A Comparison of Computer Forensic Tools: An Open-Source Evaluation

Adam Cervellone, B.S., Graduate Student, Marshall University Forensic Science Center, 1401 Forensic Science Drive, Huntington, WV 25701
901725850

Agency Supervisor-Robert Price Jr., M.S., Forensic Scientist I, North Carolina State Crime Laboratory, 121 E. Tryon Road, Raleigh NC 27601

Technical Assistant- Joshua Brunty, M.S., Marshall University Forensic Science Center, 1401 Forensic Science Drive, Huntington, WV, 25701

MU Topic Advisor-Terry Fenger, Ph.D., Marshall University Forensic Science Center, 1401 Forensic Science Drive, Huntington, WV, 25701

Abstract

The world of digital forensics is an ever-evolving field with multiple tools for analysis from which to choose. Many of these tools have very focused functions such as Mac and iOS device analysis, registry examination, steganography analysis, mobile device examination, password recovery and countless others. Other tools are full featured suites capable of analyzing a large case containing multiple items. The major problem with many of these tools is cost. While they may be robust, they may not be affordable for a smaller lab that wants to do digital forensics. This research focuses on industry standard forensic software such as: Guidance Software® EnCase® Forensic 6, AccessData® FTK® (Forensic Toolkit) 5, as well as SANS SIFT Workstation 3.0. The SIFT Workstation is a freely available open-source processing environment that contains multiple tools with similar functionality to EnCase® and FTK®. This study evaluates the processing and analysis capabilities of each tool. In addition to processing functionality, a simple cost analysis study was done. The latter portion of the research displayed how much a lab may have to spend to get a single examiner fully on-line with each tool. While comparison studies between commercially available software have been done and published, research comparing industry standard tools with an open-source tool is not well documented.

For this study, mock test cases were created using North Carolina State Crime Laboratory (NCSCL) Mac Minis and Dell Latitude D810 laptops. The hard drives contained in these items were hashed and imaged via EnCase® Forensic 6.19.7.2 and fully processed according to NCSCL guidelines in EnCase® Forensic 6.19.7.2, FTK® 5.6.3, and the SIFT Workstation 3.0. In addition to evaluating analysis, the tools were also evaluated based on their ability to create a virtual machine from the evidence file as well as on overall cost for a single examiner.

This research has shown that the SIFT workstation is a viable option to use as a forensic tool from a financial and functionality perspective. Its capabilities are vast and are similar to

those of FTK® and EnCase® Forensic, however, due to its open-source nature and heavy reliance on the Linux Terminal and command line, it is advised that only an examiner highly skilled in Linux use the SIFT Workstation for casework.

Introduction

Much like the world of Forensic Science as a whole, the discipline of digital or computer forensics is an ever-evolving field of play, pitting the examiner against the system they are trying to analyze. To accomplish this task, examiners in government labs and private companies employ software to recover information from an item in question. These software tools range in abilities from single functions such as Arsenal Recon's Registry Recon, which is a registry recovery tool, to all-encompassing software suites such Guidance Software® EnCase® Forensic, Katana Forensics Lantern 4, and AccessData® FTK® (Forensic Toolkit) just to name a few. These tools are the workhorses of modern digital forensics but are often very different in function and ability, as well as being highly variable in cost for an examiner to become fully functional [(1,2)].

As stated above, digital forensic tools often vary in overall performance. The two software tools that are the industry standard are AccessData® FTK®, current version 5.6.3, and Guidance Software® EnCase® Forensic, current version 7.10 and 6.19.7.2 are both currently in use. Both of these tools are built to work in a Windows OS (Operating System) and on highly specialized computer [(3, 4)]. EnCase® and FTK® are designed to help an examiner fully process a case and, though these suites work differently, they can retrieve different types and amounts of data. This is of interest to the digital forensics community due to the influence one software suite may play in how much and what type of evidence can be recovered. As with many commercially available products, there is a steep cost involved solely in purchasing the tool, not to mention training or certification. There are open source forensic tools that claim to be able to process a case while remaining freely available [5]. For the purpose of this study, EnCase® Forensic 6.19.7.2 will be compared to FTK® 5.6.3 and the open source tool – the SIFT Workstation 3.0.

Two major problems exist in the modern digital forensics. The first is cost of tools, which affects more than just digital forensics examiners. It affects whole labs that are often on a tight budget that may be out of their control to some extent. Most labs cannot afford to have copies of every tool on the market [4]. The use of open-source tools can address this issue, but they must be properly vetted against the industry standard tools if they are to ever be used in a forensic environment. In addition to impacting examiners and labs, open-source tools can also be used in an education environment. This is especially helpful for academic forensic programs that want to enable students to have hands on experience with tools, but have a limited budget to purchase tools. Hawthorne and Shumba used the SIFT Workstation in their study of teaching digital forensics online as a means to make learning digital forensics more affordable for students [6]. Their study focused mostly on general usability and the opinions of students and faculty; however it did not cover capabilities of the tools from an examiner standpoint.

The second problem is combating the rise of cloud computing. Many users use webmail applications such as Gmail, Outlook.com, Mail.com and many others to host their personal email instead of a desktop client. Unlike email that is read and written through a client such as Apple's Mail, webmail is stored on an offsite server hosted by a corporation. Use of a virtual machine made from the evidence image file would allow examiners to see what a suspect saw as user of the evidence computer. It can be done in a forensically sound manner by writing all "changes" to a separate cache file that does not in any way change the evidence file being examined. If an examiner had this capability and access to usernames and passwords for a system, there is a chance they could view webmail in its native state.

This study will have three primary focuses. The first focus will be the ability of the tool to be an overall case processor. This will involve using an acquired E01 (EnCase® evidence file) and processing the image in each of the three tools. The second focus will be a virtualization study of each tool's ability to create a virtual machine using the E01 image files. The third and final focus of this study will be a simple cost analysis of each of the tools that will factor in cost of a single license, available support, available certification and cost of course work and certification.

Research Questions

1. Can the SIFT Workstation hash and image an evidence item in a forensically sound manner?
2. How does the SIFT Workstation compare as a case processor to industry standard tools?
3. Is SIFT a viable option as a forensic tool in terms of cost and functionality when compared to industry standard tools?

Materials and Method

This section will outline the various computers, software tools and methods used in this study. Each tool processes and analyzes in a different fashion and as such, one concise methodology for all three tools was not able to be used.

Materials

The following materials were used for the study

- Forensic Computers Towers
  - Forensic Tower II
    - Test Case 2 EnCase® 6 Processing
  - Forensic Tower III
    - Test Case 1 Processing
    - Test Case 2 FTK® and SIFT Processing
- Guidance Software EnCase® Forensic 6.19.7.2
- AccessData® FTK® 5.6.3
- VMware Player 7 Free
- SANS SIFT Workstation 3.0
- Two Apple Mac mini A1283 computers
- Two Dell D810 Latitude Laptops
- FireWire cable
- 1TB SATA Target Hard Drive
- Oracle VirtualBox 5.0

For this study, two mock case scenarios were created, processed in each of the three forensic tools, and reports were generated for each case in each of the forensic tools if possible.

Case Preparation

This section describes how the two test cases used in this study were prepared prior to processing to forensic tool. Each subsection pertains to an individual test case that was developed.

Test Case 1

Two Apple® Mac Mini A1283 computers were restored to factory settings by using the Apple OS X Install disc. The OS was restored using the ‘Erase and Install” option in the OS X installer window. When both systems were restored, a single user account with a password was set for each computer. Various documents were generated, images from internet searches were downloaded, and a Yahoo email account for each user was made. These email accounts were synchronized with Apple’s Mail application. Emails were sent to and from each user using the Yahoo Mail addresses from Apple’s Mail application. Yahoo Messenger was also installed on each system and instant messages were sent between the systems. Originally the user’s home folder was encrypted using FileVault but due to unforeseen challenges, the FileVault encryption had to be removed. When the case was finalized, the Mac Minis were forensically imaged via a hardware write-blocker using FireWire target mode to avoid unnecessary damage to the Mac Minis.

Test Case 2

Two Dell Latitude D810 laptops running Windows XP Professional were restored using a Windows XP restore disc. This restoration returned the system to original standards. A second administrator user account was created and password protected on both laptops. This account is where all Test Case 2 evidence was generated. Webmail accounts from Gmail and Mail.com were used for suspect communication. Images were downloaded from various search engines. YouTube videos were downloaded using Basic YTD (YouTube Downloader) and converted into a standard video format, such as AVI or WMV. Documents such as .doc and .ppt files were also created.

Case Processing

This section outlines the methods used in this study. Each subsection refers to a particular step or tool used.

EnCase® 6.19.7.2 Hashing and Acquisition

Each Mac Mini was connected to the forensic tower via a FireWire cable attached to the back of the Mac mini and the external hardware write-blocker. To image the hard drive inside the Mac Mini computers, they were placed in FireWire Target mode by pressing Command + T while the computer was booting. Upon successful connection via FireWire target mode, EnCase®

6.19.7.2 was opened on the forensic tower in acquisition mode. Immediately after each item was added to the case, the drive was hashed and given an MD5 hash value. This hash value is a multi-character alpha-numeric value that serves as a unique value for a particular digital evidence item. All evidence items for each case were hashed in EnCase® 6.19.7.2.

When each item was done being hashed, each item was then acquired. This acquisition step is the EnCase® term for imaging. A compressed bit stream image known as an Expert Witness File/EnCase Image File (*.ewf1/*.E01) was created from each drive. The images generated were labeled Item #.E01. These are the image files used throughout this research. It should be noted that due to the size of the images, EnCase® creates split image files and can natively import these as a single hard drive image. Upon acquisition of both images in EnCase®, a licensing dongle was attached to the forensic tower via USB 2.0 port. The dongle switched EnCase® from acquisition mode to Law Enforcement mode, known as Forensic mode in EnCase® 7.x and beyond, which allows for full case processing.

Hard drives for Test Case 2 were removed from the Dell Latitude D810 laptops in which they were housed and then attached to the forensic tower. Both of these hard drives were IDE drives with pin 20 blocked. Once mounted into a write-blocked IDE bay in the forensic tower, EnCase® 6.19.7.2 was opened in acquisition mode and the drives were hashed and acquired just as the hard drives in Test Case 1 were.

EnCase® Forensic 6.19.7.2 Processing

For each day that a case was worked in EnCase, a USB flash drive with a known MD5 hash value was verified using EnCase. This verification step was nearly identical to the hashing and acquisition steps used to process an evidence drive. The major departing feature was that while the evidence drives were connected to a write-blocked hard drive bay or via FireWire target mode to the built in write-blocker, the USB flash drive was connected via USB 3.0 port that was present in the hardware write-blocker.

With EnCase® open in Law Enforcement mode, the test case was opened for processing. To process the cases in EnCase®, the following steps were used in order unless otherwise stated:

1. Partition Finder
2. Recovered Folders
3. Signature Analysis
4. Creation of EnCase® reports
5. Recording System Information
6. Encrypted Files Search
7. Keyword Search
8. Manual Carving
9. Pictures Search
10. Unallocated Pictures Search
11. Movies Search
12. Web Pages Search
13. Documents Search
14. Email Search
15. Chat Log Search
16. Link File Search
17. Internet Shortcuts/Bookmarks Search
18. Recycle Bin Analysis
19. Address Book Search
20. Program Search
21. Final reports created from EnCase® bookmarks using HTML reports

FTK® 5.6.3

Processing in FTK began with using the acquired images created using EnCase 6.19.7.2. During the addition of images, FTK was automatically set to index the evidence and perform file carving. The time zone was set to match the time zones used on the evidence drives. With the case open, methods similar to the aforementioned ones in EnCase® Forensic were employed. Steps executed and attempted are listed below.

1. Partition Finder
2. Recovered Folders
3. Signature Analysis
4. Creation of FTK® reports

5. Recording System Information
6. Encrypted Files Search
7. Keyword Search
8. Manual Carving
9. Pictures Search
10. Unallocated Pictures Search
11. Movies Search
12. Web Pages Search
13. Documents Search
14. Email Search
15. Chat Log Search
16. Link File Search
17. Internet Shortcuts/Bookmarks Search
18. Recycle Bin Analysis
19. Address Book Search
20. Program Search
21. Final report created from FTK® bookmarks using HTML and/or PDF reports found under File → Report

SIFT Workstation

The *.E01 images generated during the hashing and acquisition step in EnCase 6.19.7.2 were copied into a virtual machine running the SIFT Workstation 3.0. The workstation uses Ubuntu 14.04 “Trusty Tahr” LTS (Long Term Support) as its base OS. This virtual machine was run using VMware Player 7, given 4.0 GB of RAM and given the two SIFT VMDK files for hard disks. Virtual Disk 1 was 260 GB and Virtual Disk 2 was 1.0 TB. Once all of the evidence files for a case were copied to the workstation and stored in Virtual Disk 2, a Terminal window was opened and the following command “sudo su -” was run. This command allows the normal user profile to switch to the root, also known as superuser profile. This profile allows full access to all files on the system.

Once root access was obtained, the following commands were run in order; fdisk –l, ewfacquire <drive path>, ewfverify <image path>, and autopsy. Fdisk –l generates a listing of all drives that are seen by the workstation. This was used to select the evidence drive used. The drive path was /dev/sdc or /dev/sdd. Once the drive path was known, ewfacquire <drive path>

was run to acquire the drive image in an .E01 format to a specified file path. For the sake of this study the file path was /cases/Test_Case_#/Verification/Verification_mm_dd_yyyy.E01. The final command run was autopsy, which started the Autopsy forensic tool.

Within Autopsy, a new case was created for each test case to be analyzed. Once a new case was created, Autopsy then prompted the user to create a new host for the case. This host would store all case files such as: reports, log files, images (either symbolic links or whole files), keyword search output files, etc… The final step in Autopsy is to add the image files. The software can handle *.E01 files, *.dd (RAW) files, as well as *.AFF (Advanced Forensics Format) files. The split image files created during acquisition were added by typing the file path and file name to the images. For this research, the file path was /cases/Images/<Image name>.*. Autopsy used ".*" to account for split image files from a single item.

To process the case after evidence items were brought into the case, the appropriate directory was chosen for each item; /2/ for Test Case 1 items and D:/ for Test Case 2 items. By selecting the directory that contains the evidence, a file list appears with all subdirectories and files contained immediately within the parent directory. Each subdirectory is listed as a blue hyperlink with the format /subdirectory name/. When a subdirectory was clicked, it elicited the same behavior as when selecting the initial parent directory. Following this step, a string extraction and keyword search was run using the keyword text file generated for each case. Keywords were searched one at a time and search results were separated into four categories: allocated ASCII, allocated Unicode, unallocated ASCII, and unallocated Unicode. It should be noted that due to time constraints and time consuming processing that Item 1 was the only Test Case 1 item fully processed in Autopsy 2.24. Items from Test Case 2 were treated in similar fashion for Autopsy 2.24.

For Test Case 2, one other tool was tested: Foremost. Foremost is a command line file carving tool that searches evidence files for common file types such as doc, exe, pf, ost by default. It exports the file to an output directory designated by the examiner. From the shell prompt with root access the command "foremost –o '/media/sansforensics/WD External/cases/Test_Case_2/Foremost (_2)' –i /cases/Test_Case_2/Images/Item#*". The (_2) denotes a second Foremost folder, named "Foremost_2", created by the examiner to store Item 2 data. Item 1 data was stored in the "Foremost" folder. The foremost command was also rerun with identical output paths and image files but added the -t all switch before the –o switch in the command such that it took the form "foremost –t all –o '/media/sansforensics/WD External/cases/Test_Case_2/Foremost (_2)' –i /cases/Test_Case_2/Images/Item#*". The "–t all" switch selects all predefined file carvers that Foremost can run so it can carve out as many files and file types as possible files. The original runs of Foremost were sent to the recycling bin of the external drive but not deleted.

Virtualization Study

For this portion of the study, the software tools were evaluated on their ability to create a virtual machine from the E01 image of the evidence items.(7) Due to the inherent differences between the tools, methods for each tool will be described separately.

EnCase® Forensic 6.19.7.2

Following the EnCase® Computer Forensics II training manual, a test case was opened [(8)]. Within the evidence tree, an evidence item was right clicked and "Mount as Emulated Disk…" was selected. From the dialog box that opens the client info tab was selected. The "Create new cache" radio button was clicked, a cache path was given and "Disable caching" was unchecked. The EnCase recommended method says to use LiveView 0.7b. This was tried but failed due to forensic tower network restrictions. A second method employing command prompt and VirtualBox 5.0 was then tested [9]. Using a command prompt window run as an administrator, the following commands were run:

1. cd \
2. cd Program Files\Oracle\VirtualBox
3. Vboxmanage internalcommands createrawvmdk -filename < output path\filename >.vmdk -rawdisk \\.\physicaldrive#
    a. # denotes the physical drive letter assigned to the mounted E01 file when it was mounted using EnCase®

Once these commands were executed, Oracle VirtualBox, version 4.3.30 or 5.0, was run as an administrator. A new virtual machine was created, the RAM was set to 4096 MB and the virtual disk selection was pointed toward the VMDK file created from the E01 evidence file.

FTK® 5.6

Using FTK, a test case was opened. From the "Explore" tab, an evidence item was right clicked, and "Mount Image to Drive" was chosen. Based on steps fromsecurityisfun.net[9], the mount type was "physical only", the mount method was "Block Device/Writeable", and a cache folder destination in which changes to the drive would be written was created on the desktop of the C drive of the Forensic tower. A command prompt window was opened and the following commands were run in the following order:

1. cd \
2. cd Program Files\Oracle\VirtualBox
3. Vboxmanage internalcommands createrawvmdk -filename <output path\filename>.vmdk -rawdisk \\.\physicaldrive#

a. # denotes the physical drive letter assigned to the mounted E01 file when it was mounted using FTK®

Once these commands were executed, Oracle VirtualBox, version 4.3.30 or 5.0, was run as an administrator. A new virtual machine was created, the RAM was set to 4096 MB and the virtual disk selection was pointed toward the VMDK file created from the E01 evidence file. The virtual machine was then booted.

SIFT Workstation 3.0

Within SIFT, a terminal window was opened and the following commands were run:

1. Sudo su
   - SIFT password entered
2. Mkdir /mnt/ewf1
3. Mount_ewf.py <E01 image file path> /mnt/ewf1
4. qemu-img convert /mnt/ewf1/ewf1 -O vmdk /cases/VirtualMachines/Item1.vmdk

With the VMDK file created and the evidence image mounted as ewf1, Oracle VirtualBox for Linux was used to create and run the VM.

Cost Analysis

This section conveys methods used for the cost analysis study portion of this research. Guidance Software®, AccessData® , and SANS were each contacted via email or telephone and were asked for the following information: Cost of single user license, cost of support/maintenance, certification available, training cost and certification cost, and cost of any required materials for courses. The price totals received from sales representatives were then added together for each tool to develop the total cost for a single examiner.

Results

EnCase® Forensic 6.19.7.2 and FTK® 5.6.3 Analysis

Table 1 contains approximate amounts of each forensically relevant file category that was known to be present on the drives for test case before the cases were analyzed in EnCase® and FTK®. The values are separated by test case number and not by evidence drive used in the cases unless otherwise stated. Table 2 contains the amount of data recovered by each tool for each case

Table 1: Known evidence in test cases

Test Case	Documents	Allocated Picture files	Unallocated Picture files	Email	Chat Logs	Video Files	Encrypted files	Link Files/ Aliases	Address Book Created
Test Case 1	3	80	48	5-10	Various messages sent over a period of a week	N/A	0	4	Yes, using Apple Mail
Test Case 2	3	80-100/drive	50/drive	0 (Webmail used)	N/A	9	0	5	No

Table 2: Files recovered via bookmarks during case analysis

Tool/Test Case	Documents	Allocated Pictures	Unallocated Pictures	Email	Chat Logs	Video files	Encrypted files	Link Files	Recycle bin files	Address Book Files
EnCase® Forensic/ Test Case 1	3	80	48	35*	3	N/A	0	0*	23	2*
FTK®/Test Case 1	3	74	0	17	3	N/A	2	0	20	10
EnCase® Forensic/ Test Case 2	3	207	0*	0	N/A	9	0	97	21	2
FTK®/Test Case 2	4	233	0	0	N/A	8	N/A	64	24	2

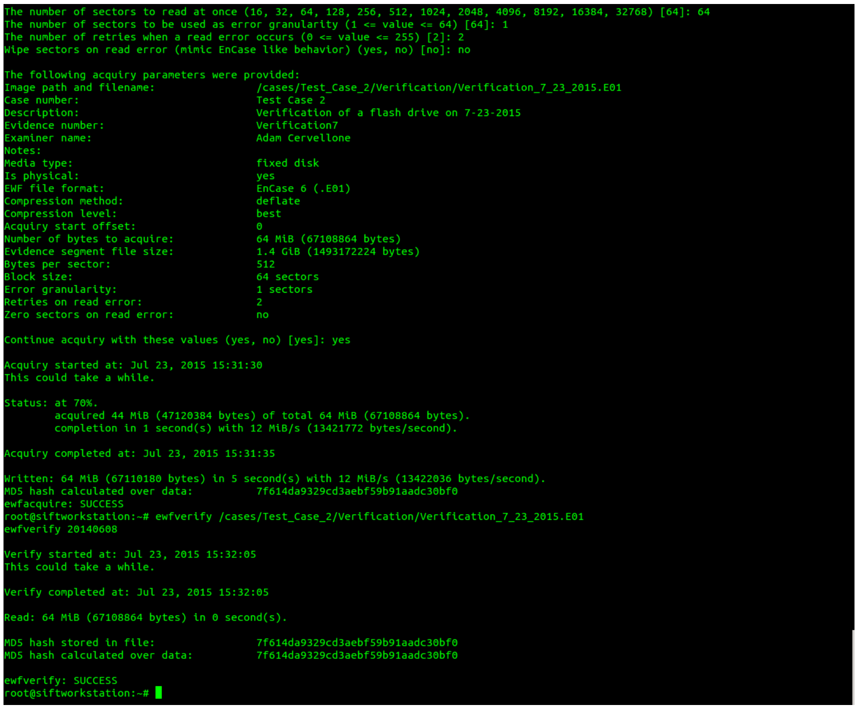
* denotes explainable differences or issues based on analysis

Both EnCase® and FTK® can make use of the NIST NSRL hash database. The NSRL hash database contains known hash values of OS files and other pieces of software [10]. This helps to filter out files that are supposed to be on an evidence drive regardless of alleged activity by the user. This would aid the examiner in their work. Currently this is not done in the Digital Evidence section of the NCSCL.

SIFT Workstation 3.0

The SANS SIFT Workstation was able to successfully hash and acquire an evidence drive attached to a hardware write-blocker that was connected to the VMware virtual machine. Figures 1 shows the terminal window that displayed the parameters used for acquiring and hashing using the libewf-tools ewfacquire and ewfverify. These tools were successful in acquiring and verifying the flash drive with a known hash value every time it was attempted.

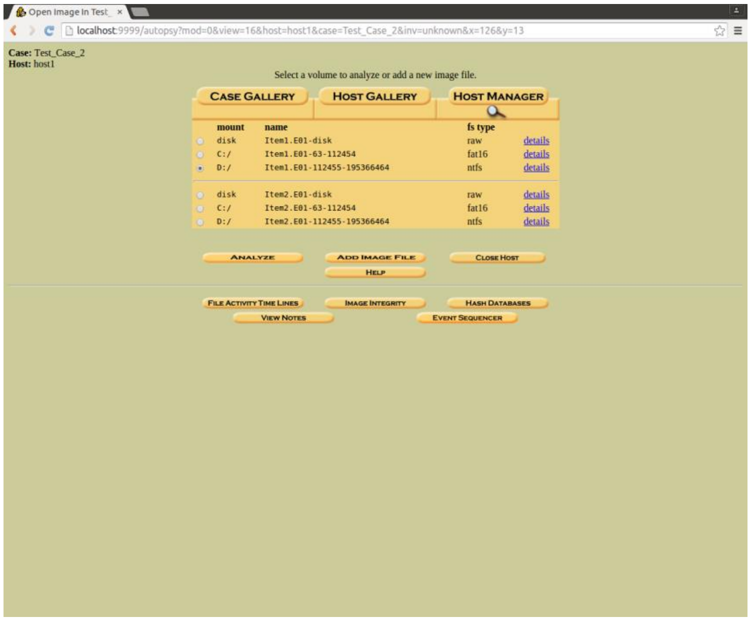
Ewfverify was also run on the Test Case 2 images and was successful at computing the correct MD5 hash value.



```
The number of sectors to read at once (16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768) [64]: 64
The number of sectors to be used as error granularity (1 <= value <= 64) [64]: 1
The number of retries when a read error occurs (0 <= value <= 255) [2]: 2
Wipe sectors on read error (mimic EnCase like behavior) (yes, no) [no]: no

The following acquiry parameters were provided:
Image path and filename:                /cases/Test_Case_2/Verification/Verification_7_23_2015.E01
Case number:                            Test Case 2
Description:                            Verification of a flash drive on 7-23-2015
Evidence number:                        Verification7
Examiner name:                          Adam Cervellone
Notes:
Media type:                             fixed disk
Is physical:                            yes
EWF file format:                        EnCase 6 (.E01)
Compression method:                     deflate
Compression level:                      best
Acquiry start offset:                   0
Number of bytes to acquire:             64 MiB (67108864 bytes)
Evidence segment file size:             1.4 GiB (1493172224 bytes)
Bytes per sector:                       512
Block size:                             64 sectors
Error granularity:                      1 sectors
Retries on read error:                  2
Zero sectors on read error:             no

Continue acquiry with these values (yes, no) [yes]: yes

Acquiry started at: Jul 23, 2015 15:31:30
This could take a while.

Status: at 70%.
        acquired 44 MiB (47120384 bytes) of total 64 MiB (67108864 bytes).
        completion in 1 second(s) with 12 MiB/s (13421772 bytes/second).

Acquiry completed at: Jul 23, 2015 15:31:35

Written: 64 MiB (67110180 bytes) in 5 second(s) with 12 MiB/s (13422036 bytes/second).
MD5 hash calculated over data:          7f614da9329cd3aebf59b91aadc30bf0
ewfacquire: SUCCESS
root@siftworkstation:~# ewfverify /cases/Test_Case_2/Verification/Verification_7_23_2015.E01
ewfverify 20140608

Verify started at: Jul 23, 2015 15:32:05
This could take a while.

Verify completed at: Jul 23, 2015 15:32:05

Read: 64 MiB (67108864 bytes) in 0 second(s).

MD5 hash stored in file:                7f614da9329cd3aebf59b91aadc30bf0
MD5 hash calculated over data:          7f614da9329cd3aebf59b91aadc30bf0

ewfverify: SUCCESS
root@siftworkstation:~#
```

Figure 1: Terminal output of Ewfacquire and Ewfverify parameters and results

Autopsy™ was able to successfully create and open both cases for this study. In a similar fashion to EnCase® and FTK® it displays all partitions and directories found on an evidence drive. Autopsy™ can display many of the file types that EnCase® and FTK® did as well. It is different in that while EnCase® and FTK® can often view files such as .doc and .ppt in their native format within the software, Autopsy™ will export these out as downloads in the browser used. Figure 2 and 3 show the primary screen for opening a case and evidence drive navigation respectively.

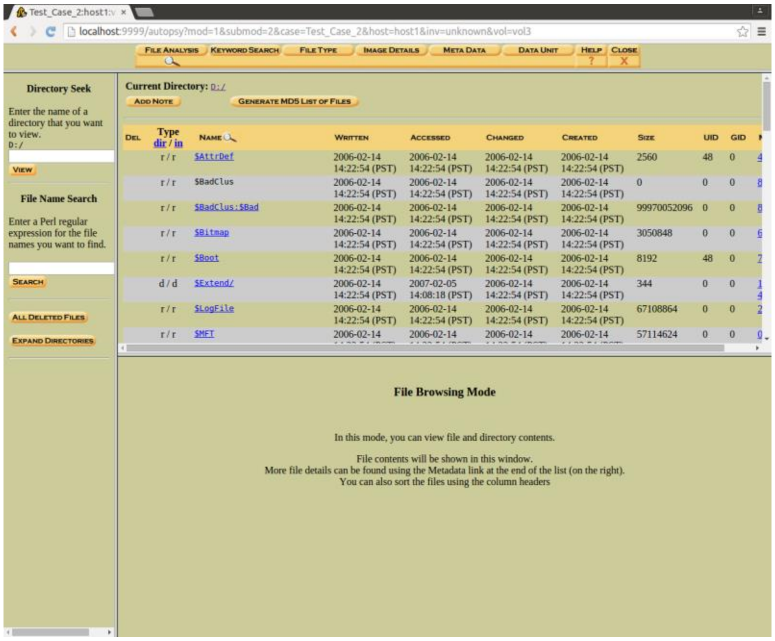


Case: Test_Case_2
Host: host1

Select a volume to analyze or add a new image file.

	mount	name	fs type	
○	disk	Item1.E01-disk	raw	details
○	C:/	Item1.E01-63-112454	fat16	details
●	D:/	Item1.E01-112455-195366464	ntfs	details
○	disk	Item2.E01-disk	raw	details
○	C:/	Item2.E01-63-112454	fat16	details
○	D:/	Item2.E01-112455-195366464	ntfs	details

Figure 2: Autopsy™ default case screen



Directory Seek

Enter the name of a directory that you want to view.
D:/

File Name Search

Enter a Perl regular expression for the file names you want to find.

Current Directory: D:/

Del	Type dir / in	Name	Written	Accessed	Changed	Created	Size	UID	GID	
	r / r	$AttrDef	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2560	48	0	4
	r / r	$BadClus	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	0	0	0	8
	r / r	$BadClus:$Bad	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	99970052096	0	0	8
	r / r	$Bitmap	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	3050848	0	0	6
	r / r	$Boot	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	8192	48	0	7
	d / d	$Extend/	2006-02-14 14:22:54 (PST)	2007-02-05 14:08:18 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	344	0	0	1 4
	r / r	$LogFile	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	2006-02-14 14:22:54 (PST)	67108864	0	0	2
	r / r	$MFT	2006-02-14	2006-02-14	2006-02-14	2006-02-14	57114624	0	0	0

File Browsing Mode

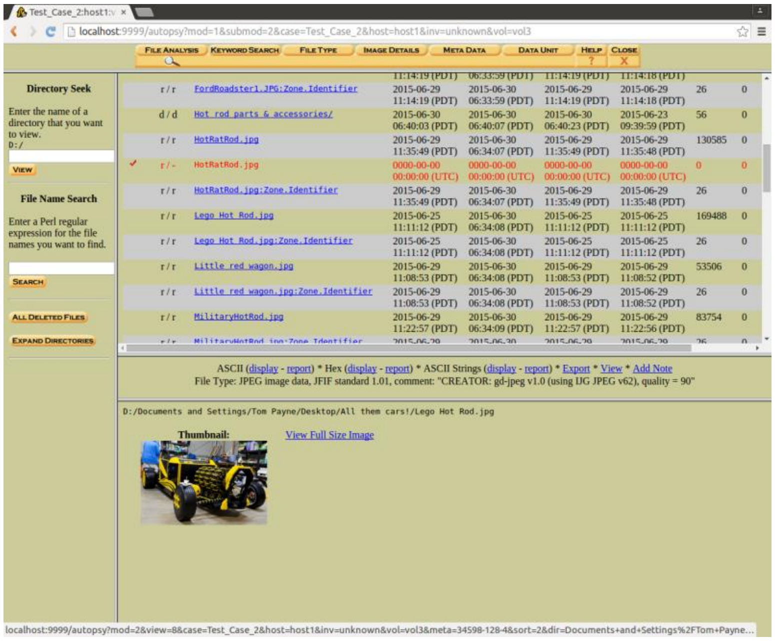
In this mode, you can view file and directory contents.

File contents will be shown in this window.
More file details can be found using the Metadata link at the end of the list (on the right).
You can also sort the files using the column headers

Figure 3: Autopsy™ file analysis screen

Figure 4 displays how Autopsy™ handles a picture files, such as a JPEG.

Using the Foremost file carving tool with the "-t all" option, the SIFT workstation was able to extract bmp, doc, exe, gif, jpg, mp4, ost, pf, png, wav and zip files from Test Case 2, Item 1. From Item 2 it extracted bmp, exe, gif, jpg, ost, pf, png, wav and zip files. Figure 5 shows the terminal window displaying the initial Foremost command without the "–t all" option just to demonstrate how the command works. Figure 6 shows Foremost processing part of an image file while running with the "-t all" option



localhost:9999/autopsy?mod=1&submod=2&case=Test_Case_2&host=host1&inv=unknown&vol=vol3

Enter the name of a directory that you want to view.
D:/

Enter a Perl regular expression for the file names you want to find.

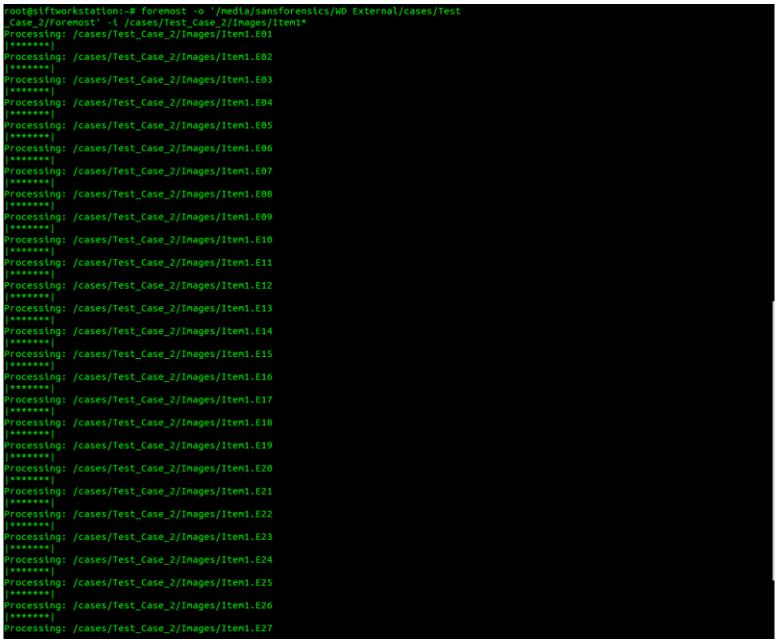
			11:14:19 (PDT)	06:33:59 (PDT)	11:14:19 (PDT)	11:14:18 (PDT)		
	r / r	FordRoadster1.JPG:Zone.Identifier	2015-06-29 11:14:19 (PDT)	2015-06-30 06:33:59 (PDT)	2015-06-29 11:14:19 (PDT)	2015-06-29 11:14:18 (PDT)	26	0
	d / d	Hot rod parts & accessories/	2015-06-30 06:40:03 (PDT)	2015-06-30 06:40:07 (PDT)	2015-06-30 06:40:23 (PDT)	2015-06-23 09:39:59 (PDT)	56	0
	r / r	HotRatRod.jpg	2015-06-29 11:35:49 (PDT)	2015-06-30 06:34:07 (PDT)	2015-06-29 11:35:49 (PDT)	2015-06-29 11:35:48 (PDT)	130585	0
✓	r / -	HotRatRod.jpg	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0	0
	r / r	HotRatRod.jpg:Zone.Identifier	2015-06-29 11:35:49 (PDT)	2015-06-30 06:34:07 (PDT)	2015-06-29 11:35:49 (PDT)	2015-06-29 11:35:48 (PDT)	26	0
	r / r	Lego Hot Rod.jpg	2015-06-25 11:11:12 (PDT)	2015-06-30 06:34:08 (PDT)	2015-06-25 11:11:12 (PDT)	2015-06-25 11:11:12 (PDT)	169488	0
	r / r	Lego Hot Rod.jpg:Zone.Identifier	2015-06-25 11:11:12 (PDT)	2015-06-30 06:34:08 (PDT)	2015-06-25 11:11:12 (PDT)	2015-06-25 11:11:12 (PDT)	26	0
	r / r	Little red wagon.jpg	2015-06-29 11:08:53 (PDT)	2015-06-30 06:34:08 (PDT)	2015-06-29 11:08:53 (PDT)	2015-06-29 11:08:52 (PDT)	53506	0
	r / r	Little red wagon.jpg:Zone.Identifier	2015-06-29 11:08:53 (PDT)	2015-06-30 06:34:08 (PDT)	2015-06-29 11:08:53 (PDT)	2015-06-29 11:08:52 (PDT)	26	0
	r / r	MilitaryHotRod.jpg	2015-06-29 11:22:57 (PDT)	2015-06-30 06:34:09 (PDT)	2015-06-29 11:22:57 (PDT)	2015-06-29 11:22:56 (PDT)	83754	0
	r / r	MilitaryHotRod.jpg:Zone.Identifier	2015-06-29	2015-06-30	2015-06-29	2015-06-29	26	0

ASCII (display - report) * Hex (display - report) * ASCII Strings (display - report) * Export * View * Add Note
File Type: JPEG image data, JFIF standard 1.01, comment: "CREATOR: gd-jpeg v1.0 (using IJG JPEG v62), quality = 90"

D:/Documents and Settings/Tom Payne/Desktop/All them cars!/Lego Hot Rod.jpg

localhost:9999/autopsy?mod=2&view=8&case=Test_Case_2&host=host1&inv=unknown&vol=vol3&meta=34598-128-4&sort=2&dir=Documents+and+Settings%2FTom+Payne...

Figure 4: Autopsy™ handling a jpeg image and displaying it



```
root@siftworkstation:~# foremost -o '/media/sansforensics/WD External/cases/Test
_Case_2/Foremost' -i /cases/Test_Case_2/Images/Item1*
Processing: /cases/Test_Case_2/Images/Item1.E01
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E02
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E03
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E04
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E05
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E06
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E07
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E08
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E09
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E10
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E11
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E12
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E13
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E14
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E15
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E16
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E17
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E18
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E19
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E20
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E21
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E22
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E23
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E24
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E25
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E26
|*******|
Processing: /cases/Test_Case_2/Images/Item1.E27
```

Figure 5: Foremost terminal printout for default functionality

```
[illegible]
foundat=liveupdt.grdeZen[illegible]
[illegible]
foundat=liveupdt.sig[illegible]
foundat=liveupdt.sig[illegible]
[illegible]
***|
Processing: /cases/Test_Case_2/Images/Item2.E67
|*|
root@siftworkstation:~#
```

Figure 6: Foremost processing an image file using the –t all option

Like EnCase® and FTK®, Autopsy™ supports use of Hash Databases such as the NIST NSRL database.

Virtualization Study

Using the Physical Disk Emulator (PDE) Module in Encase® Forensic 6.19.7.2, an evidence drive was successfully mounted. LiveView 0.7b could not be used due to network restrictions on forensic towers. Using command prompt and VirtualBox 5.0, a Test Case 2, Item2.vmdk file was created but was not able to fully boot. Item 1 from Test Case 2 was not tested for this method.

FTK® was able to successfully create a bootable virtual machine of Test Case 2 Item 1, but Item 2 was not tested. FTK® was able to create a vmdk for Test Case 1 items, but the resulting VMs were not bootable.

The SIFT workstation was able to create a roughly 85 GB VMDK file from Test Case 2 Item 1 using qemu-img. Oracle VirtualBox for Ubuntu 14.04 recognized the presence of the Item1.vmdk file, interesting it could not bring it in as the chosen virtual hard disk

Cost Analysis

Table 3 displays the cost of the following: software, training, certification, and supplemental requirements for each of the three tools tested. The data is based on a single license used by a single examiner.

Table 3: Forensic Software Cost Comparison (Loayza, Myers, Quinto)

Software Tool	Software Cost	Support/Maintenance	Certification Available	Training Cost	Certification Cost	Total Cost per examiner
EnCase® Forensic 6	$2,995	$599/year	EnCE®	$2195 each for Computer Forensics 1&2 online	$300	$8,284 with online courses

				$2750 each at EnCase training center		$9,394 with training center courses
FTK® 5.6	$3,995	$1,119/year	ACE®	$2,495 for 3 Day Bootcamp	$0	$7,609
				$2,495 for 3 Day Bootcamp and $2,495 for Windows Forensics for ACE Prep		$10,104
				$7,000 for 1 year of unlimited training		$12,114
SIFT Workstation 3.0	$0	$0	GCFE from GIAC	$5350 for FOR508 + shipping and handling of materials if applicable	$629	$5979 + s&h

Discussion

This study encountered numerous restrictions and setbacks of varying severity. The most cumbersome restriction was the requirement that all forensic towers not be networked. While it helps to protect the network from any malicious software that may be on an evidence drive, it inhibited some software such as LiveView 0.7b from being used due to the need of a network connection to download Java. The most recent version of Java was downloaded to a USB flash drive then deployed on the forensic tower. Despite having the installer file, Java also needed a network connection to fully install. This greatly hindered the virtualization study from achieving its full potential.

The next setback came in the form of FileVault encryption on the Mac Mini computers from Test Case 1. When the drives were imaged via FireWire target mode in EnCase® it was discovered that EnCase could not deal with the encryption in any way. It was not detected and all probative data was inaccessible. However, the sparse bundle files that are indicative of FileVault encryption were easily found in the file table. To remedy this, the Mac Mini computers were turned back on, logged into, and had FileVault turned off. The computers were then rehashed and reimaged. This is an issue in the field of digital forensics as examiners do not have the luxury of being able to alter their evidence and are increasingly encountering encrypted files, folders and even whole disks.

The most problematic setback of the study arose from the VM that ran the SIFT Workstation. It presented two issues of interest. The first was disk space. The images for each case were stored in /cases/Test_Case_#/Images. This path was linked to the 1TB virtual disk 2 which had plenty of space during each case. The problem was with virtual disk 1. When extracting strings to speed up keyword searching, the unallocated string extraction files could be upwards of 100GB. The string extraction files were stored on virtual disk 1 and caused the disk to fill rapidly.

The second issue with the VM was stability. As days went by in the work of processing Test Case 1, it became increasingly unstable and often returning multiple unexplained system errors and virtual disk failures. These caused spontaneous shutdown of the system. It is still unknown what caused these and how to fix them. To prevent loss of data, all Test Case 1 files were backed up on a 1TB Western Digital external hard drive before the VM was deleted and recreated for Test Case 2. Editing of the vmdk files from within VMware Player was attempted to make them larger, however these attempts were unsuccessful. One way to remediate the VM issue would be to install Ubuntu 14.04 on a large hard drive as the only OS present and use SANS' instructions on how to upgrade to SIFT Workstation. This would negate the need for a virtual machine and virtual disks.

In terms of case processing issues, there were very few overall. Each of the three tools possesses potential drawbacks that should be addressed by the software developers. EnCase® often takes an hour or more to open a case that has been started depending on how large the case file is. FTK® takes very little time to open regardless of what work has been done on a case as does Autopsy™ in the SIFT Workstation. A downside to FTK® is that it will completely lock a user out of a case if the hard drive it resides on changes drive letter, and the case cannot be unlocked. To remedy this, a new case using the same evidence image needed to be created. EnCase® deals with this easily by asking for the new path to the image files in the case. Based on how Autopsy™ opens a new case, this could be an issue as well but was not encountered during this study. While case issues were few, there were a few more data recovery issues that came up that can be explained.

The amount of data recovered by both EnCase® and FTK® fell in line with expectations with a few exceptions. The first difference that is noticeable when referring to Table 2 is unallocated pictures. For Test Case 1, EnCase® was able to recover unallocated pictures that were relevant to the case, namely those of a famous couple and a firearm used in the scenario. It also recovered hundreds of other unallocated pictures that were not relevant to the case such as images of icons, buttons, and the OS X user interface. Test Case 2 on the other hand was not able to find any relevant unallocated pictures. It did find scores of other unallocated images that were not case relevant. These were comprised of image files that come on the Windows XP OS by default. It is unknown as to why EnCase® was able to recover probative unallocated images from Test Case 1 but not Test Case 2. In regards to FTK® it does not handle unallocated images in the same way as EnCase®. While unallocated space is searchable, many of the files are only viewable as a hexadecimal or hexadecimal and text translation. The jpeg hexadecimal/text header “JFIF” was searched for, but no probative files were found in either test case. The next difference between the data recovered is in the email category. The suites were able to recover many of the same emails found on the E01 hard drive images. However upon reviewing the EnCase® and FTK® email reports from Test Case 1, it appears that EnCase® recovered copies of each email sent between the suspects but each evidence item had a nearly identical amount of probative email files. This is suspected because FTK® had roughly half of the number of recovered emails but the information in those emails is the same as those in EnCase®. For Test Case 2, Gmail, a popular webmail account was used. Due to the nature of webmail, none of the emails were stored on the evidence hard drive and therefore neither suite was able to recover any relevant information. The issue of recovered data in EnCase and FTK pertains to Link Files and Address Books. For the link files, while shortcuts to files and folder were made in Test Case 1, OS X refers to these as aliases. In EnCase® and FTK® these aliases are not readable and do not have a file signature or file extension and are not readable in the software. While these files perform the same function as link files in a Windows OS, they are technically not link files.

For the address book files in Test Case 1, the address books were created in the Mail app native to OS X. Upon analysis it was found that various files with extensions beginning with .abcd or abcd_.SQLite were files created for the address book. The abcdp files were bookmarked in EnCase, one for each item, while .abcddb and abcdmr files were manually carved as text fragments. In FTK it was known that abcdp, abcddb, abcdmr and their accompanying SQLite files were relevant to the address book and were bookmarked. In Test Case 2, the two address book files were .wab files found on Item 2. It is not known how or when these were created as Gmail does not create these files and Item 1 did not contain similar files.

Regarding file carving, while markedly different from EnCase® and FTK® manual requirements, Foremost is simple, quick, and highly customizable based on what evidence is being looked for. In an environment where an examiner is limited to his or her scope of search by what is on the search warrant from a submitting agency, a file carver that allows for specificity of file types is essential. While Foremost did recover many forensically relevant file types and files of interest, it did not recover everything that was known to be on the drive that was relevant to the test case. This was exhibited most in the jpeg pictures recovered and the documents.

While the SIFT Workstation is more than capable of imaging a drive and verifying its MD5 hash value, albeit in a more user intensive fashion using ewfacquire and ewfverify, its Autopsy™ HTML case processor leaves much to be desired. Keyword searching is far more user intensive as it must be done one word at a time and the extracted string files created to help speed up searching can be rather large. Its most damaging drawback is that it cannot, at present, verify E01/ewf images in a case. This may be a minor issue as the Ewfverify command works as it was intended. It also lacks a concise reporting feature that EnCase® and FTK® both have;

instead it allows a user to create a report for each relevant file which would create more work for an examiner. Both EnCase® and FTK® can deal with unallocated space well, but Autopsy™ relies on extracting unallocated strings and keyword searching to this. Unallocated string extraction creates rather large ASCII and Unicode files that consume large amounts of disk space. It was only feasible to extract unallocated strings from one item per test case in this research due to limited virtual hard disk 1 space.

Finally, there are two features of the SIFT Workstation that an examiner needs to be aware of. First, since SIFT is an open-source tool, a knowledgeable examiner could edit its source code once granted access to it after completing FOR508 (Arthur 2015). This could be problematic as an examiner could change the viability of the tool by doing this. Any changes made to the source code by an examiner would need to be validated to ensure that the workstation is still forensically sound. The second feature is that the SIFT workstation is far more user intensive than EnCase® and FTK® due to its heavy dependence on the Linux command line as used in the Terminal. Only an examiner who is highly skilled in the Linux Terminal should be examining cases using the SIFT Workstation.

As each digital evidence case varies greatly, the results presented in this research should be considered as representative for the capabilities of the tools in general. The cases were manufactured by the researchers for this study and as such, there were known types of data that would be recovered.

Conclusion

In light of all the results and discussion above, the SIFT Workstation is a viable alternative to tools such as EnCase® Forensic and FTK®. If an agency had a need for digital forensics capabilities but was on a tight budget and had personnel who were willing to learn, it could be used. Its Libewf- tools do a fine job at imaging a drive and verifying the resulting E01 file. Foremost is an exceptional file carver considering how simple it is, however it did not recover everything that was known to be present in the evidence files. Processing a case in Autopsy™ can be cumbersome and lacking in reasonably expected features. Compared to the industry standard tools, its ability to create a VM from an evidence file is more difficult than it needs to be. Despite its drawbacks, it is the least expensive of the three tools with which to bring one examiner fully on-line. It would be necessary for a laboratory to design highly specific standard operating procedures written by one or more individuals with at least above average skills in the Linux terminal if they wish to employ SIFT. Due to the open-source nature of SIFT, it should be used with caution because someone with the knowledge of editing its source code could alter its viability in a forensic environment.

Future Work

Should this research be continued, it is advised that future researchers implement Autopsy 3.x and EnCase 7.x as they are the most recent iterations of the software used. Autopsy 3.x is independent from SIFT and runs on Windows only. It is also recommended that SIFT be run as a standalone OS on a hard drive instead of being virtualized or use a custom vmdk instead of the two that come as part of the initial download. This may counteract some the instability and storage space issues encountered by this study. SIFT is still an active project and should be tested again in a similar way as newer versions are released.

Acknowledgements

- Reviewers
    - Robert Price, M.S., Agency Supervisor/Forensic Scientist I
    - Josh Brunty, M.S., Technical Assistant
    - Terry Fenger, Ph.D., MU Topic Advisor
- NCSCL Staff
    - Ben Smith, NCSCL Digital Evidence Intern
    - Ben Trotter, NCSCL Forensic Scientist II, Digital Evidence
    - Jim Trevillian, NCSCL Forensic Scientist I, Digital Evidence
    - Karen Morrow, NCSCL Forensic Science Manager, Digital/Latent Evidence
    - Katie Williams, former NCSCL Forensic Scientist I, Digital Evidence
- Software & Sales Personnel
    - John Loayza – Guidance Software Inc.
    - Megan Meyers – SANS
    - William Quinto – AccessData
- QEMU Team
- Preston Miller

References

1. Kröger K, Creutzburg R. A practical overview and comparison of certain commercial forensic software tools for processing large-scale digital investigations. Proc. SPIE 8755, Mobile Multimedia/Image Processing, Security, and Applications May 2013; 875519

2. Garfinkel SL. Digital forensics research: The next 10 years. Digital Investigation 2010; 7:64-73
3. https://www.guidancesoftware.com/products/Pages/encase-forensic/overview.aspx?cmpid=nav
4. http://accessdata.com/solutions/digital-forensics/forensic-toolkit-ftk
5. http://digital-forensics.sans.org/community/downloads
6. Hawthorne EK, Shumba RK. Teaching Digital Forensics and Cyber Investigations Online: Our Experiences. European Scientific Journal Sept 2014; Special (2): 255-261
7. http://forensicswiki.org/wiki/Virtual_machine
8. Lesson 14-EnCase® Physical Disk Emulator (PDE) Module. In: Guidance Software. EnCase® Computer Forensics II. Pasadena: 2014; 173-185
9. http://www.securityisfun.net/2014/06/booting-up-evidence-e01-image-using.html
10. http://www.nsrl.nist.gov/Downloads.htm