April 28, 2009

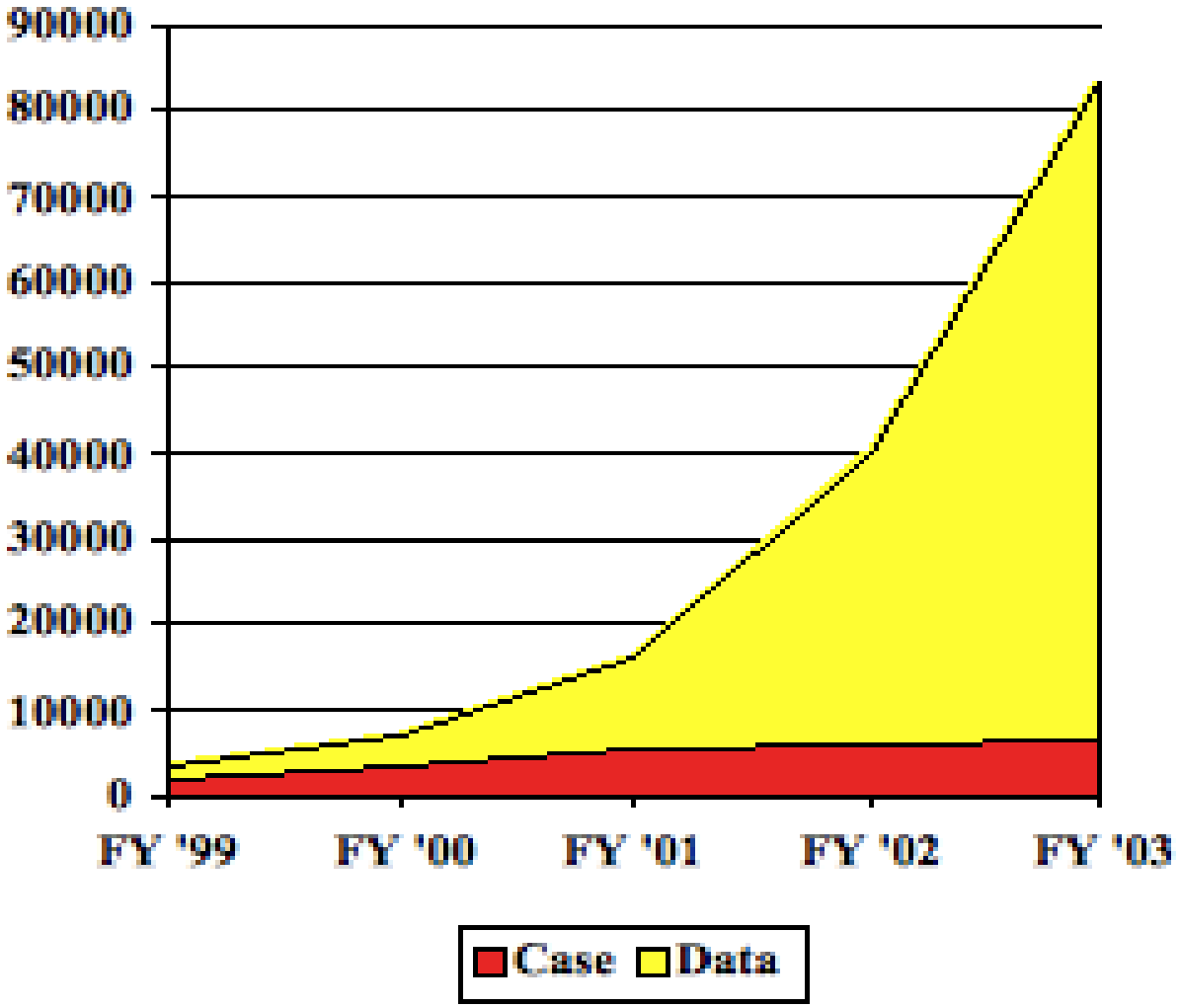
NIST United States Department of Commerce
National Institute of Standards and Technology

Disclaimer

Trade names and company products are mentioned in the text or identified. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose.

Project Partners

The NSRL project is supported by the U.S. Department of Justice's National Institute of Justice, federal, state, and local law enforcement, and NIST. Other federal agencies and industry organizations provide resources.



FBI's Cyber Caseload and Dataset Size Growth

Source: FBI CART, Oct 2003

NSRL History

NSRL evolved from the FBI KFF

2001 : 233,281 hashes (400,000 files)

460 products

primarily Windows software

2009 : 15,000,000 hashes (75,000,000 files)

10,000 products

Various OS, languages

Contains malware

National Software Reference Library & Reference Data Set

The NSRL is conceptually three objects:

- A physical collection of software
- A database of meta-information
- A subset of the database,

  the Reference Data Set

The NSRL is designed to collect software from various sources and incorporate file profiles computed from this software into a Reference Data Set of information.

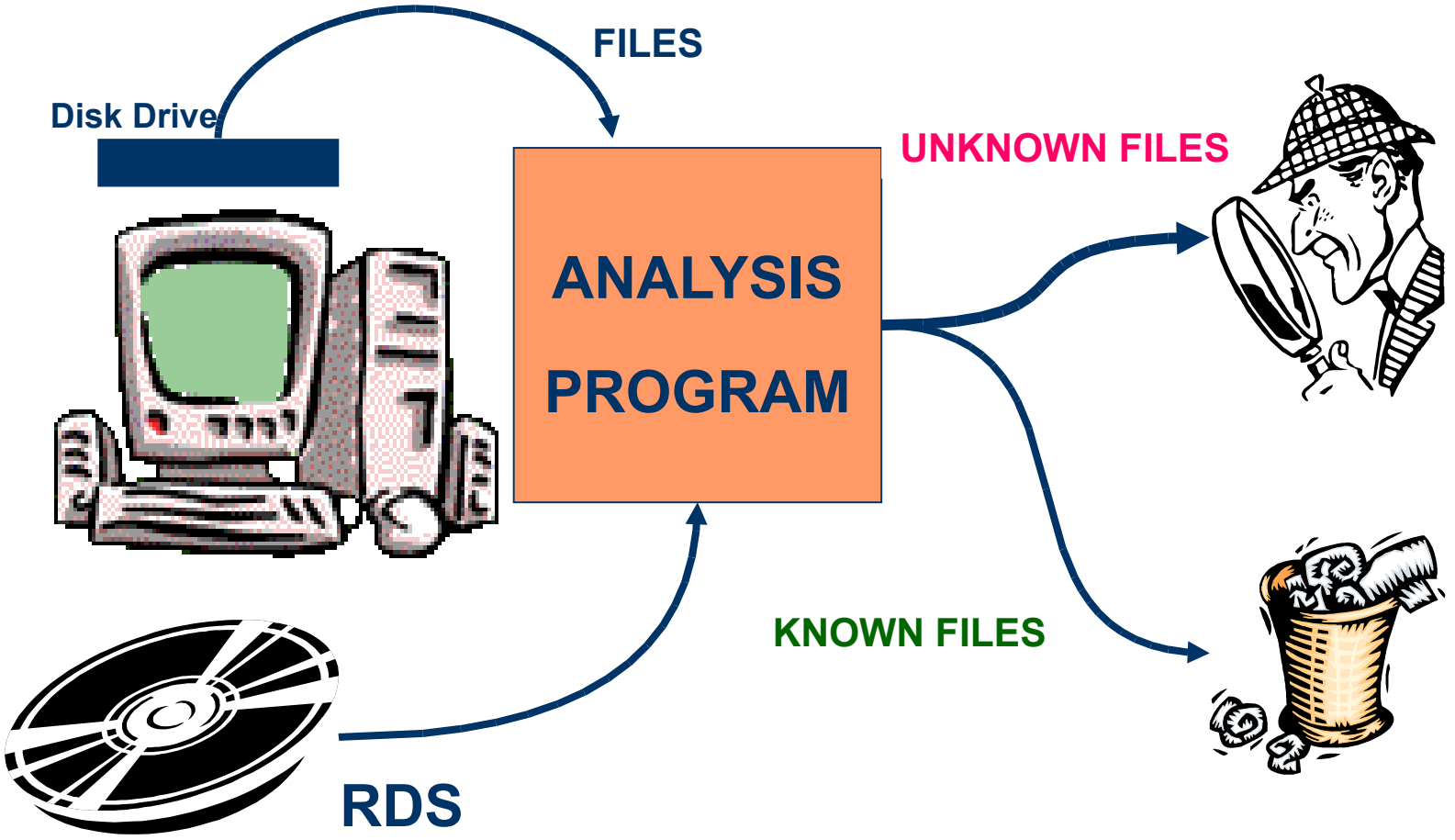
Use in Commercial Products

Typically the RDS data is imported into one of many forensics software tools.

NIST works with vendors to provide RDS import functionality.

NIST works with investigators to address evolving threats and trends.

NIST provides support relating to RDS content to users.


FILES
Disk Drive
ANALYSIS
PROGRAM
UNKNOWN FILES
KNOWN FILES
RDS

NSRL Impact

Referenced in 2001 seizure of bogus MS media in CA.

Referenced by Simpson Garfinkel in 2002 efforts with reclaimed disks.

Imported into EnCase, FTK, Ilook, Hashkeeper, Maresware.

Essential to FBI CART, copied for every field office.

Used by private organizations to eradicate P2P use.

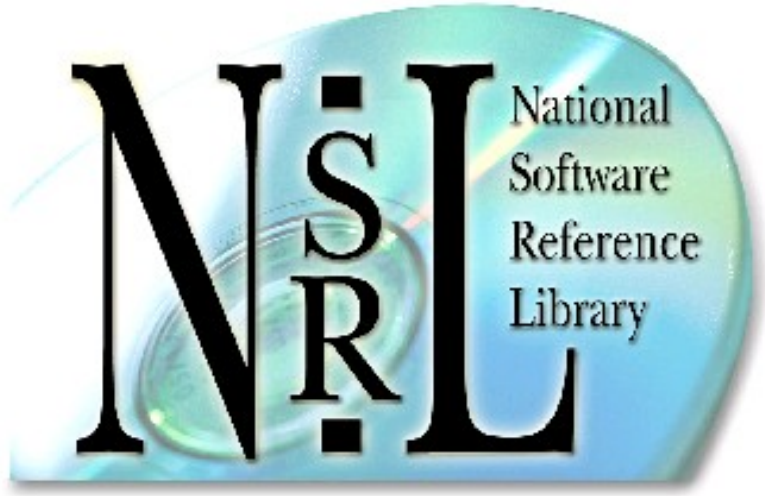
Used by ISPs to track app sharing on servers.

Used by sysadmins to confirm valid OS file state.

Used by FDA in FL Botox case.

International use - UK NHTCU, EU JRC, etc.

NIST Special Database #28



Reference Data Set

Version 1.5 03/03/2003

NIST

Original Metadata Intent

The project sponsors were initially concerned with identification of known application files, to allow known files to be ignored, focusing investigation on user-generated data.

NIST does not assign “malicious” nor “notable” values to applications.

Evolving Metadata Intent

The NSRL does assign application categories, e.g. image manipulation, steganography, encryption. Original directory/ path location is noted.

The NSRL metadata has been used to determine the "pedigree" of NARA systems. Can determine the upgrade path of a PC such as from NT3.5 to NT4 to W2K.

Other requested data are original MAC date/time, alternate data streams, byte signature info (Unix "magic")

NDIC Hashkeeper

- DoJ's National Drug Intelligence Center (NDIC) HashKeeper project produces hashsets
- Based on seized data and original media
- http://groups.yahoo.com/group/hashkeeper
- Three main FTP sites
- Over 300 hash sets

Other Hashset Sources

- Maresware
- Tripwire FSDB
- SARC Steganography set
- Hashkeeper, CFTT, iLook, CFID email lists
- Professional connections

Processing of Media

NSRL is using dcfldd to create a library of media images.

The media images can be processed by many algorithms automatically.

While the images are not available publicly, the metadata generated by algorithms is available.

Hashing Environment

The hashing of media is performed on various operating systems.

Both physical and virtual machines are used to generate metadata.

Any data processing algorithm can be run against the media collection.

Hash Collision News

- **The NSRL project does not see any fatal ramifications from the collision announcements.**
- Details posted at http://www.nsrl.nist.gov/collision.html
- We have not seen a "pre-image" attack; that is, the researchers did not identify a known file in the NSRL and attempt to generate a different file with a matching hash value.
- There are known MD5 collisions and weaknesses; the NSRL data provides an MD5 to SHA-1 mapping to facilitate the migration away from MD5.
- SHA-1 will be superceded in 2010 by FIPS 180-2, Secure Hash Standard (SHA-224, 256, 384,512). The NSRL will provide a SHA-1 to SHA-256 mapping.
- The NSRL provides several hash values and the file size, and it is highly improbable that a pre-image attack will be found soon that can generate a combination of hash collisions.

Foreign Languages

- Applications available in various languages yield hash sets with a 50% collision rate.
- Analyzing text and HTML tags can increase the collision rate to 99%.
- Metadata from English software assists foreign language investigations (and vice versa)

Block Hashes

We are collecting the hashes of 4,096 Byte blocks in application files.

If files change on installation, many blocks remain the same.

Statistics can be used to identify matching files.

Screening can occur on raw disks or VM images.

Identification of Issues

- Hash of file contents is easily changed without perturbing contents
- Amount of data input to investigation is immense
- Commonly used hash algorithms can not identify suspect files similar to known files
- Commonly used hash algorithms do not yield useful data on partial or deleted files

Perturbing File Hashes

Use of cryptographic hashes to automatically identify files is absolute when applied to a file as a whole; it is unambiguously categorized.

When dealing with morphing digital objects, such sorting leaves many files to be dealt with by manual review.

The NSRL hashset is commonly used to automatically remove benign known items from human processing, which is fail-safe.

Reducing Data Inflow

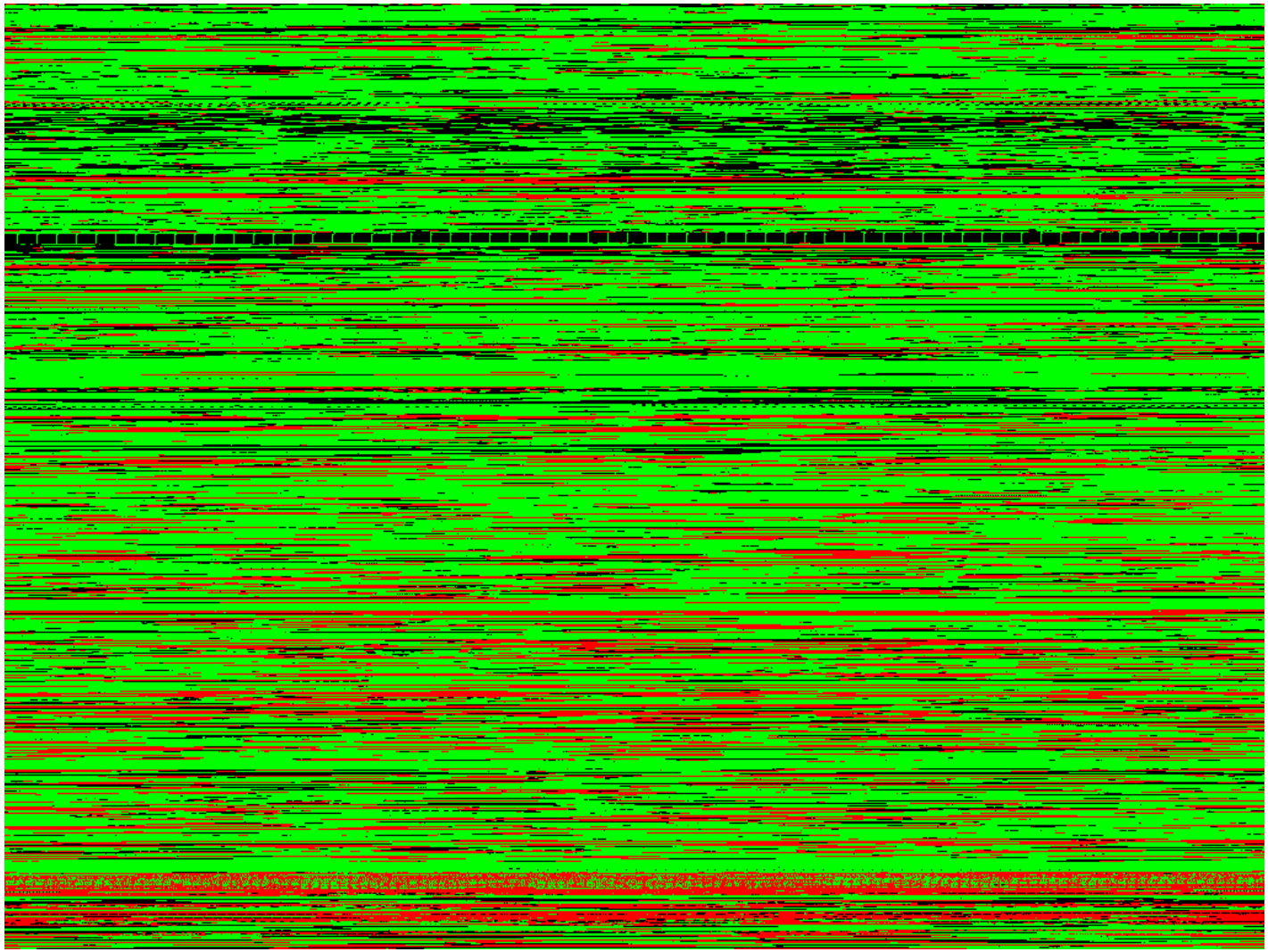
NSRL file content hash values allow investigators to automatically remove benign known items from view.

Known benign data can be identified before it arrives to investigators.

Is it technically possible to meaningfully reduce the amount of incoming data?

Known - Unknown - Zero

2nd 512 MB in W2K NTFS VM



Ssdeep Signatures

Also called “Fuzzy” hashing.

We are collecting the signatures of files as computed by the Ssdeep tool.

The signatures can be compared using the Levenshtein difference to calculate similarity.

This is like block hashing, but accommodates a greater number of dynamic file changes.

Bloom Filters

The space requirements for distributing hashes for each 4KB is enormous.

A Bloom filter 512MB in size can contain 100 million MD5 hashes.

A Bloom filter 4GB in size can contain 1 billion MD5 hashes.

Query speed is optimized.

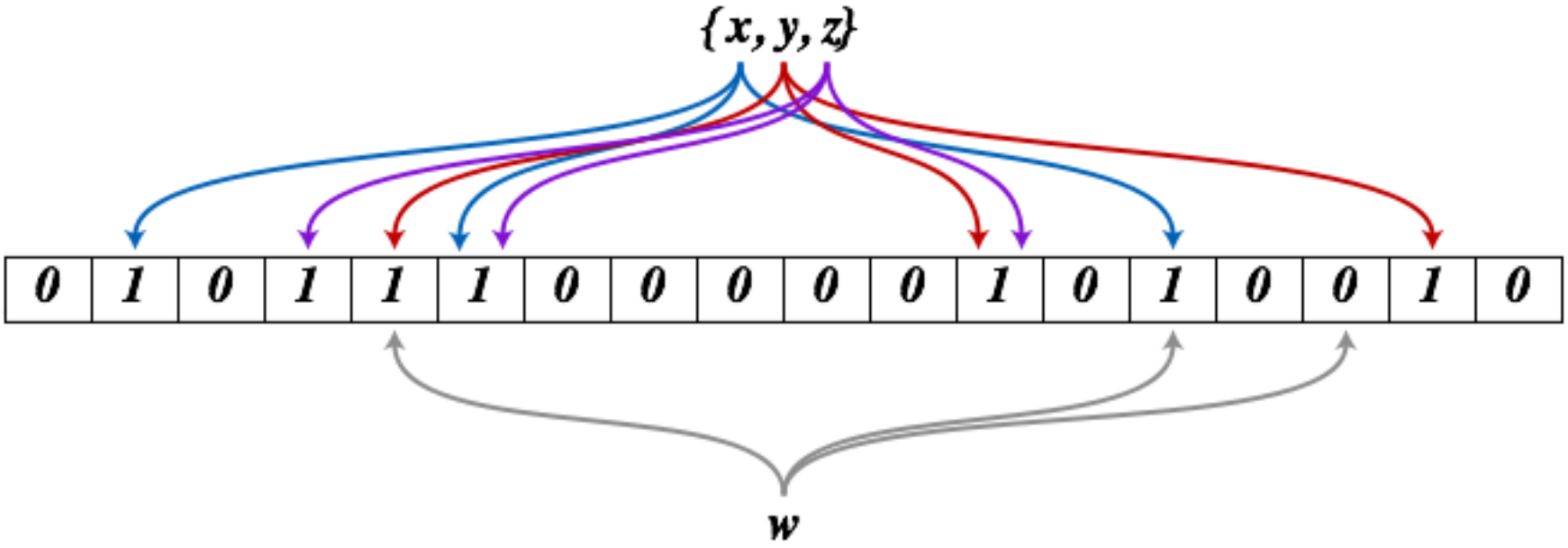
Issues Identified

- Storage and distribution of tens of millions of hash values
- Storage and distribution of hundreds of millions of block hash values
- Speed of testing acquired hash values
- Interagency awareness without information release

Bloom Filter

A Bloom filter is a space-efficient probabilistic data structure that is used to test whether an element is a member of a set. False positives are possible, but false negatives are not. Elements can be added to the set, but not removed.

Most implementations are dynamic, growing as data is added to a back-end storage system.



Items x, y, and z have been added to the Bloom filter.

A search for item w yields a negative result.

Storage Space

Bloom filters have an advantage over other data structures for representing sets such as self-balancing binary search trees, tries, or hash tables which require storing at least the data items themselves.

A Bloom filter with 1% error and an optimal value of k, on the other hand, requires only about 9.6 bits per element regardless of element size.

The false positive rate can be reduced by a factor of ten each time 4.8 bits per element are added.

Storage Space

NSRL investigated values of m = 2^32 and n = 10^8, which equates to a 512MiB bit array containing 100,000,000 items.

A value of k = 16 allows a false positive rate of 0.000001%.

This compares favorably to 1.6GB needed for 100,000,000 MD5 hashes.

Storage Space

NSRL investigated values of m = 2^35 and n = 10^9, which equates to a 4GiB bit array containing 1 billion items.

A value of k = 16 allows a false positive rate of 0.000014%.

This compares favorably to 16GB needed for 1 billion MD5 hashes.

Speed of Access

Bloom filters have the unusual property that the time needed to either add items or to check whether an item is in the set is a fixed constant, O(k), completely independent of the number of items already in the set.

No other constant-space set data structure has this property.

In a hardware implementation, however, the Bloom filter shines because its k lookups are independent and can be parallelized.

Measurements

Experiments focused on a 512MiB filter, as math could be performed with 32 bit integers and 512MiB was easily held in RAM.

Using a 2GHz intel Core 2 Duo, 10 million MD5 values can be added to a 512MiB filter in less than 10 seconds.

Average query speed is on the order of 15,000 results per second.

Query speed increases as the ratio of unknown items increases.

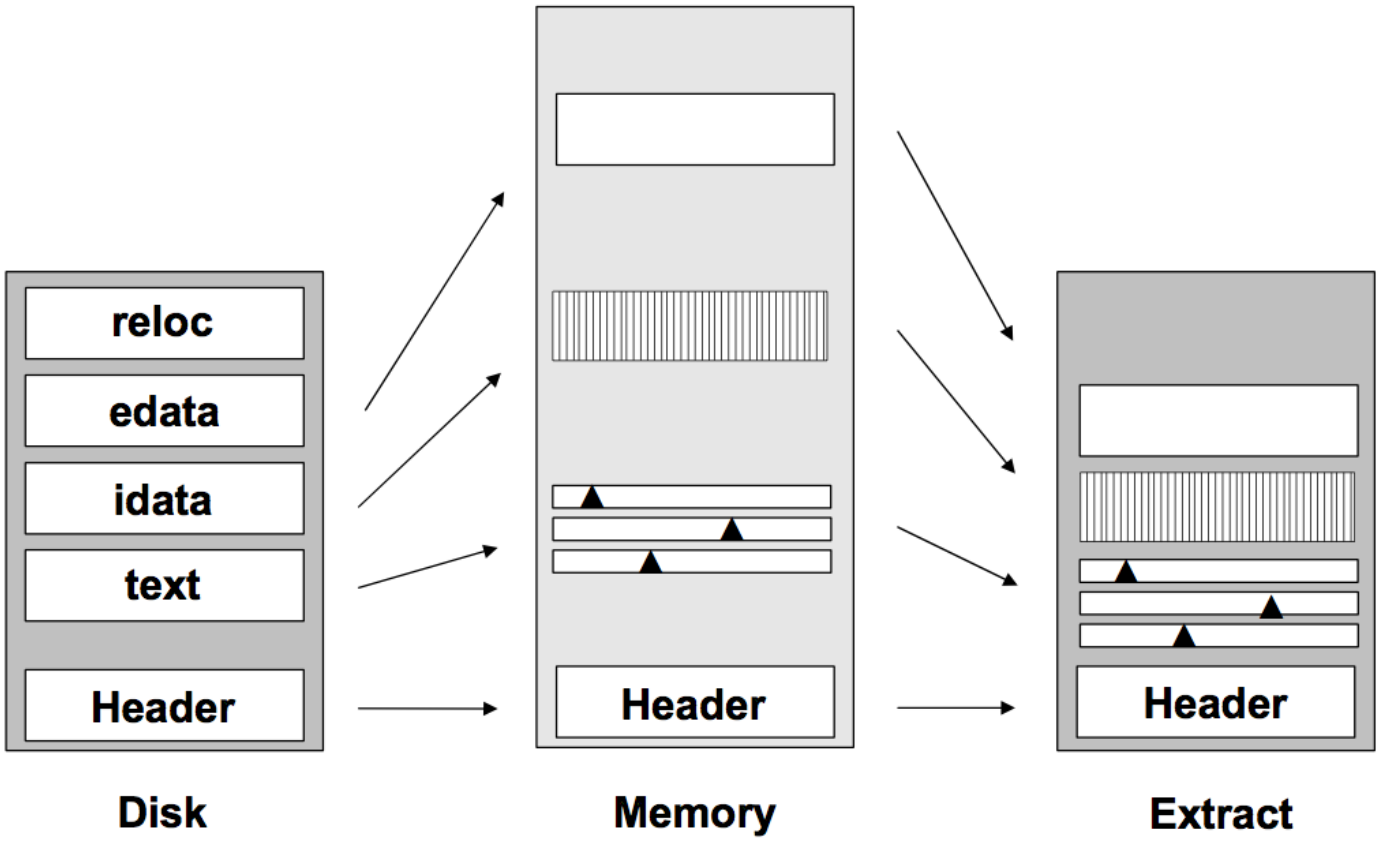
Volatile Memory Hashes

We are collecting the hashes of executable pages from Windows files.

These hashes can identify programs in RAM.

The hashes are also piecewise, so modified executables can be identified.

Volatile Memory



Windows Registry Dataset

We have prototype tools available which track windows registry changes.

Installation, execution and removal of applications can be documented.

The data is based on an FBI database schema.

www.nsrl.nist.gov/WIRED/WIRED-060511.iso

Windows Registry Data Set

It is possible to compile a historical list of applications that have been installed and removed from a computer system based on RDS metadata, but there must be residue files.

There are many methods that can be used to remove all application files. In many cases these methods do not completely purge the Registry of associated information.

Examining the Registry for residue can augment a historical list of applications or provide additional context about system use.

Windows Registry Data Set

The Windows Registry Dataset (WiReD) contains the changes to the Windows Registry caused by application installation, de-installation, execution or other Registry modifying operations.

The applications are chosen from the NSRL collection, to be of interest to computer forensic examiners.

WiReD is currently an experimental prototype.

NIST is soliciting feedback from the computer forensics community to improve and extend its usefulness.

WIRED

The WiReD dataset currently has the following fields:

- CHANGE_TYPE - if the Registry entry was added, deleted or modified
- APP_NAME - the application's name
- NSRL_APP_ID - if the application is part of the NSRL, its ID
- ACTION - whether the application was installed, deinstalled, executed or some other type of registry modification occurred
- ENTRY_TYPE - is the Registry entry a key or value?
- PATH - the Registry entry's path
- VALUE_NAME - if the entry is a value, its name
- VALUE_DATA - if the entry is a value, the data it contains

Continuing Efforts

The task and prioritization of identifying and acquiring software is difficult. The needs of sponsoring agencies drive accession activities.

The task of processing software for inclusion in the dataset currently is person-intensive. Automation and virtualization are goals, but these require large investments.

The current prototype is seen as a step in a much larger scheme that includes an XML database for managing the Registry difference files. This will allow for the efficient query and manipulation of acquired Registry data.

Expansion of Registry modification detection to beyond just application installation to include all phases of an application's life cycle on a given machine is the long term forensic information we seek.

National Vulnerability Database & Common Platform Enumeration

The NVD is the U.S. government repository of standards based vulnerability management data .This data enables automation of vulnerability management, security measurement, and compliance.

The NVD includes databases of security checklists, security related software flaws, misconfigurations, product names, and impact metrics.

The NVD contains content (and pointers to tools) for performing configuration checking of systems implementing the FDCC.

http://nvd.nist.gov/

The official CPE Dictionary is hosted by NIST as part of the NVD. This dictionary is maintained by NIST to conform to the CPE Specification and be the source for all known CPE Names.

It includes contributions from the community and is supported by many software vendors that have helped validate the names of their products.

CPE is a structured naming scheme for information technology systems, platforms, and packages. CPE includes a formal name format, a language for describing complex platforms, a method for checking names against a system, and a description format for binding text and tests to a name.

http://cpe.mitre.org/

Smart Unpacking

We are constantly upgrading how the hashing processors handle file types.

Archive files are our main target; we strive to get more metadata from media we currently own.

In tandem with block hash imaging, smart unpacking may help with data reduction.

Further Research Areas

Download-only applications

Offsite collections

Manufacturer resources

Non-NIST Physical Collections

There are collections off-site that NIST could specify as "verifiable" - National Archives, Library of Congress, National Library of France, Dutch Forensic Institute, IRS.

NIST can be a clearinghouse for unverifiable "hashes of interest" based on files that cannot be traced to original media, e.g. website downloads, one-off CDs.

Investigators can choose level of rigor needed - court admissible, peer reviewed, etc.

Downloads - legal issue for RDS, technically possible to collect.

Contacts

Doug White
www.nsrl.nist.gov
nsrl@nist.gov

Barbara Guttman
barbara.guttman@nist.gov

Sue Ballou, Office of Law Enforcement Standards
Rep. For State/Local Law Enforcement
susan.ballou@nist.gov

April 28, 2009

NIST United States Department of Commerce
National Institute of Standards and Technology