Building a Low Cost Forensics Workstation

GIAC Security Essentials (GSEC)
Practical Assignment
Version 1.4b
Option 1

Abstract

This paper will outline the fundamentals of computer forensic investigation and then, based on these essentials, create requirements for a low cost forensics workstation for use in electronic investigations.

Specific technologies will be used as examples of how these requirements can be met, however, the paper's intent is not to provide a cookie cutter solution, but rather provide a set of requirements that can be used to build a system with the resources available to most medium to large sized corporations.

Some legal issues are discussed in this paper, but are not addressed at a sufficient breadth or depth to provide adequate advice. It is recommended that proper legal advice be sought regarding the legalities of electronic investigation as it pertains to your specific country or region.

Electronic Investigations 101

One of the most important aspects of a successful electronic investigations program is establishing proper incident response procedures. Without these procedures, organizations run the risk of losing critical evidence, as well as jeopardizing criminal prosecution.

According to the book *Incident Response: Investigating Computer Crime*, a good incident response procedure can be broken down into eleven steps [1]:

1) Planning and preparation
2) Incident Detection
3) Initial response
4) Response strategy formulation
5) Forensic backups
6) Investigation
7) Security measure implementation
8) Network monitoring
9) Recovery
10) Reporting

11) Follow-up

Planning and Preparation

Incidents never happen at a convenient time. Because of this, it is important to prepare for an event before it occurs. Creating checklists is an easy way to ensure that all necessary evidence is collected and documented for the initial response, as well as any subsequent investigation.

The checklists should focus on the fundamentals of who, what, when, and where. For investigations that may lead to prosecution, the checklists should also include information regarding chain of custody and data integrity.

Based on the author's professional experience, a basic initial response checklist should look similar to this:

- Date and time of report
- Name of person reporting the incident and contact information
- What is the nature of the incident?
- How was the incident detected?
- What are the compromised or affected systems?
    - Hardware
    - OS name and version
    - Physical Location
    - Network information (IP Address, MAC address, AppleTalk name, dial-up phone number, other)
    - Contact names of support personnel
    - Business function and criticality of the affected system
- What actions have already been taken?
- Any possible further reaching impact of the incident?
- Any other considerations, such as legal, regulatory aspects of the incident

In addition to the checklists, having the correct tools and sufficient materials available is another fundamental of successful incident response. Tools and materials can be broken down into two categories: reusable and consumable.

The reusable components of a response toolkit primarily consist of hardware and software. Based on the size and complexity of your environment, you will need varying amounts of the following:

- Forensics duplication and analysis workstation supporting both IDE and SCSI for disk duplication and platform on which to run your forensics software
- Forensics software to perform analysis, gather & document evidence, and in some cases, perform duplication of drives

- Network sniffer to capture network traffic for use as an investigative tool, as well as evidence.
- Network cabling for use with forensic duplication devices and network sniffers
- Hubs for use with network sniffer and forensic duplication devices. The Hub provides and easy way to directly connect two or more computer without a crossover cable
- CD or DVD Burner, or other removable media to store and transport disk images and evidence
- High Capacity IDE and SCSI drives to store forensic images
- Different types of SCSI and IDE connectors and cables to ensure that you can connect as many drive types to your forensic duplication device
- Screwdriver set to take apart computer chassis to remove hard drives
- Boot disk with forensic tools for all hardware and OS combinations in your environment to provide a software based drive duplication solution if local duplication or physical access to the drive is not possible
- Tools disk with statically linked executable with basic OS and forensics applications for all hardware and OS combinations in your environment to provide a safe set of binaries to perform forensic analysis of a running system

Consumable aspects of a response toolkit can include:

- Blank DVD-R or CD-R media
- Evidence labels
- Permanent markers
- Evidence bags
- Note pads

Evidence Gathering

There are several avenues that can be followed to collect evidence to support an electronic investigation, but for the purposes of this paper, we will focus exclusively on collecting evidence from computer hard drives.

In a forensic examination, an investigator must meet the following requirements: preservation of evidence, lead formulation, data searches, recreation of timelines and evidence recovery. [2]

Each operating system and investigation is unique. However, to meet the core investigative requirements, there is key information that should always be gathered and preserved:

- *MAC Times:* Modify, Access, and Change times of a file
- *Deleted files:* Files removed from the operating system's file structure
- *File system structure*: The system that an operating system or program uses to organize and keep track of files [3]
- *MD5 hashes*: A one-way hash function which can be can be used to verify that a file has not been altered
- *Content:* allocated space in a file system
- *Slack Space*: The unused space in a file system

Forensic Duplication

The first step in collecting electronic evidence from a hard drive is creating a forensically sound duplicate image. This is accomplished by duplicating, bit for bit, the entire hard drive.

There are two basic approaches for creating binary images of hard drives: hardware and software based duplication.

The two main advantages of using a hardware based duplications system are speed and data integrity. With a hardware-based solution, there is also less risk that the data contained on the evidence hard drive will be modified or corrupted. An example of a hardware based duplication system is LOGICUBE® SF-5000. Logicube claims that under certain conditions, the SF-5000 can copy drives and speed in excess of 1.6MB per minute. [4] The main disadvantage of using a hardware-based solution is that they require physical access to the computer, and in most cases physical access to the hard drive itself.

If physical access to the computer is not possible, a software-based solution must be used. For most systems, a simple utility created by GNU call *dd* does the trick. This application provides a great deal of flexibility in duplication. Copying can occur from device to device, device to file, and file to device. [5] The other advantage of *dd*, when used in conjunction with *netcat*, is the ability to duplicate a hard drive to, or from, a remote computer. [6] To perform this, a *netcat* session must be established on both the local and remote host. This can be accomplished in a manor similar to this:

```
Forensics (10.100.0.140)% nc -l -p 37337 | dd of=/dev/hda
```

```
Evidence% dd if=/dev/hdb | nc 10.100.0.140 37337
```

The primary disadvantage of using a software based duplication approach is the risk of modifying the original hard drive. Although some operating systems are better than others about modifying data when mounting drives, measures must be taken to ensure that any modification to the original media is detected and documented.

The Requirements

After reviewing the basics of forensic duplication and related investigation techniques, we can now outline some core requirement for building a forensics workstation:

1. The system must support IDE
2. The system must support SCSI
3. The system must have network connectivity
4. The system must support hardware based drive duplication
5. The system must support remote and network based drive duplication
6. The system must support duplication and analysis of these common file system types:
    a. NTFS
    b. FAT16/32
    c. Solaris UFS
    d. BSD UFS
    e. EXT2 (Linux)
    f. EXT3 (Linux)
    g. HFS & HFS+ (Macintosh)
    h. Swap
        i. Solaris
        ii. BSD
        iii. Linux
7. The system must have the ability to validate image and file integrity
8. The system must be able to identify dates and times that files have been modified, accessed and created
9. System must have the ability to create file system activity timelines
10. The system must be able to identify deleted files
11. The system must be able to analyze allocated drive space
12. The system must be able to isolate and analyze unallocated drive space
13. The system must allow the investigator to directly associate disk images and evidence to a case
14. The system must allow the investigator to associate notes to cases and specific evidence
15. The system must support removable media for storage and transportation of evidence and disk images
16. Evidence collected by the system must be admissible in a court of law

Building the workstation

In picking software and hardware for building a forensic workstation, you will want to accommodate as many of the investigative requirements and duplication techniques as possible. Doing so will allow an investigator to react to most common situations in a timely manor. That being said, even with proper preparation, it is impossible to predict all possible scenarios, such as unusual physical locations and outdated hardware.

Hardware and Software Choices

To build an example workstation, the following hardware configuration was used:

Tower chassis with 4 drive bays
Two-processor motherboard with support for IDE, SCSI and USB
2 200 mhz Pentium processors
3 removable drive bays
External USB hard drive
10/100 Ethernet Card
CD Writer

With the exception of the removable drive bays, comparable components are on hand in most corporate environments.

Although many commercial forensic software packages are available that provide better user interfaces, and easier setup, @Stake Sleuth Kit (TASK) with the Autopsy Forensics Browser (AFB) seemed to be the obvious choice from a functionality and cost perspective.

According to the @stake web site, the combination of TASK and AFB provide the following abilities: [7]

- View Allocated and Deleted Files and Directories
- Access to low-level file system structures
- Keyword searches including grep regular expressions
- Timeline of file activity
- File category sorting and extension checking
- Investigator notes
- Report generation
- Analyzes file system images generated by the 'dd' command, which is found on all UNIX systems and is available for Windows systems.
- Supports the NTFS, FAT, FFS, EXT2FS, and EXT3FS file systems
- Displays the details and contents of all attributes for NTFS files. This includes all Alternate Data Streams.
- Creates timelines of file activity and can import logs and other time-based events

- Time-based tools take a time zone and time skew as arguments so that you can view times as they existed on the original host.
- File[s] can be organized based on their file type. For example, all graphic images and/or executables can be easily identified and examined. While they are being sorted, hash databases can be consulted to ignore known files (such as system files that are trusted) and to alert when known bad files are found (such as known rootkits or inappropriate photographs). The extensions of files are also verified to identify files that are being hidden.

In addition to its robust functionality, TASK and Autopsy Forensic Browser have been tested and run on the following operating systems:

- Linux
- Mac OS X[*]
- Open & FreeBSD
- Solaris

Redhat Linux was chosen as the underlying OS because of its known compatibility for the chosen hardware, it's out of the box functionality, and its support in many corporate environments.

Assembling the Hardware

If you don't have experience as a PC technician, assembling the equipment can prove to be quit a challenge. Specifically, if you are inexperience with resolving hardware conflicts on the Window/Intel platform, it is recommended that you have a skilled professional put the system together, or even purchase a pre-built system to avoid many hours of unnecessary frustration.

Platforms from Sun and Apple may ease some of the hardware issues, but are generally not supported by commercial forensics software such as EnCase.

No special preparation is needed for installing the underlying OS, however, it is important to keep the system patched, as well as disabling all unnecessary services. It is also recommended that the forensics workstation be kept on a standalone LAN, although this is not always possible.

Installing and Configuring TASK

[*] The author tested Autopsy Forensic Browser on Mac OS X 10.2.4 and found that the installed version of the Unix utility *strings* did not accept some of the flag set by AFB

If you are installing TASK and Autopsy Forensic Browser on a supported system, the installation procedure is fairly simple. However, there are two critical things to watch out for:

- When selecting your Evidence Locker directory for the AFB setup, make sure that the partition has enough space to support both the evidence collected, as well as the disk image files
- Make sure you configure and compile the application in the directory you intend to run it from, or the paths to the AFB and TASK executables will become out of sync if the folder is later moved.

TASK and Autopsy Forensic Browser can be found at the following locations:

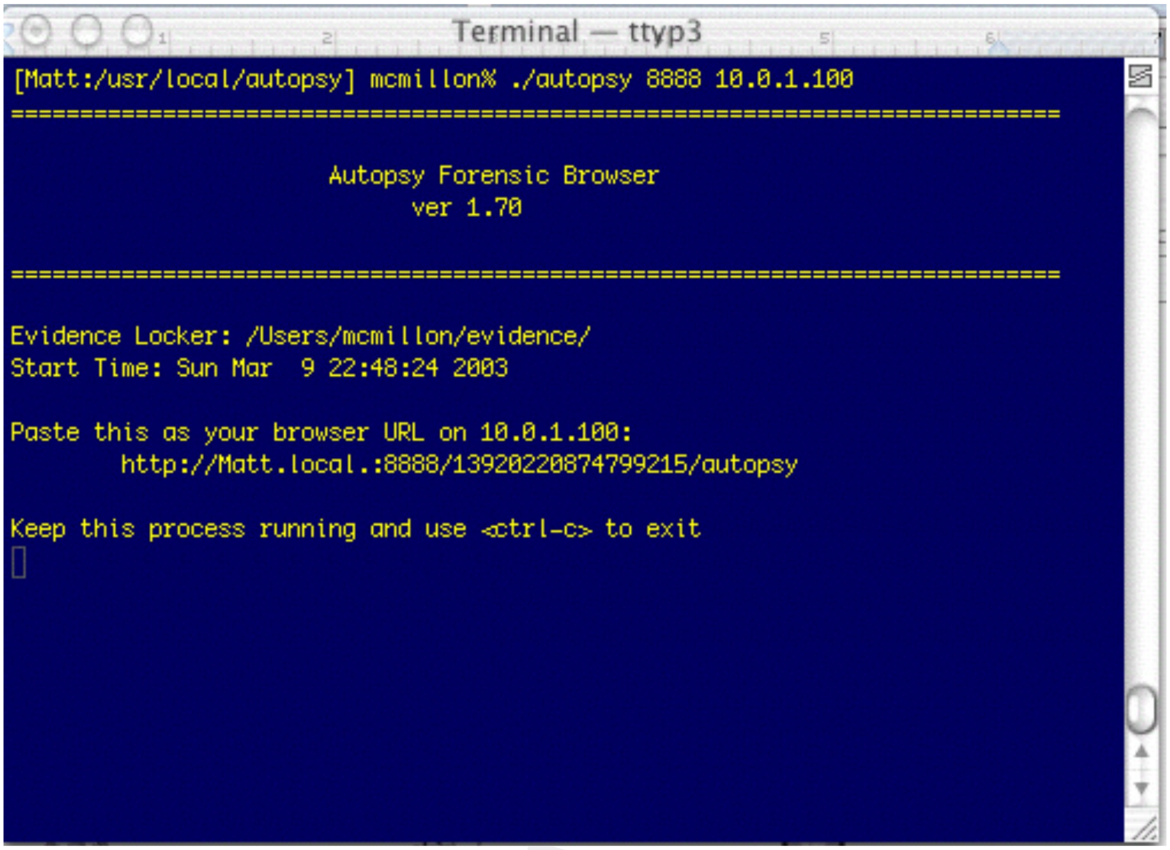
Task: http://prdownloads.sourceforge.net/sleuthkit/task-1.60.tar.gz?download

Autopsy Forensic Browser:
http://prdownloads.sourceforge.net/autopsy/autopsy-1.70.tar.gz?download

Starting a Forensic Examination

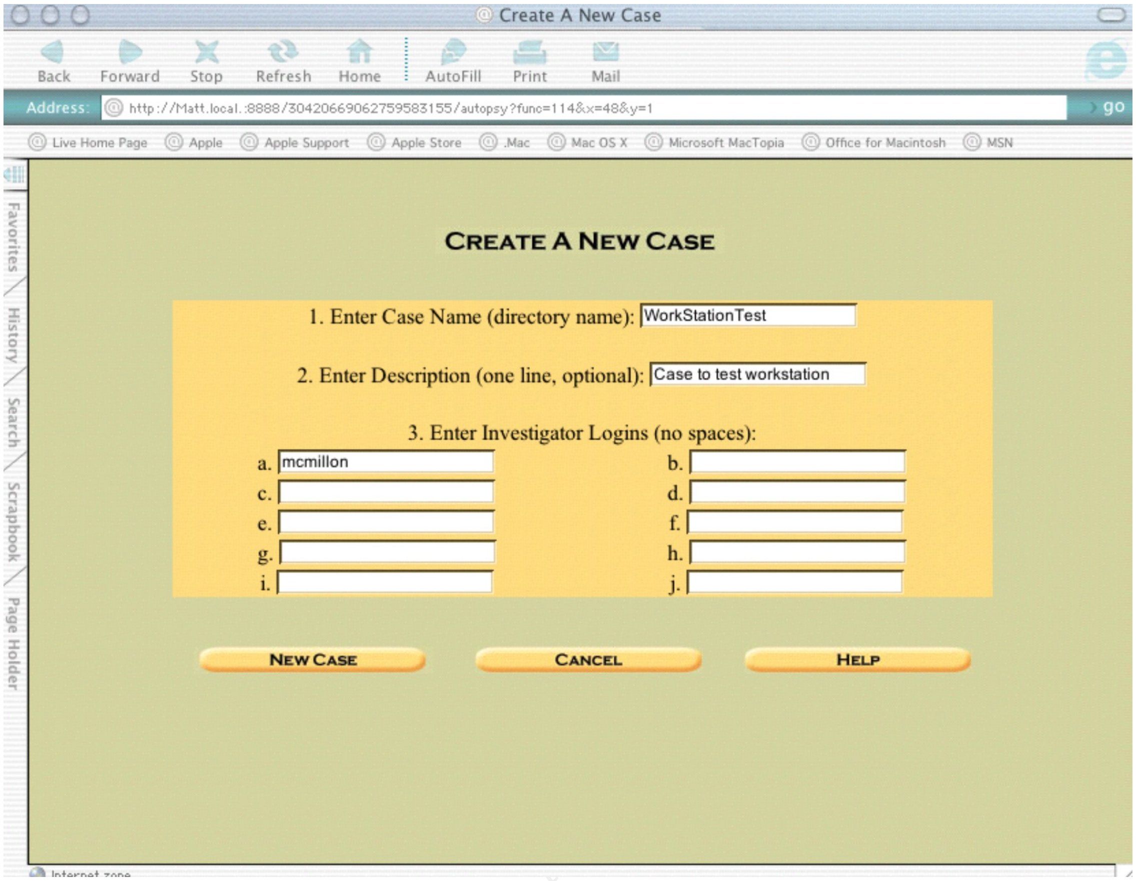
Once you have TASK and AFB successfully installed, you can start your first examination. To test the system and validate requirements, sample image files were downloaded from The Honey Net Project's Forensic Challenge. [8] This was done because the images were created using *dd*, and the information contained in the images are already publicly available and analyzed.

AFP is browser based and is started from the command-line with two arguments: port and hostname.



```
[Matt:/usr/local/autopsy] mcmillon% ./autopsy 8888 10.0.1.100
============================================================================

                       Autopsy Forensic Browser
                             ver 1.70

============================================================================

Evidence Locker: /Users/mcmillon/evidence/
Start Time: Sun Mar  9 22:48:24 2003

Paste this as your browser URL on 10.0.1.100:
        http://Matt.local.:8888/13920220874799215/autopsy

Keep this process running and use <ctrl-c> to exit
```

Some operating system and browser combinations do not resolve 127.0.0.1 or localhost properly, so, if you are connecting from the workstation itself, it is best to use its fully qualified domain name or IP address as the host argument
.
The first thing you want to do is create a new case. One nice feature of AFB is the ability to assign multiple investigators to a single case.



CREATE A NEW CASE

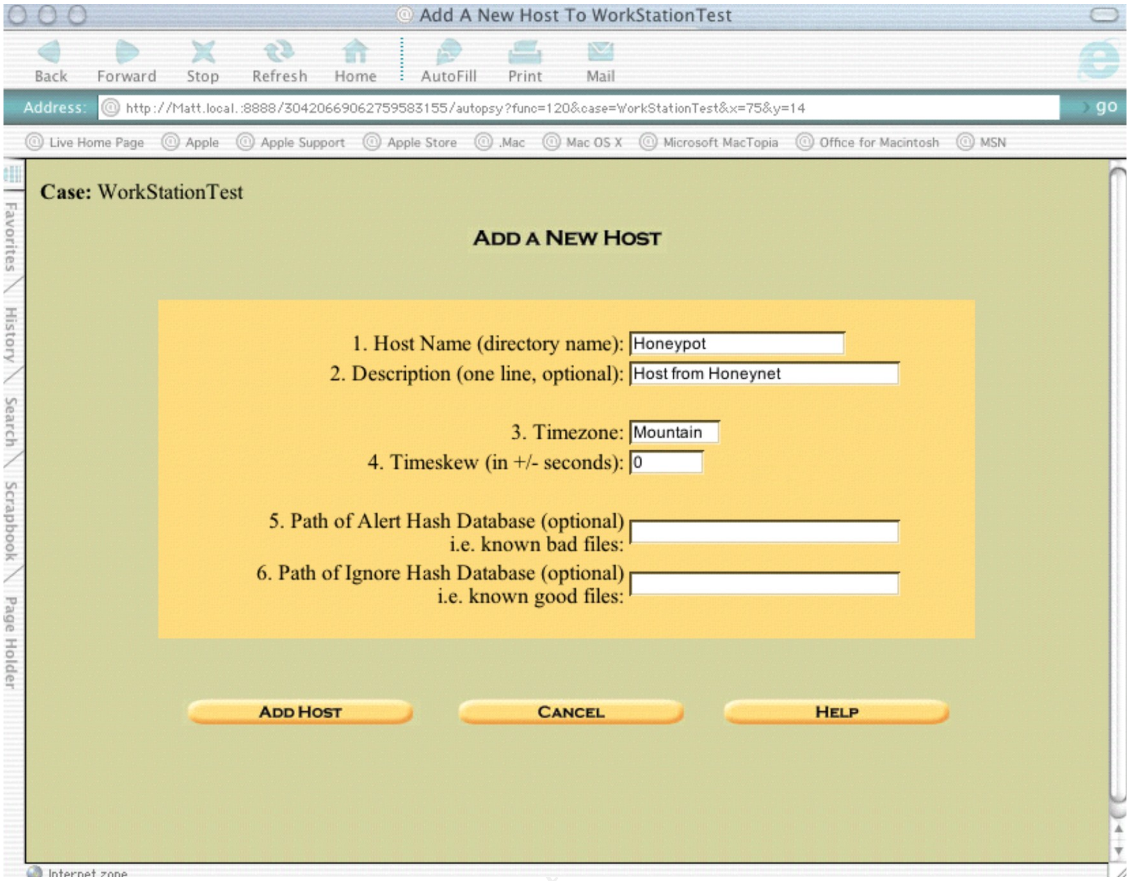
1. Enter Case Name (directory name): WorkStationTest

2. Enter Description (one line, optional): Case to test workstation

3. Enter Investigator Logins (no spaces):

a.	mcmillon	b.	
c.		d.	
e.		f.	
g.		h.	
i.		j.	

The next step is adding hosts to the case. AFB allows multiple hosts to be associated with an investigation. Some legwork upfront will help to determine the time zone and time skew for a given host. This information is not easy to gather from the disk images, and is difficult to change once the host is associated with a case.



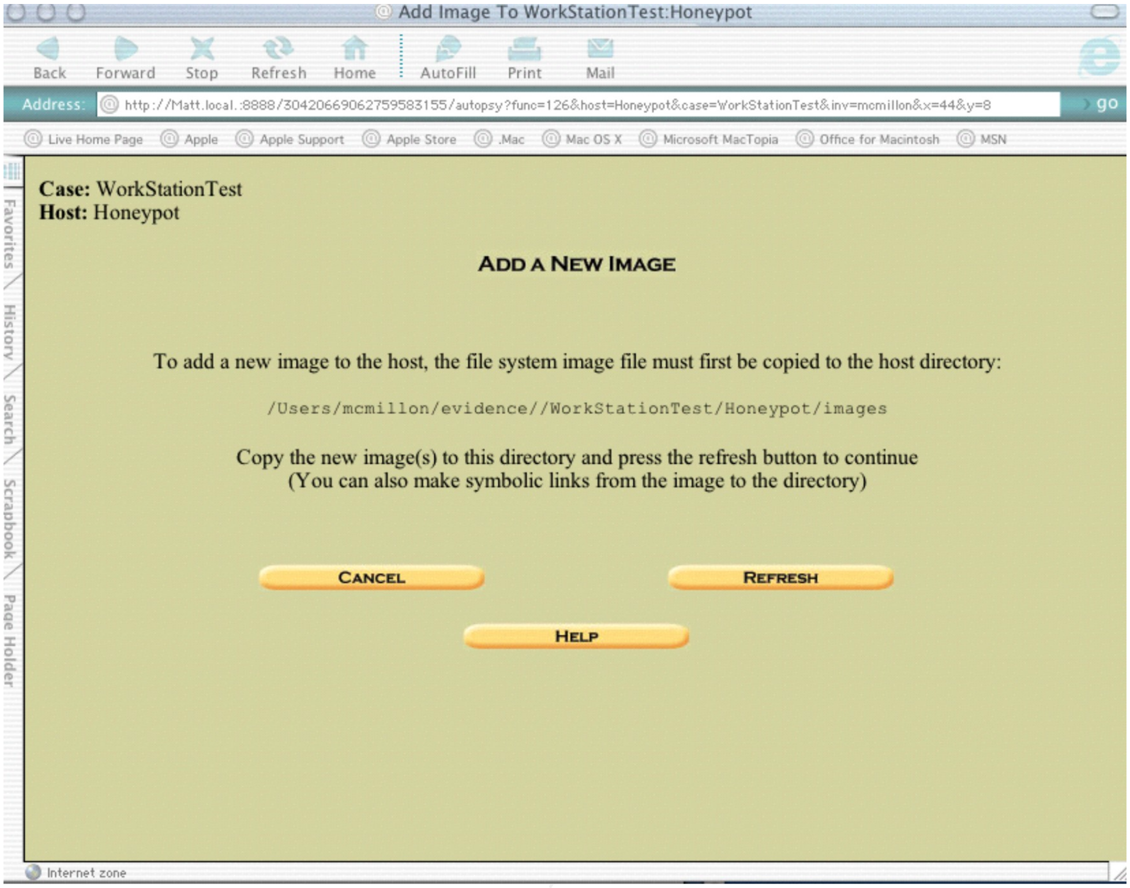
http://Matt.local.:8888/30420669062759583155/autopsy?func=120&case=WorkStationTest&x=75&y=14

Case: WorkStationTest

ADD A NEW HOST

1. Host Name (directory name): Honeypot
2. Description (one line, optional): Host from Honeynet
3. Timezone: Mountain
4. Timeskew (in +/- seconds): 0
5. Path of Alert Hash Database (optional) i.e. known bad files:
6. Path of Ignore Hash Database (optional) i.e. known good files:

The next step is copying the disk images into the host directory. This is where it becomes important to verify that the disk or partition where the evidence locker is located has sufficient space to hold both the images and files produced by TASK and AFB.



http://Matt.local.:8888/30420669062759583155/autopsy?func=126&host=Honeypot&case=WorkStationTest&inv=mcmillon&x=44&y=8

Case: WorkStationTest
Host: Honeypot

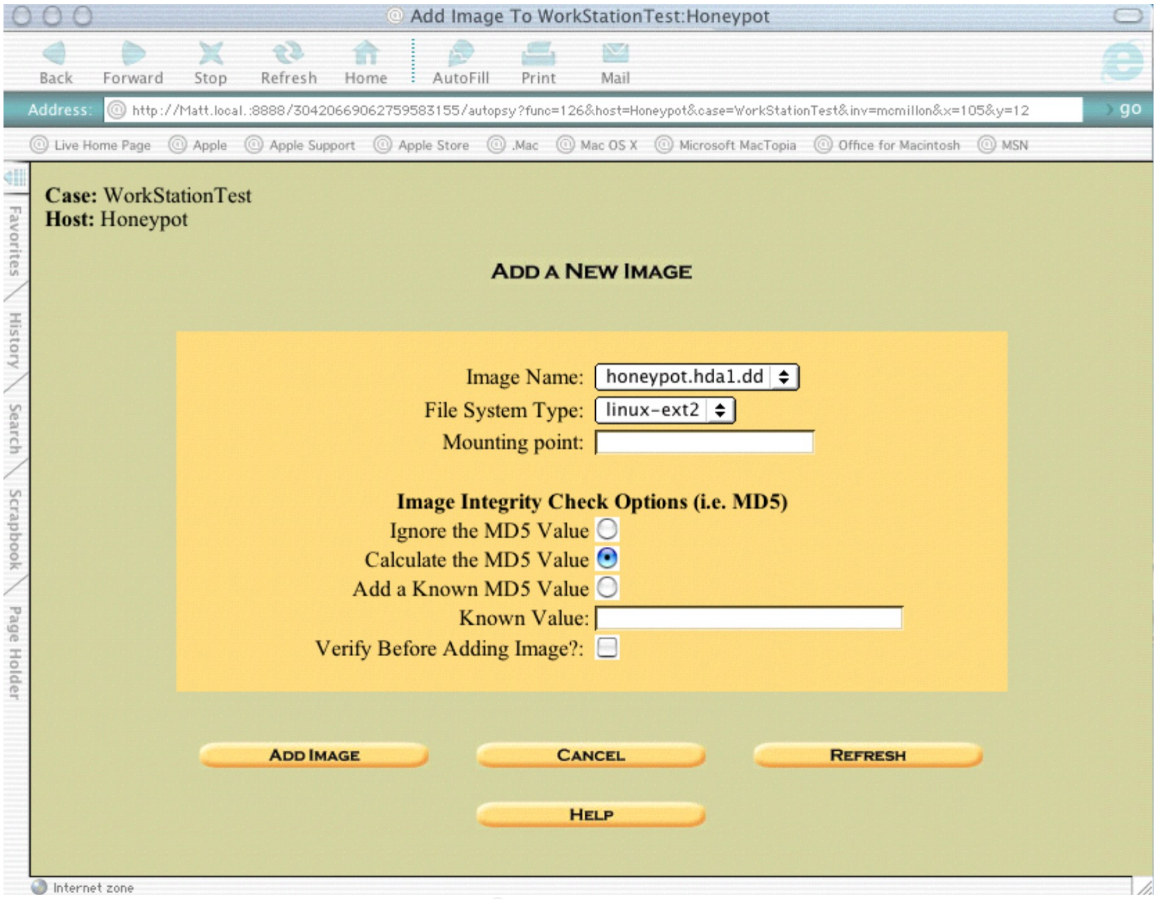
ADD A NEW IMAGE

To add a new image to the host, the file system image file must first be copied to the host directory:

```
/Users/mcmillon/evidence//WorkStationTest/Honeypot/images
```

Copy the new image(s) to this directory and press the refresh button to continue
(You can also make symbolic links from the image to the directory)

Once you have copied the files into the directory, select refresh and the images are loaded into AFB.



http://Matt.local.:8888/3042066906275958З155/autopsy?func=126&host=Honeypot&case=WorkStationTest&inv=mcmillon&x=105&y=12

Case: WorkStationTest
Host: Honeypot

ADD A NEW IMAGE

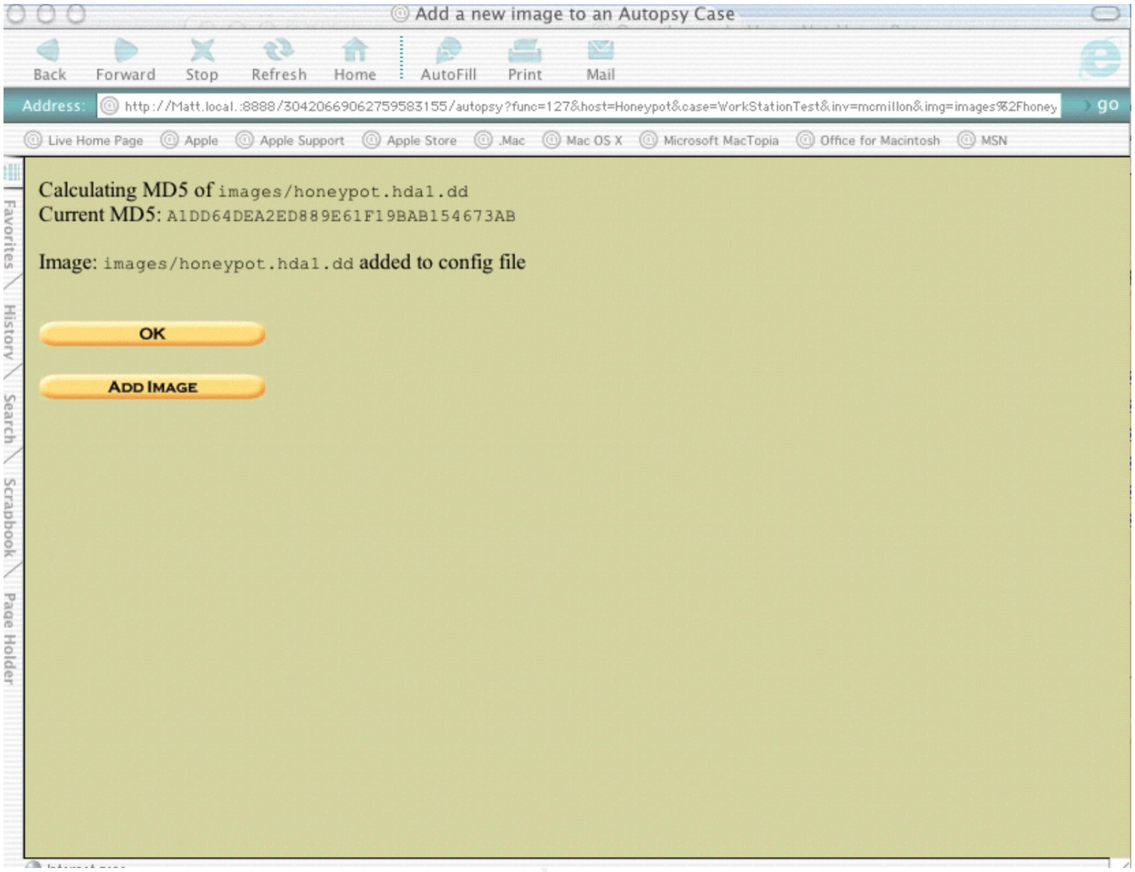
Image Name:	honeypot.hda1.dd
File System Type:	linux-ext2
Mounting point:	

Image Integrity Check Options (i.e. MD5)

Ignore the MD5 Value	○
Calculate the MD5 Value	●
Add a Known MD5 Value	○
Known Value:	
Verify Before Adding Image?:	□

AFB does not have the ability to detect the file system type, so you have to select the file system, as well as manually enter the mount point. One shortcoming of AFB is that it provides no support for swap partitions or HFS/HFS+. For most investigations, this would not prove to be a serious problem, although it could prove to be an issue under certain circumstances.

Once the image is assigned a file system type and mount point, an MD5 hash is created and the image is associated with a host and saved in the case file.



Address: http://Matt.local.:8888/3042066906275958 3155/autopsy?func=127&host=Honeypot&case=WorkStationTest&inv=mcmillon&img=images%2Fhoney

Calculating MD5 of `images/honeypot.hda1.dd`
Current MD5: `A1DD64DEA2ED889E61F19BAB154673AB`

Image: `images/honeypot.hda1.dd` added to config file

System Performance

From a hardware perspective, the workstation performed adequately, however, due to a relatively slow processor configuration, certain actions, such as creating MD5 hashes, slowed the investigation process significantly.

One of the major limitations of TASK and AFB is its file recovery capabilities. TASK does provide functionality to extract unallocated space from a partition, but the analysis is limited to string search, which can be tedious, and unfruitful for the novice user—in other words, you have to know what you are looking for to find it.

The other drawback is the lack of support for swap file systems. For most users and investigations this is not a major concern, however, the advanced user may find the absence of this functionality a bit troublesome.

Mapping the Requirements

The system, as built, met most the core requirements with these notable exceptions:

1) The system did not support hardware-based duplication of hard drives.

2) The system did not support swap, or HFS/HFS+
3) For beginners, the system had limited ability to analyze unallocated drive space
4) The evidence produced by the system may not be universally excepted in a court of law
5) The system had limited ability to detect changes to the file system contained on the original hard drive

To meet the requirements of remote duplication, it is necessary to create either a boot disk with *dd* and *netcat*, or a tools disk containing the same applications but compiled with statically linked libraries. This will allow duplication of both a running machine and an off-line machine.

Conclusion

For under $150, it is very possible to build an effective forensics workstation from parts commonly found in a corporate environment. However, the effective use of such a system to create admissible evidence for legal proceedings is still in debate.

The court systems in the United States hold a pre-trial hearing to determine if the scientific evidence to be presented in the trial has been gathered with techniques and methodologies that are fundamentally sound, and produce reliable results. This pretrial hearing, know as a Daubert hearing, uses four general guidelines to evaluate the evidence gather procedure [9]:

1. Can the procedure be tested?
2. Is there a known error rate for the procedure?
3. Has the procedure been published and subject to peer review?
4. Is the procedure generally accepted in the relevant scientific community?

Based on these criteria, @Stake argues that an open source tool could pass the evaluation guidelines because of the inherent openness of its nature, as well as the stringent peer review of the open source development process. The tool used to image the disks for this paper—*dd*—has been subjected to a review by the Computer Forensics Tool Testing (CFTT) project [10], which gives it a leg up on most commercial software, which have relied primarily on tests published by SC magazine. [9]

The bottom line is, if you are an experience intrusion analyst, building a low cost forensics workstation is a viable option. Beginners, however, should consider using commercial software and pre-built hardware to ensure success.

References

1. Mandia & Prosise. Incident Response: Investigating Computer Crime, Osborne/McGraw-Hill 2001, p. 16-17
2. Eoghan, Casey (Editor), Handbook of Computer Crime Investigation: Forensic Tools and Technology, Academic Press 2002, p. 133-167
3. Webopedia.com, "file management system " 8 July 2002 URL: http://www.webopedia.com/TERM/f/file_management_system.html
4. Logicube, Inc., "Foresic SF-5000" URL: http://www.logicube.com/store/forensic_features.html
5. GNU.org, "GNU file utilities: dd invocation",3 May 2002 URL: http://www.gnu.org/manual/fileutils-4.1/html_node/fileutils_39.html
6. Kumar, Rajeev "Wonders of 'dd' and 'netcat' :: Cloning Operating Systems", 30 August2001 URL: http://www.rajeevnet.com/hacks_hints/os_clone/os_cloning.html
7. @Stake, "@stake Research Tools: TASK", URL: http://www.atstake.com/research/tools/task/
8. Honeynet Project, "Forensics Challenge", URL: http://www.honeynet.org/challenge/
9. Carrier,Brian, Open Source Digital Forensics Tools: The Legal Argument., 2002, Page 3, URL: http://www.atstake.com/research/reports/acrobat/atstake_opensource_forensics.pdf
10. National Institute of Justice, "Publications and Products ", URL: http://www.ojp.usdoj.gov/nij/pubs-sum/196352.htm