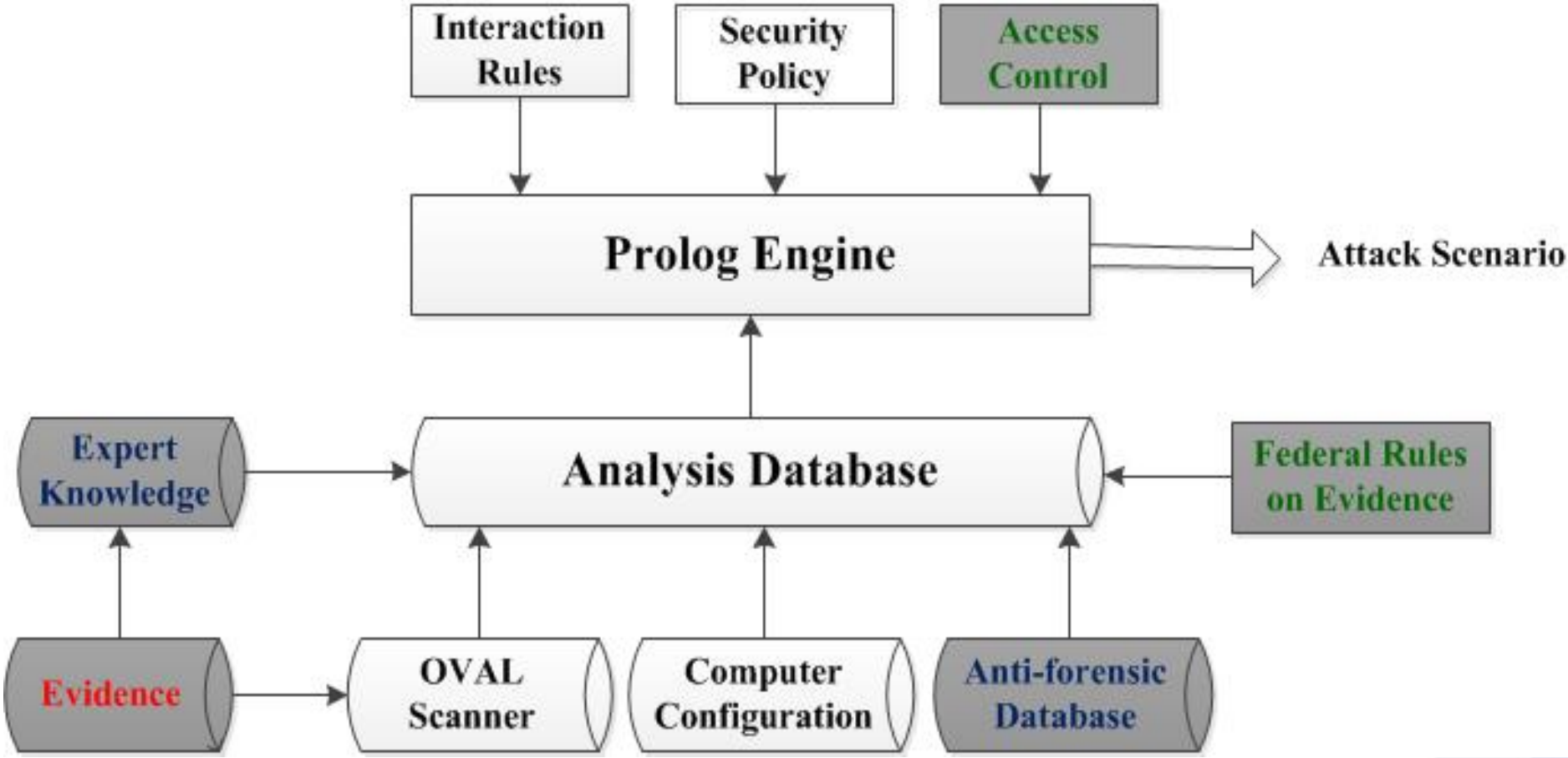
Methods to Mitigate Errors

- Map Evidences to existing vulnerabilities
- Use an anti-forensic database to detect deletion of evidences
- Use primary, secondary and tertiary storage methods to continuously back up the events

Overview of Architecture



The Architecture of Logic Based Model

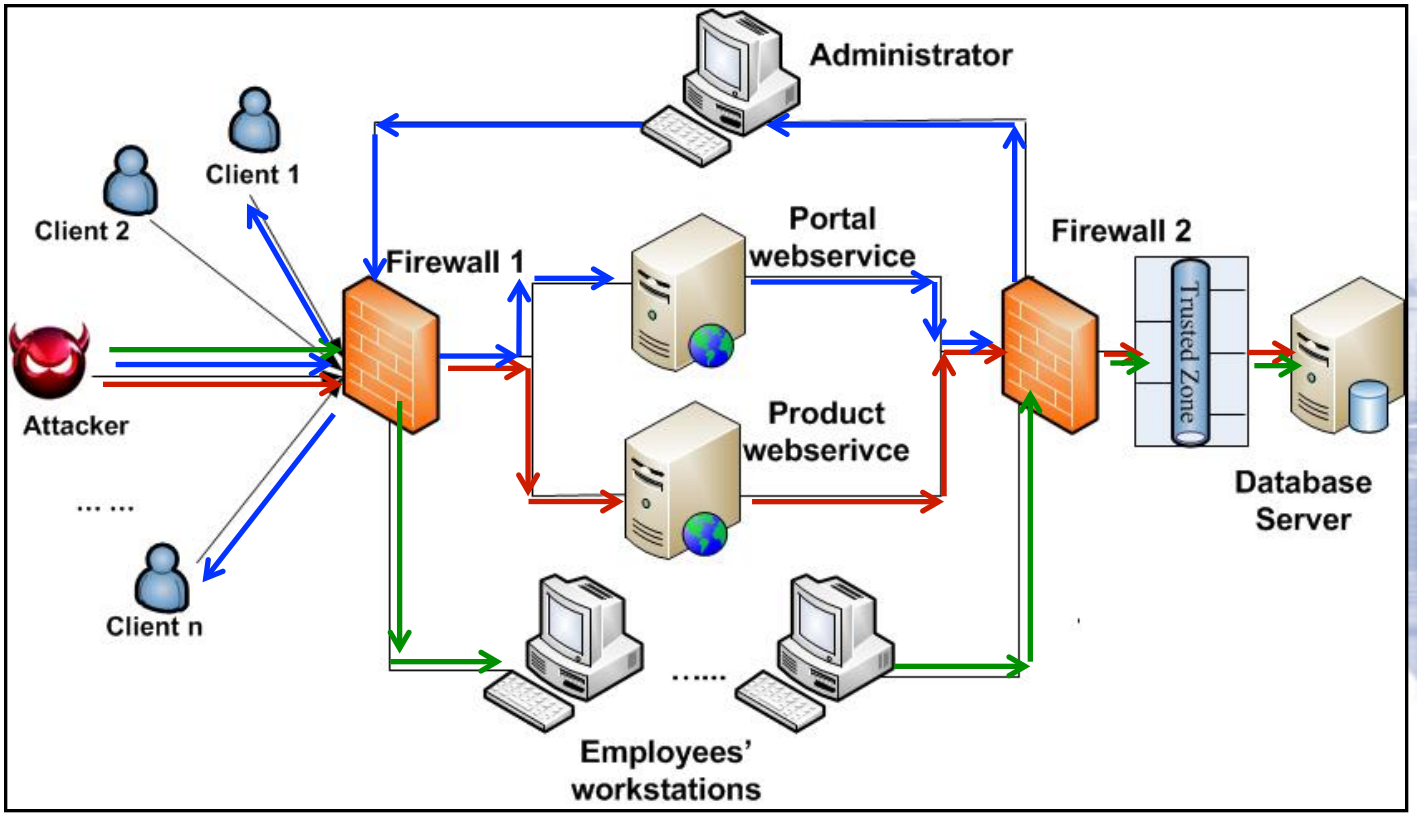


The Implementation of Logic Based Model

- The model extends MulVAL, a Prolog based tool that generates attack graphs by reasoning facts including computer configuration, network topology and vulnerability information

- Extensions
  - **ANTI-FORENSIC** and **EXPERT KNOWLEDGE** databases used to generate explanations for the missing or destroyed evidence

An Example Network



Red path:
SQL injection attack(CWE-89)

Green path:
Compromise the workstation by IE to access the database server (CVE-2009-1918)

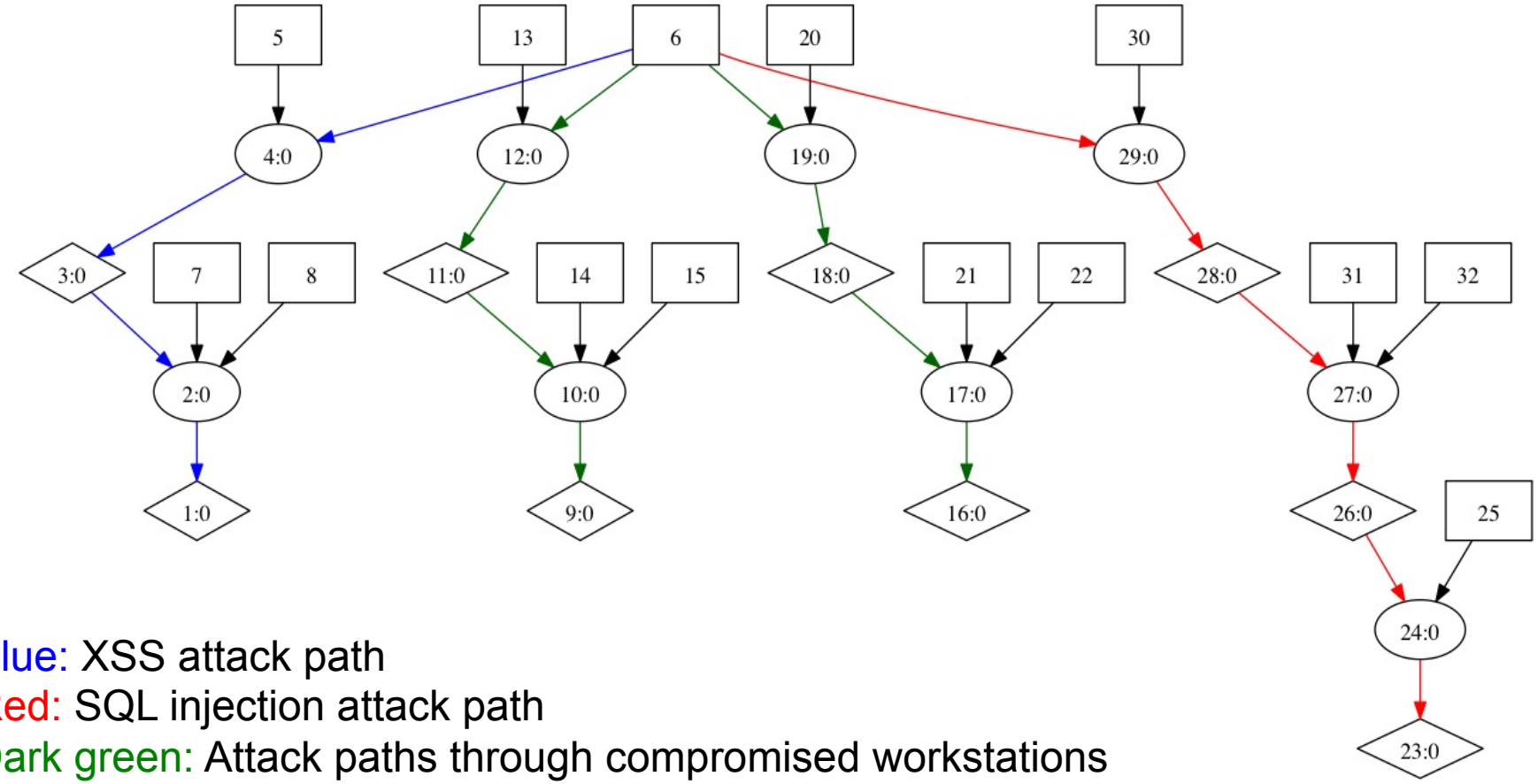
Blue path:
Compromise admin session id by XSS attack

IDS(Snort) deployed
Servers configured for access and query logging

Example Vulnerabilities

Machine	IP Address: Port	Vulnerability
Attacker	129.174.124.122	
Workstations	129.174.124.184/185/186	HTML Objects Memory Corruption Vulnerability (CVE-2009-1918)
Webserver 1— Product web service	129.174.124.53:8080	SQL Injection (CWE89)
Webserver 2— Portal web service	129.174.124.53:80	SQL Injection (CWE89)
Administrator	129.174.124.137	Cross Site Scripting Flaw (XSS)
Database server	129.174.124.35	

Construct Attack Scenario from Evidence



Blue: XSS attack path
Red: SQL injection attack path
Dark green: Attack paths through compromised workstations

Question: *is this constructed attack scenario(so-called evidence graph) complete and validated?*

“What if” Scenarios

- Network forensics analysis is not complete without hypothesis testing.
  - hypothesis is a “what if” proposition made for possible explanation of missing information.
- Examples
  - What if the buffer overflow alert from *attacker* to *a workstation* is a false positive?
  - What if the attacker used the compromised workstation to login into the database?
- Questions on Hypothesis
  - How do we generate relevant ones?
  - How do we choose the best one?

12

Solution 1: Use Attack Graph or Expert Knowledge

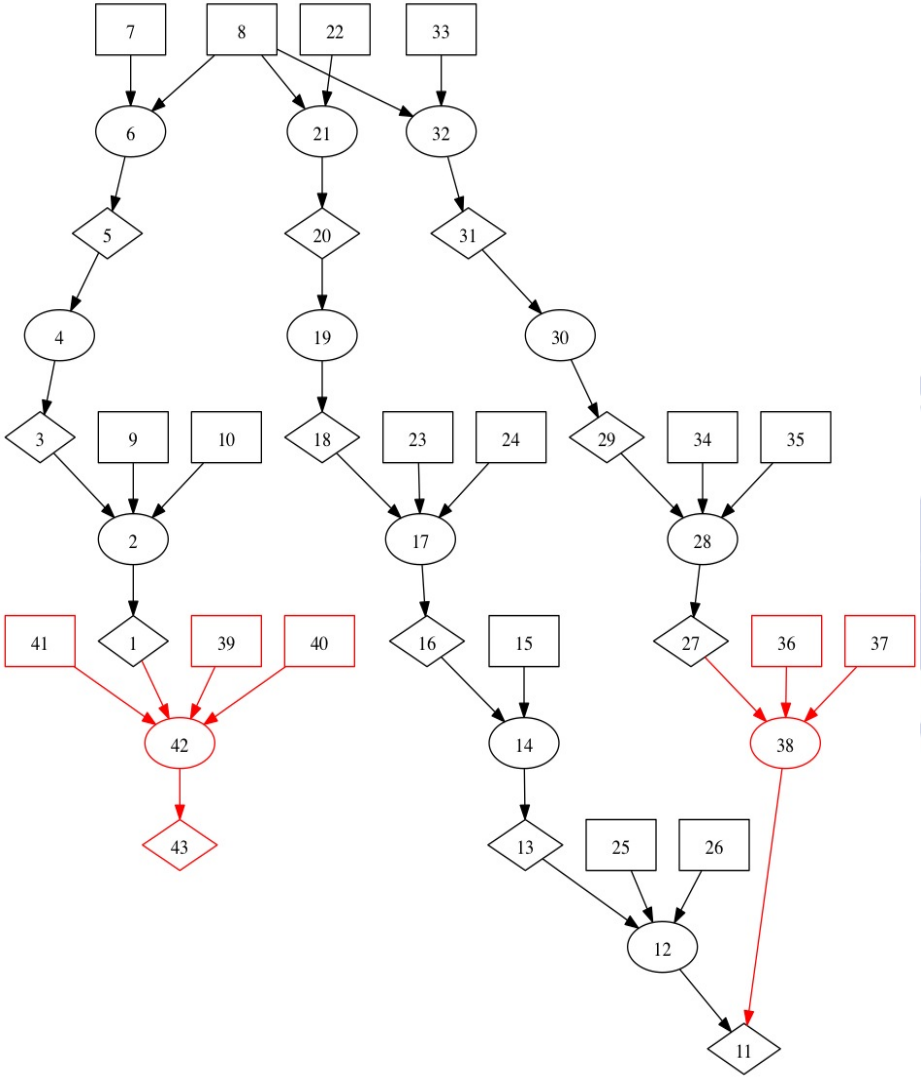
- The hypothesis are additions to the attack graph or the expert knowledge
  - The attack graph is constructed by using vulnerability information from the National Vulnerability Database

Solution 2: Use an Anti-forensic Database

- Attackers might have used anti-forensic technique to destroy evidence

ID	Category	Tool	Technique	Windows	Linux	Privilege	Access	Software	Effect
A1	Attack tool		Obfuscate signature	All	All	User	remoteClient	SNORT Rule	Bypass being detected by rules
D1	Destroy data	BCWipe	Delete file content	98 Above	All	User	localClient		Delete data permanently
D2	Destroy data		Remove log file	All	All	User	remoteClient	MySql 5.0 above set log off command	Set general log off
..	…								

Evidence Graph with Our Extended Modules



- The left path is XSS attack
- The middle path is SQL injection Attack
- The right path is the attack to the database by a compromised Windows computer

Notice:

- One attack path through compromised computer has been removed, because Linux system does not support CVE-2009-1918
- Nodes and paths in red are added by using hypothesis.

Future Work

- Different explanations on the same evidence
  - Using Bayesian Network to assist finding best explanation
    - Add probabilities
- Use some real attack data to test the system