Computer Forensics: Tool Testing & National Software Reference Library

Jim Lyle and Doug White
Information Technology Laboratory
24 April 2003
Outline

- Overview of computer forensics at NIST
- Description of CFTT and NSRL projects
- Questions and answers
A Shocking Revelation . . .

Computers can be involved in crime . . .

- As a victim
- As a weapon
- As a witness
- As a record
- As contraband
Outline of an Investigation

- Get proper authorization
- Seize evidence (Hard drives, floppies …)
- Create duplicates for analysis
- Analyze the duplicates
  - Exclude known benign files
  - Examine obvious files
  - Search for hidden evidence
- Report results
Investigators Need …

Computer forensic investigators need tools that …

- Work as they should and
- Produce results admissible in court
- Reference data to reduce analysis workload
Goals of CF at NIST

- Establish methodology for testing computer forensic tools (CFTT)
- Provide international standard reference data that tool makers and investigators can use in an investigations (NSRL)
Computer Forensics Partners

NIST (OLES)

DoJ (NIJ, FBI)

DoD (DCCC)

TREASURY (USCS, USSS)

National State/Local Agencies

Homeland Security
Why NIST/ITL is involved

- Mission: Assist federal, state & local agencies
- NIST is a neutral organization – not law enforcement or vendor
- NIST provides an open, rigorous process
Computer Forensics in ITL

Located in Software Diagnostics and Conformance Testing (SDCT) Division

- Includes development of specifications and conformance tests for use by agencies and industry
- Work is funded by Federal agencies and NIST internal funds

Homeland Security support of agencies investigating terrorist activities
Computer Forensics Tool Testing (CFTT)
A Problem for Investigators

Do forensic tools work as they should?

- Software tools must be …
  - Tested: accurate, reliable & repeatable
  - Peer reviewed
  - Generally accepted

- … by whom?

- Results of a forensic analysis must be admissible in court
CFTT Presentation Overview

- Project Tasks
- Current activities
- Challenges
- Testing Hard Drive Imaging Tools
- Benefits of CFTT
Project Tasks

- Identify forensics functions e.g.,
  - disk imaging,
  - hard drive write protect,
  - deleted file recovery
- Develop specification for each category
- Peer review of specification
- Test methodology for each function
- Report results
Current Activities

- Hard drive imaging tools
- Software hard drive write protect
- Hardware hard drive write protect
- Deleted file recovery
Challenges

- No standards or specifications for tools
- Forensic vocabulary incomplete
- Arcane knowledge domain (e.g. DOS)
- Reliably faulty hardware
Hard Drive Imaging

- SCSI vs IDE
- Drive access
- Clone vs image
- Excess sectors on dst
- I/O errors
- Corrupt image file
Testing Hard Disk Drive Imaging Tools

Need to verify...

- Source disk not changed
- Copied information is accurate
- Behavior if source is smaller than destination
- Behavior if source is larger than destination
Testing Hard Disk Drive Imaging Tools

Testing support Tools

- Detect change
- Compare Source to Destination
- Track relocated information

<table>
<thead>
<tr>
<th>ASCII String</th>
<th>25 bytes</th>
<th>Fill Bytes 487 Bytes</th>
</tr>
</thead>
</table>

4/24/2003
Testing Hard Disk Drive Imaging Tools

Setup Source
Wipe
Load OS
Hash
Testing Hard Disk Drive Imaging Tools

- Select Source
- Wipe Destination
- Run Tool
- Compare Src : Dst
- Hash Source
Impact

- Release 18 (Feb 2001) - A US government organization was doing some testing and uncovered an issue under a specific set of circumstances.
- Linux doesn’t use the last sector if odd
- Several vendors have made product or documentation changes
Benefits of CFTT

Benefits of a forensic tool testing program

- Users can make informed choices
- Neutral test program (not law enforcement)
- Reduce challenges to admissibility of digital evidence
- Tool creators make better tools
Lab Facilities
Introduction

The National Software Reference Library is:
- A physical collection of over 3,800 software packages on secured shelves
- A database of file “fingerprints” and additional information to uniquely identify each file on the shelves
- A Reference Data Set (RDS) extracted from the database onto CD, used by law enforcement, investigators and researchers
Addressing Industry Needs

- No unbiased organizations were involved in implementing investigative tools
- Law enforcement had no control over quality of data provided by available tools – data was market-driven
- Traceability - No repositories of original software available for reproducing data
- Each tool provided a limited set of capabilities
NSRL Software Collection

- Media in format as available to the public
- Consumer products available in stores
- Developer products available as vendor services
- Malicious software
- “Cracked” software
NSRL Software Collection

- Balance of most popular (encountered often) and most desired (pirated often)
  - Currently 32 languages
- Software is purchased commercially
- Software is donated under non-use policy
- List of contents available on website
  www.nsrl.nist.gov
NSRL Software Database

- Information to uniquely identify every file on every piece of media in every application
- Database schema is available on website
- 4,200 Bytes per application
- 750 Bytes per file
- Total database size now 9 GB for 3,800 applications with 13,400,000 files
NIST Special Database #28

Reference Data Set
Version 1.5 03/03/2003
NSRL Reference Data Set

- The Reference Data Set (RDS) is a selection of information from the NSRL database.
- Allows positive identification of manufacturer, product, operating system, version, file name from file “signature”.
- Data format available for forensic tool developers.
- Published quarterly.
Use of the RDS

- Eliminate as many known files as possible from the examination process using automated means
- Discover expected file name with unknown contents
- Identify origins of files
- Look for malicious files, e.g., hacker tools
- Provide rigorously verified data for forensic investigations
You are looking for facility maps on a computer which is running Windows 2000.

Windows 2000 operating system software contains 5933 images which are known gifs, icons, jpeg files.

By using the RDS and an analysis program the investigator would not have to look at these files to complete her investigation.
RDS Field Use Concept

KNOWN FILES

RDS ANALYSIS PROGRAM

UNKNOWN FILES

Disk Drive

RDS

KNOWN FILES
Haunted By Ghosts Of Hard Drives Past


(AP) So, you think you cleaned all your personal files from that old computer you got rid of?

Two MIT graduate students suggest you think again.

Over two years, Simson Garfinkel and Abhi Shelat bought 158 used hard drives at secondhand computer stores and on eBay. Of the 129 drives that functioned, 69 still had recoverable files on them and 49 contained "significant personal information" - medical correspondence, love letters, pornography and 5,000 credit card numbers. One even had a year's worth of transactions with account numbers from a cash machine in Illinois.

Hashes

- Like a person’s fingerprint
- Uniquely identifies the file based on contents
- You can’t create the file from the hash
- Primary hash value used is Secure Hash Algorithm (SHA-1) specified in FIPS 180-1, a 160-bit hashing algorithm
  - $10^{45}$ combinations of 160-bit values
- “Computationally infeasible” to find two different files less than $2^{64}$ bits in size producing the same SHA-1
  - $2^{64}$ bits is one million terabytes
Hashes

- SHA-1 values can be cross-referenced by other products that depend on different hash values.
- Other standard hash values computed for each file include Message Digest 5 (MD5), and a 32-bit Cyclical Redundancy Checksum (CRC32), which are useful in CF tools and to users outside LE.
# Hash Examples

<table>
<thead>
<tr>
<th>Filename</th>
<th>Bytes</th>
<th>SHA-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT4\ALPHA\notepad.exe</td>
<td>68368</td>
<td>F1F284D5D757039DEC1C44A05AC148B9D204E467</td>
</tr>
<tr>
<td>NT4\I386\notepad.exe</td>
<td>45328</td>
<td>3C4E15A29014358C61548A981A4AC8573167BE37</td>
</tr>
<tr>
<td>NT4\MIPS\notepad.exe</td>
<td>66832</td>
<td>33309956E4DBBA665E86962308FE5E1378998E69</td>
</tr>
<tr>
<td>NT4\PPC\notepad.exe</td>
<td>68880</td>
<td>47BB7AF0E4DD565ED75DEB492D8C17B1BFD3FB23</td>
</tr>
<tr>
<td>WINNT31.WKS\I386\notepad.exe</td>
<td>57252</td>
<td>2E0849CF327709FC46B705EEAB5E57380F5B1F67</td>
</tr>
<tr>
<td>WINNT31.SRV\I386\notepad.exe</td>
<td>57252</td>
<td>2E0849CF327709FC46B705EEAB5E57380F5B1F67</td>
</tr>
</tbody>
</table>
Related History

- CRC concept dates from 1960’s
- MD5 algorithm published in 1991
- Tripwire open source tool 1992
- Unix command “md5sum” available
- FIPS 180-1 (SHA-1) published in 1995
- Unix command “sha1sum” available
- Known File Filter project 1998
- FIPS 180-2 (SHA-512) published in 2002
Hashes in P2P

KaZaA Peer-to-Peer (P2P) FastTrack File Formats

http://kzfti.cjb.net/

4/24/2003
SHA-1 Mathematics

- Bit sequence is padded to a multiple of 512
- Messages of 16 32-bit words, $n \times 512$, $n > 0$
- 80 logic functions are defined that accept 3 32-bit words and produce 1 32-bit word
- 80 constants defined, 5 32-bit buffers initialized
- 80 step loop:
  - Manipulate message into 80 32-bit words
  - Use shifts, functions, addition on buffers
- 160-bit SHA is string in the 5 32-bit buffers
## Application of RDS

<table>
<thead>
<tr>
<th>OS/Apps</th>
<th>Files installed</th>
<th>Percent identified</th>
<th>Files unknown</th>
<th>Files on distribution CD(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virgin Win 98</td>
<td>4,266</td>
<td>93%</td>
<td>297</td>
<td>18,662</td>
</tr>
<tr>
<td>Virgin NT4 WS</td>
<td>1,659</td>
<td>86%</td>
<td>239</td>
<td>17,904</td>
</tr>
<tr>
<td>Virgin Win 2Kpro</td>
<td>5,963</td>
<td>86%</td>
<td>839</td>
<td>16,539</td>
</tr>
<tr>
<td>Virgin Win ME</td>
<td>5,169</td>
<td>93%</td>
<td>383</td>
<td>11,512</td>
</tr>
<tr>
<td>Win 98+Office 2K</td>
<td>23,464</td>
<td>98%</td>
<td>596</td>
<td>43,327</td>
</tr>
<tr>
<td>Win ME+Office 2K</td>
<td>24,112</td>
<td>98%</td>
<td>526</td>
<td>32,758</td>
</tr>
<tr>
<td>NIST PC #1 W2K</td>
<td>18,048</td>
<td>35%</td>
<td>11,839</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #2 W2K</td>
<td>59,135</td>
<td>20%</td>
<td>47,124</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #3 WNT</td>
<td>14,186</td>
<td>54%</td>
<td>6,618</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #4 W98</td>
<td>16,397</td>
<td>55%</td>
<td>7,404</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #5 W98</td>
<td>34,220</td>
<td>75%</td>
<td>8,667</td>
<td>N/A</td>
</tr>
</tbody>
</table>
NIST Research

- Hash collisions
- Software distribution metrics
- Operating/File system effects
- Physical/Virtual machine effects
- “Mining” dynamic files
- Offsite hashing
Software Installation Issues

- Dynamic files are “missed” by RDS
- Installed on virtual machines which can be saved in the NSRL on media
- Delineation of static sections of files for probability of identification
- Independent of installation location
NARA Research

- Use hashing process on non-classified Presidential materials
- Identify application files
- Identify duplicate files
- Access to older installed software
NARA Statistics

- 93 computer systems
  - Pre-filtered to contain only software
- 51,146 individual files
- 7,610 file names
- 11,118 distinct files (SHA-1)
- 8,077 files originating in specific application(s)
- 4,326 of 8,077 exactly match application file names
Further NARA Research

- Building profile of a “master” image
- Statistical weights for application identification
- Cross-system relationships
- Installation locations
- Old compression technologies
NSRL Environment

- Isolated network with domain controller, DHCP
- Database server, File server, Web server
- Batching stations use web browser interface
- Hashing constellation
- Virtual machines for installations
- CVS source code repository
Input Process

- Package is acquired
- Web interface used to enter information about manufacturer, product, OS and assign an ID
- Media are batched
- Approximately 15 minutes per package
### Package Information

**NSRL Builder**

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Your Eyes Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>1.0</td>
</tr>
<tr>
<td>Bar Code</td>
<td>833648120467</td>
</tr>
<tr>
<td>Language</td>
<td>English</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Symantec</td>
</tr>
<tr>
<td>Application Type</td>
<td>Utility</td>
</tr>
</tbody>
</table>

**Contact Us:**

National Institute of Standards and Technology  
ATTN: NSRL Project  
100 Bureau Drive, Stop 8770  
Gaithersburg, MD 20899-8970 USA

4/24/2003
Hashing Operations

- Spring 2003 – accepting software
- Hashing constellation runs 24/7
- Processed over 13.4M files, 9M SHAs
- Byte signature file type verification
- CAB, ZIP, TAR, SFX, UU, compress
Data Verification

- Multiple and independent techniques from different perspectives
  - We use test files with known signatures
  - Parallel database system: Match results with other system
  - Human verification
  - Database rules and constraints
  - Periodic database queries: Predefined procedures to search for and report anomalies in the database
  - User feedback: Error reports and RDS updates
Future Operation Tasks

- More hardware platforms
- More archive tools
- Redundant hashing in constellation
- Scheduled rebatching
- Additional algorithms – AES
- Open source LAMP distribution
NSRL Accomplishments

- RDS CD Version 1.5 distributed 3/3/2003
  - 102 subscriptions (Vendors, corporations, universities, and law enforcement agencies)
  - Free redistribution, NIST traceable
- Incorporated into vendor products
- Used by FBI, DCCC, Secret Service, Customs Service (Homeland Security)
NSRL/CFTT Team
Contacts

Jim Lyle
www.cftt.nist.gov
cftt@nist.gov

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

Barbara Guttman
barbara.guttman@nist.gov

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