Computer Forensics Tool Testing at NIST

Jim Lyle
Information Technology Laboratory

Phone: (301) 975-3207
E-mail: JLYLE@NIST.GOV
WWW: http://www.cftt.nist.gov
Computers & The Internet

- Marvelous tools
- Improve quality of life
- Enable global communication
- Improve productivity
- Makes many activities easier, faster, ...
- Even criminal activity
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
- Produce results admissible in court
Admissible Results

Software tools must meet Daubert criteria
- Tested: accurate, reliable & repeatable
- Peer reviewed
- Generally accepted methodology
Response to Problem

- Independent testing of forensic tools
- Public review of results
- Apply black box testing theory to tools
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 investigations (NSRL)
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
Project Sponsors

- NIST/OLES (Program management)
- NIJ (Major funding)
- FBI (Additional funding)
- DOD (Equipment and support)
- Homeland Security (Technical input)
- State & Local agencies (Technical input)
Project Tasks

- Identify forensics functions e.g.,
  - Disk imaging,
  - Hard drive write protect,
  - Deleted file recovery
  - String searching

- Develop specification for each function
- Peer review of specification
- Test methodology for each function
- Test Tools (by function) & Report results
Current Activities

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

- No standards or specifications for tools
- Arcane knowledge domain (e.g. DOS, Windows drivers)
- Reliably faulty hardware
- Many versions of each tool
Overview of Methodology

- CFTT directed by Steering Committee
- Functionality driven
- Specifications developed for specific categories of activities, e.g., disk imaging, hard drive write protect, etc.
- Test methodology developed for each category
Developing a Specification

After tool function selected by SC …

- Focus group (law enforcement + NIST) develop tool function specification
- Spec posted to web for public comment
- Comments incorporated
- Develop test environment
Tool Test Process

After SC selects a tool …
- Acquire tool & review documentation
- Select test cases
- Execute test cases
- Produce test report
## Disk Imaging Test Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functions</td>
<td>Copy, Image, Verify</td>
</tr>
<tr>
<td>Source interface</td>
<td>BIOS to IDE, BIOS to SCSI, ATA, ASPI, Legacy BIOS</td>
</tr>
<tr>
<td>Dst interface</td>
<td></td>
</tr>
<tr>
<td>Relative size</td>
<td>Src=Dst, Src&lt;Dst, Src&gt;Dst</td>
</tr>
<tr>
<td>Errors</td>
<td>None, Src Rd, Dst Wt, Img R/W/C</td>
</tr>
<tr>
<td>Object type</td>
<td>Disk, FAT12/16/32, NT, Ext2</td>
</tr>
<tr>
<td>Remote access</td>
<td>Yes, no</td>
</tr>
</tbody>
</table>
Capabilities to test disk imaging

- Accuracy of copy
  - Compare disks
  - Initialize disk sectors to unique content
- Verify source disk unchanged
- Corrupt an image file
- Error handling: reliably faulty disk
Test Case Structure: Setup

1. Record details of source disk setup.
2. Initialize the source disk to a known value.
3. Hash the source disk and save hash value.
4. Record details of test case setup.
5. Initialize a destination disk.
6. If the test requires a partition, create and format a partition on the destination disk.
7. If the test uses an image file, partition and format a disk for the image file.
Test Case Structure: Run Tool

8. If required, setup I/O error
9. If required, create image file
10. If required, corrupt image file
11. Create destination
Test Case Structure: Measure

12. Compare Source to Destination
13. Rehash the Source
Test Logging

- Log everything, automatically if practical
- Hardware, Software, Versions
- Time/date
- Operator
Legacy BIOS Quirks

- Some may under report drive size
- Example, Quantum SIROCCO1700A has 3335472 sectors 3309/16/63 spc 1008
- BIOS: 3,330,432 sectors with geometry 826/64/63 spc 4032
- BIOS under reports by 1.25 logical cyls and 5 physicals
Evaluating Test Results

If a test exhibits an anomaly …
1. Look for hardware or procedural problem
2. Anomaly seen before
3. If unique, look at more cases
4. Examine similar anomalies
Refining the Test Procedure

- During `dd` testing some results seemed to indicate that the Linux environment was making a change to the source disk.
- After investigation we found that the problem was actually the test procedure.
Hard Drive Write Protect

- Can be done either in hardware or software
- Software write protection limited to specific environment: BIOS access or device driver
- Hardware write protection more general
Hard Drive BIOS Access

Application program

issue int 0x13 cmd

BIOS interrupt 0x13

issue cmd to drive return

Disk drive & controller
SWB Tool Operation

Application program

issue int 0x13 cmd

SWB tool

block -> allow

return

BIOS interrupt 0x13

issue cmd to drive

Disk drive & controller

return
Test Harness Operation

Test harness

issue int 0x13 cmd

query result

SWB tool

block

return

allow

interrupt 13 monitor

block

tally

allow

BIOS interrupt 0x13

issue cmd to drive

return

Disk drive & controller
HWB Testing

CPU → BUS 1 → HWB
Send I/O CMD to Device

HWB → BUS 2 → PROTOCOL ANALYZER
Monitor Bus Traffic

PROTOCOL ANALYZER → Device
Return result to CPU
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
- CFTT cited in some high profile court cases
Available Specifications

- Hard Drive Imaging (e.g., Safeback, EnCase, Ilook, Mares imaging tool)
- Write Block Software Tools (e.g., RCMP HDL, Pdblock, ACES)
- Write Block Hardware Devices (A-Card, FastBlock, NoWrite) – not final
Specifications Under Development

- String Searching
- Deleted File Recovery
- Revised Disk Imaging
Available Test Reports

- Sydex SafeBack 2.0
- NTI Safeback 2.18
- EnCase 3.20
- GNU dd 4.0.36 (RedHat 7.1)
- FreeBSD 4.4 dd
- RCMP HDL V0.8
Test Reports in Production

- RCMP HDL V0.4
- RCMP HDL V0.5
- RCMP HDL V0.7
Available Testing Software

- FS-TST – tools to test disk imaging: drive wipe, drive compare, drive hash (SHA1), partition compare. (DCCI uses these tools)
- SWBT – tools to test interrupt 13 software write blockers
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
Contacts

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

Doug White
www.nsrl.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

5/8/2017