Computer Forensics:
Tool Testing
&
National Software Reference Library

Jim Lyle
Information Technology Laboratory
9 September 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
An Investigation

- Suspect
- Authorization
- Seize evidence
- Deliver to lab
- Make image
- Analyze
- Report Results
Getting the Evidence

1. Seize suspect system
2. Remove hard drive
3. Make working copy
4. Never touch original
Investigators Need …

Computer forensic investigators need tools that …

- Work as advertized and
- Produce results admissible in court
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)
In the Beginning

- Houston, we have a problem
- Money, who’s got the money?
- Setting up the program
Why NIST/ITL is involved

- NIST (OLES, ITL/SDCT)
- DoJ (NIJ, FBI)
- DoD (DCCC)
- TREASURY (USCS, USSS)
- National State/Local Agencies
- Software Vendors

- 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 meet vendor specs?

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

- … by whom?

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

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

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

Evaluating test methodology for …
- Hard drive imaging tools
- Software hard drive write protect
- Hardware hard drive write protect
- Deleted file recovery
Challenges

- No standards 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>

9/9/2003
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

9/9/2003
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
- CFTT is a topic of discussion at various forensic meetings
Web Hits by Country

Usage by Country for June 2002

- Unresolved/Unknown (31%)
- US Commercial (22%)
- US Government (4%)
- United Kingdom (4%)
- US Military (3%)
- Canada (2%)
- United States (2%)
- US Educational (2%)
- Australia (2%)
- Other (8%)

9/9/2003
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
Outline

- NSRL Description
- RDS Description
- RDS Field Use
- Hash System Overview
What is the NSRL?

- **National Software Reference Library (NSRL)**
  - Physical library of software, 1900 products
  - SQL Server database of known file signatures
  - Reference Data Set (RDS)
    - Extract of database on CD: 6,500,000 file signatures

**Goals**

- Automate the process of identifying known files on computers used in crimes
- Allow investigators to concentrate on files that could contain evidence (unknown and suspect files)
Addressing Law Enforcement Needs

- LE needed an unbiased organization
- LE needed traceability for the NSRL contents
- No repositories of original software available for reproducing data
- NSRL needs to work with many CF tools
Scope of the NSRL

- NIST has collected software for 2 years
- Software is recorded as the original source for known files and stored as a part of the NSRL
- Versions of OS, DBMS, photo editors, word processors, network browsers, compilers...
- Data formats, data dictionary and project status information is available on the website for RDS users and industry reference
What is the RDS?

NIST Special Database #28

NSRL
National Software Reference Library

Reference Data Set
Version 1.2 06/06/2002
What is the RDS?

Reference set of file profiles

- Each profile includes file name, file size, 4 file signatures (SHA1, MD4, MD5, CRC32), application name, operating system, etc.
- Extracted from files on original software CDs, diskettes, and network downloads
- A single application may have 10,000 separate file profiles
How to Use the RDS

- Eliminate as many known files as possible from the examination process using automated means.
- Discover files that do not contain expected contents (.exe file containing a bomb schematic, facility map).
- Look for files that should be installed, but are missing (incomplete deletion of pirated software).
- Look for files that could be suspect (hash matches, but file name does not).
- Provide rigorously verified data for forensic investigations.
RDS Field Use Example

You are looking for facility maps on a computer which is running Windows NT 4.0 Workstation.

Windows NT 4.0 operating system software contains 6753 images which are known gifs, icons, jpeg files
e.g.,

By using the RDS and an analysis program the investigator would not have to look at these files to complete his investigation.
Hash System Overview

- Environment
- Input Process
- Hashing
- Verification
- Future
Environment

- Isolated LAN
- Domain controller
- Database server
- File server
- CVS repository
- Virtual Machines
- Batching stations
- Hashing constellation
General Process

- Acquire
- Batch
- Hash
- Verify
- Accept/Reject
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>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Name</td>
<td>Your Eyes Only</td>
</tr>
<tr>
<td>Version</td>
<td>win95 1.0</td>
</tr>
<tr>
<td>Bar Code</td>
<td>057648128487</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>
<tr>
<td>Packaged Within</td>
<td></td>
</tr>
<tr>
<td>Comments</td>
<td></td>
</tr>
<tr>
<td>Location</td>
<td>G</td>
</tr>
</tbody>
</table>

Contact Us:

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

- July 2002 – accepting software, performing installations
- Currently hashing 5 p.m. until done
- 6 200MHz PCs in hashing constellation
- Averaging 10.5 hashes/sec.
- DVD with 250,000 files needed 30 hours
Hashes

- Compute a unique identifier for each file based on contents
- Primary hash value used in the NSRL RDS is the Secure Hash Algorithm (SHA-1) specified in Federal Information Processing Standard (FIPS) 180-1, a 160-bit hashing algorithm
- SHA-1 values can be cross-referenced by other products that depend on different hash values
Hashes

- Other standard hash values computed for each file include Message Digest 4 (MD4), Message Digest 5 (MD5), and a 32-bit Cyclic Redundancy Checksum (CRC32), which are useful in many CF tools and to users outside LE.
- Separate, parallel, and independent process is used to validate the results of the primary RDS implementation.
- Once verified and validated, the RDS is written to a master CD, duplicated, and distributed through NIST’s Standard Reference Data Office as Special Database #28 (www.nist.gov/srd/nistsd28.htm).
### 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>
Installation Hashes

- 300-800 files are “missed” by current RDS
- Compare automated hashes with real-world installed hashes
- Compare installed file sets across machines and OS’es
- Installed on virtual machines which can be saved in the NSRL on media
Use on Actual Machines

Clean OS
- 4622 files -360, 92% known – w98
- 7720 files -864, 89% known – w2k
- 5412 files -370, 93% known – wme

Actual NIST PCs
- 39631 files -7902, 80% known – w2k
- 18262 files -6395, 65% known – w98
- 75834 files -41638, 45% known – w2k, mgmt

9/9/2003
<table>
<thead>
<tr>
<th>OS/Apps</th>
<th>Files installed on HD</th>
<th>HD Files not in Hashkeeper</th>
<th>HD Files not in NSRL</th>
<th>Files on distribution CD(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virgin Win 98</td>
<td>4,266</td>
<td>142 (3%)</td>
<td>297 (7%)</td>
<td>18,662</td>
</tr>
<tr>
<td>Virgin NT4 WS</td>
<td>1,659</td>
<td>1,211 (72%)</td>
<td>239 (14%)</td>
<td>17,904</td>
</tr>
<tr>
<td>Virgin Win 2Kpro</td>
<td>5,963</td>
<td>783 (13%)</td>
<td>839 (14%)</td>
<td>16,539</td>
</tr>
<tr>
<td>Virgin Win ME</td>
<td>5,169</td>
<td>2,973 (57%)</td>
<td>383 (7%)</td>
<td>11,512</td>
</tr>
<tr>
<td>Win 98+Office 2K</td>
<td>23,464</td>
<td>313 (1%)</td>
<td>596 (2%)</td>
<td>43,327</td>
</tr>
<tr>
<td>Win ME+Office 2K</td>
<td>24,112</td>
<td>3,119 (13%)</td>
<td>526 (2%)</td>
<td>32,758</td>
</tr>
<tr>
<td>NIST PC #1 W2K</td>
<td>18,048</td>
<td>13,137 (72%)</td>
<td>11,839 (65%)</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #2 W2K</td>
<td>59,135</td>
<td>46,277 (79%)</td>
<td>47,124 (80%)</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #3 WNT</td>
<td>14,186</td>
<td>7,543 (53%)</td>
<td>6,618 (46%)</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #4 W98</td>
<td>16,397</td>
<td>8,360 (51%)</td>
<td>7,404 (45%)</td>
<td>N/A</td>
</tr>
<tr>
<td>NIST PC #5 W98</td>
<td>34,220</td>
<td>8,366 (25%)</td>
<td>8,667 (25%)</td>
<td>N/A</td>
</tr>
</tbody>
</table>

*Lower percentage is better*
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 Tasks

- Byte signature file type verification
- Self-extracting EXE files
- Redundant hashing in constellation
- Scheduled rebatching
- Additional algorithms - AES
NSRL Accomplishments

- RDS CD Version 1.2 distributed 6/6/2002
  - 124 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)
CFTT/NSRL Team
Contacts

Jim Lyle  Doug White
cfft@nist.gov  nsrl@nist.gov

Mark Skall
Chief, Software Diagnostics & Conformance Testing Div.
www.itl.nist.gov/div897  skall@nist.gov

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

9/9/2003