Digital Forensics on the Cheap: Teaching Forensics Using Open Source Tools

Richard D. Austin
Kennesaw State University
4756 Jamerson Forest Circle
Marietta, GA 30066
+1 404 630 7515
austin_r@bellsouth.net

ABSTRACT
Digital forensics capability is a critical function in any information security program but the cost of the necessary technology and tools can be a barrier to its presentation in the undergraduate curriculum. This paper discusses how Open Source tools can be used to provide students with a realistic introduction to and experience in the digital forensics process.

Categories and Subject Descriptors
K.5.0 [**Legal Aspects of Computing**]: General

General Terms
Security, Legal Aspects

Keywords
Digital Forensics

1. INTRODUCTION
It is a given that we live in a very litigatious society where issues ranging from corporate governance (SOX, GLBA) and privacy (HIPAA, CA SB1386) to workplace behavior (sexual harassment, child pornography) can involve an organization in legal proceedings where digital information will play an important role in deciding the facts of the matter. The collection, analysis and presentation of digital information for use in the legal system is the province of digital forensics. Formerly the term computer forensics was widely used but with the critical relevance of network and other sources of information, the more inclusive "digital forensics" is the better term.

With digital forensics being a core capability within an organization's information security function, it is natural that students should be exposed to the process and practice of digital forensics as part of the curriculum. However, teaching digital forensics can be an expensive proposition due to the specialized tools and equipment normally regarded as "tools of the trade." This paper will explore the use of Open Source tools to provide this introductory experience with minimal requirements for specialized hardware and software beyond that usually found in an academic computer lab.

Information Security Curriculum Development Conference '07, September 28-29, 2007, Kennesaw, Georgia, USA.

2. THE FORENSICS PROCESS
There are many models of the forensics process such as Casey's staircase (2004) but I prefer the simplified model shown below.

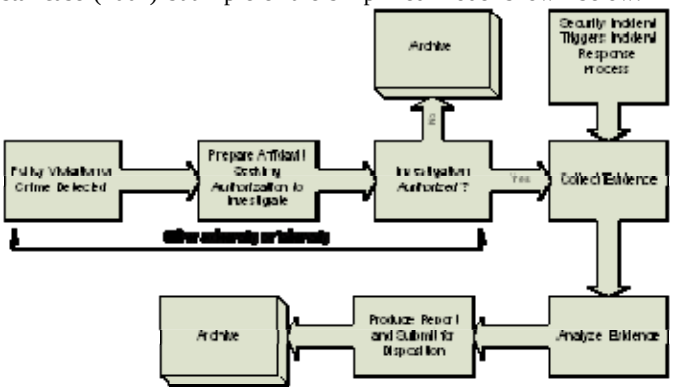

Figure 1 Overall Forensics Process

An investigation can be triggered either by an allegation of wrongdoing or by the detection of a security incident. After an investigation is authorized, the next step is to identify and collect all relevant evidence[1] in a fashion that will preserve its ability to serve as evidence in legal proceedings. After the evidence is collected, it is then analyzed to produce a report with findings and recommendations.

This model suggests the following tasks which students might complete during a digital forensics course:

1. Prepare an affidavit based on allegations of wrongdoing or a policy violations
2. Assess a scene to identify sources of relevant evidence and collect that evidence in a forensically sound fashion
3. Analyze the collected evidence to support or dismiss the allegations
4. Prepare a report to present the findings

[1] As noted by Brown (2006), an item does not become "evidence" until it is admitted as such in a court of law after meeting any challenges from opposing counsel. I follow common usage of "evidence" as a synonym for "item of potential evidentiary value."

This paper will concentrate on Open Source tools that can be used during steps 2 and 3.

2.1 What is "Evidence?"

Evidence is "the stuff of proof – manifesting truth about particular facts or circumstance" (Nemeth, 2001, p. 1) and the legal system imposes three basic requirements that must be met by items before they can be introduced into a legal proceeding:

1. Relevant – it must serve an important purpose in deciding a question of fact
2. Authentic – it must be the "real thing."
3. Integrity – it must not have been modified or contaminated at any point during its collection or subsequent processing

Relevancy basically is a criterion of efficiency and seeks to prevent time being wasted on peripheral matters that have little relevance to the matter under consideration. This criterion plays a major role during the assessment of the scene to determine what sources of information are likely to be most relevant to the matter under investigation. During analysis it helps to pare down the wealth of information to only those items that bear on the questions being investigated.

Authenticity and integrity present major challenges in the world of digital evidence where the contents of a disk sector, network log or a server's memory are highly volatile and are easily changed. Good process, documentation and proper technical controls function together to create credible assurances of authenticity and integrity.

3. Open Source Tools

It is a common saying that when dealing with a potentially compromised system, the wise investigator will trust nothing she didn't bring with her. This has created the need for collections of useful tools installed on a bootable CD which can be brought to the scene and used during the investigation without need for additional software. The particular CD that I will discuss is Helix, freely downloadable at www.e-fense.com/helix. While there are many other examples that could be used (F.I.R.E., Knoppix STD, etc), Helix is a well known tool that is optimized for forensic use. An introductory manual for Helix is available at http://www.e-fense.com/helix/Docs/Helix0307.pdf

3.1 Disk Imaging

Making a forensic copy of a disk for later analysis is the "bread-and-butter" of traditional forensic practice. The requirements for forensic imaging are that the source (or suspect) disk not be modified in any way by the process and that the copy contains an exact block by block copy of the original (NIST CFTT, 2001).

The second requirement means that the copy must include not only files but deleted files, free space, etc, as shown below.

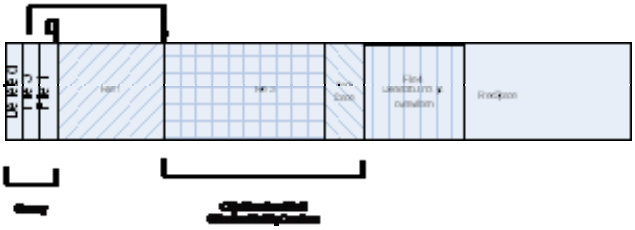

Figure 2 Forensic Imaging

A normal copy of this disk would only copy the two files (File 1 and File 3) and a utility that "undeleted files" would also recover File 4 but at the cost of modifying the disk. A forensic copy (or image) that copied each block of the disk would include the directory, the files, the fragment of File 2 that had not been overwritten, File 4 and the contents of the free space.

The authenticity of digital information is assured by both documentation of sound process and a digital fingerprint (Rivest, 1992) provided by a cryptographic hash, commonly MD5. The MD5 algorithm accepts an input of any size and produces a 128 bit value with two important properties (ibid):

- It is computationally infeasible to produce two messages with the same hash value
- It is computationally infeasible to produce an object with a predetermined hash value

While there have been a number of recent findings(e.g., Wang & Yu, 2005) that raise serious questions about MD5's fulfillment of these two properties and at least one legal case that hinged on these questions (theage.com.au, 2005), MD5 is still widely used in authenticating digital evidence (AccessData, 2006).

For purposes of authenticating a disk image, the general process is to calculate a pre-image hash of the source disk, create the disk image and then create a post-image hash. The pre-image hash establishes a point-in-time reference that uniquely identifies that particular piece of evidence while the post-image hash demonstrates that the imaging process produced a true and accurate copy of the original. At any time during the history of the image, the hash can be recalculated and shown to match the pre-image hash. In this way, the properties of authenticity (it is the image made of the disk from John Doe's computer on 1July2007) and integrity (the image has not been modified since its collection) can be demonstrated.

3.2 Imaging Using Helix

A field imaging setup is shown in Figure 3 where the suspect drive has been removed from its system and attached to the laptop through a hardware write blocker that will prevent any writes to the suspect drive (it also supplies power to the drive and bridges its IDE interface to USB for connection to the host). The forensic target disk is installed in the USB enclosure on the right side of the photograph.

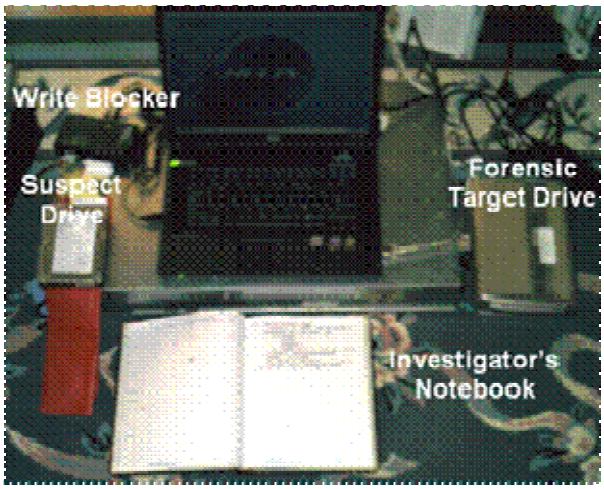

Figure 3 Field Imaging Setup

However, as hardware write blockers cost several hundred dollars, they may be out of reach for many programs. As Helix is optimized for forensic use it treats all devices as read-only by default (with the exception of a ramdisk filesystem to provide temporary storage).

> *Helix provides no protection whatsoever to prevent its user from "shooting themselves in the foot" and performing commands that will modify the suspect disk.*

To further reduce the hardware requirements for teaching digital forensics, a virtualization platform such as vmware or Microsoft VirtualPC can be used to eliminate the need for physical devices to use as suspect drives and forensic targets. The following walkthrough illustrates the use of Helix running on Microsoft Virtual PC.

After booting from the Helix CD, the graphical console appears as shown below.

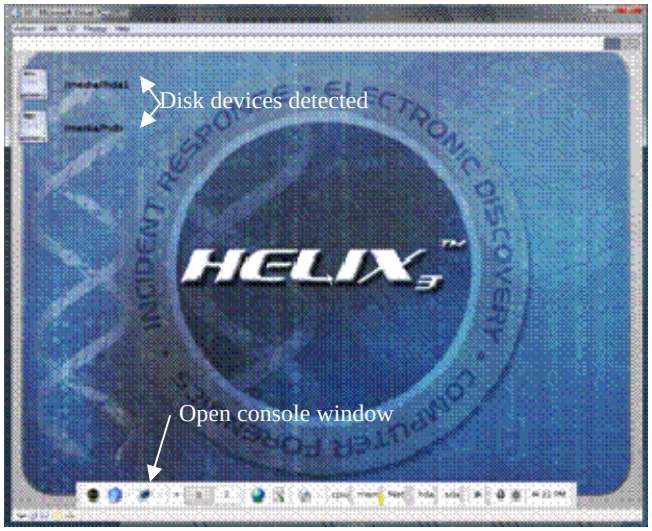

Figure 4 Helix Main Screen

In the upper left, any disk devices (IDE/SATA/USB/SCSI etc) are displayed – in this case two IDE drives. The first step in imaging is to assure that one can reliably identify the source and target drives and this is easily done using the `fdisk -l` command. A command console window is opened by clicking on the "Open console window" icon

Figure 5 Helix Console Window

In this example, the suspect disk is taken from a Windows XP desktop system and Helix has identified it as /dev/hda (note the HPFS/NTFS under the "System" heading). The target drive is a 32GB drive identified as /dev/hdb and formatted with a Linux filesystem.

In times past, the normal practice was to do drive-to-drive imaging but with the proliferation of drive sizes, it is more efficient to image the suspect drive to a file on the target drive. This has the added benefit of allowing hash values, etc, to be stored on the same medium as the image(s). Since Helix does not mount a filesystem on disks by default, the next step is to mount a writeable filesystem on the target drive using the following commands:

```
mkdir /mnt/target
mount /dev/hdb1 /mnt/target
```

First a directory is created as a mount point (/mnt/target) and then the first partition of the target drive is mounted there.

> *Note that this operation does change the contents of the mounted drive – if an unwary user were to perform these commands on the suspect drive, its contents would be modified and its evidentiary usefulness would be destroyed.*

Digital investigations tend to produce a lot of evidence and tracking each piece can become a confusing endeavor unless some simple process conventions are followed. One of the most common is a simple naming convention for evidence items where items are identified by a type prefix, the case number followed by an item number. Assuming the case number for this investigation were 2007-0014, the conventional names having to do with disk imaging the first item of evidence are shown in Table 1:

Table 1Evidence Naming Convention

Name	Contents
PR2007-0014-001.md5	**PR**e-image hash of item 1
DI2007-0014-001.img	**D**isk **I**mage of item 1
PO2007-0014-001.md5	**PO**st image hash

Each organization will have its own conventions and process standards but these work well in a classroom setting.

The first step in creating a forensic copy (or image) is to calculate the baseline or pre-image hash[2] using the md5sum command. The output of the hash is stored in a file on the target drive using the first command shown below:

Figure 6 Pre-Image Hash and Imaging

The second command produces the actual block-by-block copy of the suspect disk. There are many utilities to perform this function but the venerable dd command is quite effective. In its most basic form, the dd command accepts two parameters:

1. The input file specified in this case as /dev/hda
2. The output file which is /mnt/target/DI2007-0014-001.img

Other parameters are optional but it is common to specify a blocksize with the bs= parameter (8192 in this case) to improve performance.

The final steps are to compute a hash of the image and then to compare it to the pre-image hash to verify that a true and accurate copy has been made.

Figure 7 Post Image Hash and Comparison

4. Analysis Using Helix

One of the challenges in introducing students to forensic analysis lies in their varying backgrounds. While a computer science major may have the background in operating systems, etc, to understand analyzing disk organization and other system artifacts through command line tools, an information systems major is much less likely to have that preparation. Commercial forensic tools such as EnCase and Forensic Toolkit (FTK) automate many of the analysis tasks but they are expensive and may be beyond the reach of some programs. Fortunately, Helix includes several Open Source tools which will allow students of varying backgrounds to conduct a basic forensic analysis.

4.1 Autopsy

Autopsy (www.sleuthkit/org/autopsy) is a graphical front-end to The Sleuth Kit written by Brian Carrier and is described in both the Helix manual referenced earlier and in (The Honeynet Project, 2004). Autopsy is launched by clicking on the "Helix" icon and then selecting it from the "Forensics" menu as shown below.

Autopsy is located in the "Forensics" sub-menu off the "Helix" menu. Autopsy runs in a browser and guides the user through creating a new case and adding a disk image to it. Once the image is added, analysis is begun by selecting the volume and clicking "Analyze."

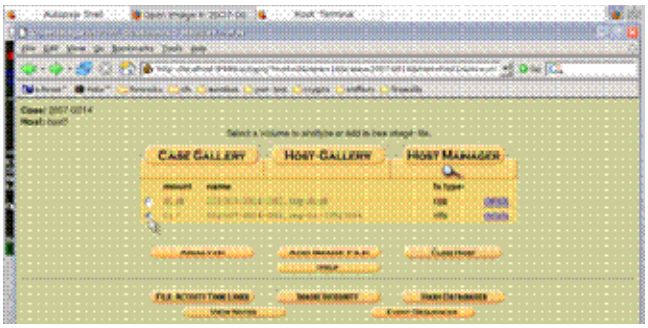

Figure 8 Autopsy Select Volume

This brings up the main analysis page where a file level analysis can be begun by selecting "File Analysis."

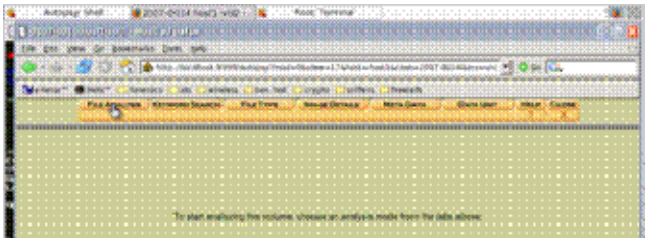

Figure 9 File Analysis

One of the most common forensic tasks involves retrieving deleted files and this process is easily launched by clicking on "All Deleted Files."

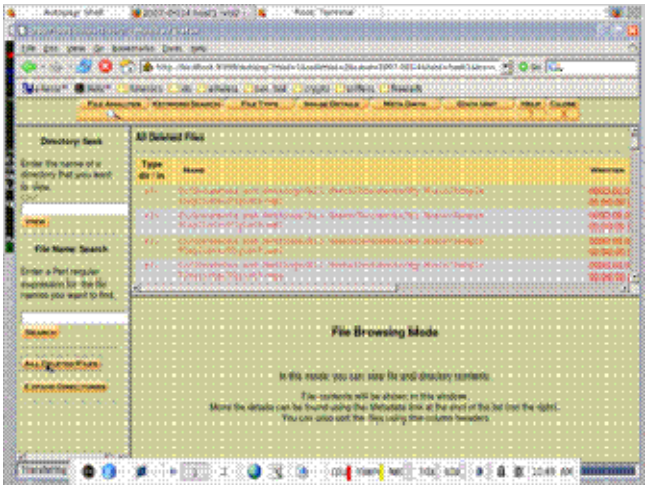

Figure 10 Browsing Deleted Files

Directory and file names function as hot links where clicking on one will navigate to that directory or display the contents of the file. The following example shows the contents of the boot.ini file from the "My Documents" subdirectory for the user "TestUser."

[2] The author is aware that cryptographers will wince at this use of "pre-image" but this usage is common in forensic practice.

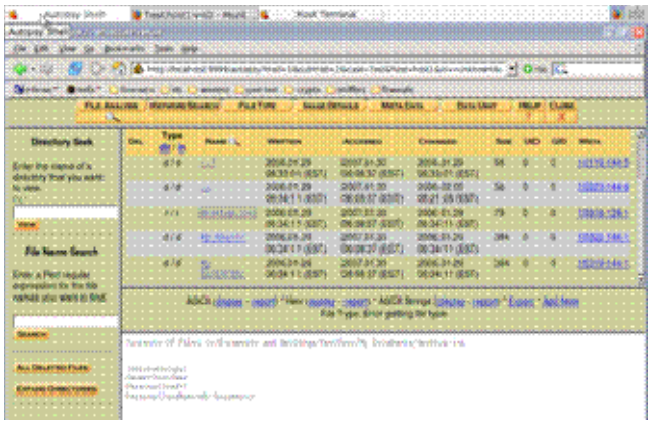

Figure 11 Browsing Files

Besides deleting the file, another common way that users will attempt to hide information is by changing the file extension (e.g., renaming CREDITCARDS.DOC to CREDITCARDS.JPG). Forensic tools attempt to identify the true type of a file by examining its internal structure and headers and this is easily done in Autopsy by clicking on "File Type," selecting options and clicking on "OK." In the example shown below, options are selected to sort the files into categories by type (saving graphics images and thumbnails) and to verify file extensions.

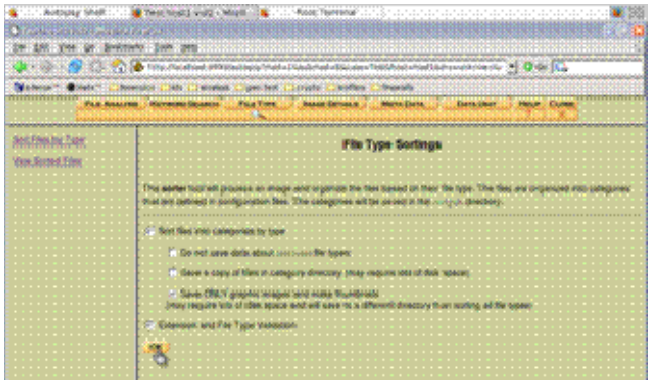

Figure 12 File Type Sorting

The next useful tool in forensic analysis is the ability to do keyword searches and can be conducted in Autopsy by clicking on the "Keyword Search" tab.

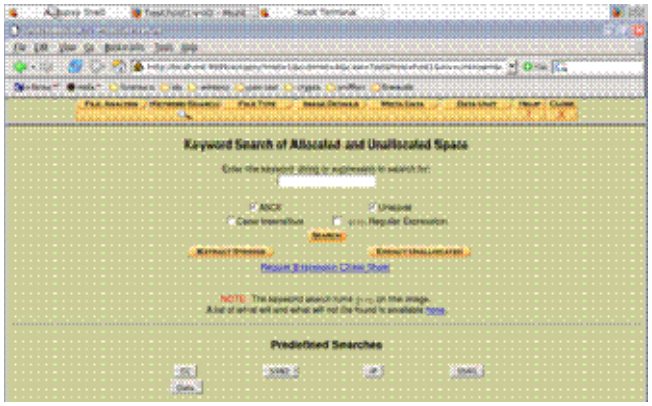

Figure 13 Keyword Searches

Searches for either specific terms or regular expressions is supported and predefined searches for credit card numbers, social security numbers, etc, are available.

4.2 Additional Tools

Helix includes additional tools such as a registry viewer, hex editor, etc, that are useful in forensic analysis. Additional tools such as PASCO2 (http://sourceforge.net/projects/pasco2/) add the capability to parse Internet Explorer history and cache files.

5. Conclusion

Wide the wide availability of Open Source tools such as Helix and Autopsy, the cost of specialized hardware need no longer be a barrier to teaching digital forensics in the information security curriculum.

These tools make it possible for a wide range of students to explore the introductory process and practice of digital forensics by conducting a realistic investigation without requiring tools or equipment beyond that commonly found in the collegiate computer lab.

6. References

AccessData (2006). MD5 Collisions: The Effect on Computer Forensics. Downloaded from http://www.accessdata.com/media/en_US/print/papers/wp.MD5_Collisions.en_us.pdf on 15July 2007.

Brown, C. (2006). Computer Evidence: Collection and Preservation. Hingham: Charles River Media.

Casey, E. (2004). Digital Evidence and Computer Crime (2ed). Boston:Elsevier.

Nemeth, C. (2001). Law & Evidence: A Primer for Criminal Justice, Criminology, Law and Legal Studies. Upper Saddle River: Prentice-Hall.

NIST CFTT (2001). Disk Imaging Tool Specification, version 3.1.6. Downloaded from http://www.cftt.nist.gov/DI-spec-3-1-6.doc on 14July 2007.

Rivest, R. (1992). The MD5 Message-Digest Algorithm, RFC-1321. Downloaded from ftp://ftp.rfc-editor.org/in-notes/pdfrfc/rfc1321.txt.pdf on 15 July2007.

Theage.com.au (2005). NSW Speed Cameras in Doubt. Downloaded from http://www.theage.com.au/articles/2005/08/10/1123353368652.html on 15uly 2007.

The Honeynet Project (2004). Know Your Enemy: Learning about Security Threats. Boston:Addison-Wesley.

Wang, X. & Yu, H. (2005). How to Break MD5 and Other Hash Functions. Eurocrypt 2005.