Laureate Online Education

Computer Forensics

Computer Forensics

Seminar for Week 4: Investigating Windows Systems

In this seminar, we will examine the basic techniques and software tools to investigate computers running Microsoft Windows operating systems.

Overview

The important things that a digital evidence investigator must understand about computers with Windows operating systems include file systems, account management systems, log files, and other advanced techniques such as .NET framework. There are several variants of Windows operating systems, and each of them may store valuable forensic information in different locations. Furthermore, different cases require digital investigators to explore and research different components. In this seminar, we will discuss the important *common* aspects of Windows systems with the expectation that you will know how to extract valuable information from Windows for each different, new case, even if we do not address the exact location for that kind of information.

Digital Evidence Acquisition Techniques for Windows

In the previous seminar, we discussed several approaches to acquire a forensic hard drive. For example, one can use a Linux bootable CD to bit-by-bit copy the target drive (even if the target drive is Windows based) to a third place, or use commercial tools such as EnCase or Forensic Tool Kit (FTK) to do that. Furthermore, EABD describes a method for generating an Evidence Acquisition Boot Disk (EABD) in Windows 95, which could be used to acquire hard drives. These tools will generally boot the computer with the basic components of the operating systems and then copy the data to a third drive. However, the booting process may unexpectedly write to the target drive. For example, the Microsoft Windows NT/2000/XP booting process accesses more than 500 files (Windows 9x accesses more than 400 files). Without sufficient preservation mechanisms, it could be difficult for a digital investigator to explain in court why the system still contains the acceptable evidence after more than 500 files were “changed” during the boot process (before the evidence was acquired). In order to avoid the unexpected writes, one may use the physical drive blocker (PDBLOCK) to disable both the standard Interrupt 13 and the Interrupt 13 Extensions (int13h).

After the target hard drive image is acquired, we need to search/analyze the image for potential evidence. In order for this process to be effective, it is essential to understand the basics of FAT12/FAT16/FAT32/FATNTFS/etc. file systems, which are used by Windows operating systems. See [FAT] for comprehensive discussions on FAT-based file systems.

NTFS Alternate Data Stream, Compressed Files, and EFS

Microsoft introduced the New Technology File System (NTFS) for Windows NT. Since then, NTFS has been used in Windows 2000 and XP. The detailed descriptions of NTFS can be found at (NTFS). A NTFS disk begins with the partition boot sector, which starts at sector 0 and can expand to 16 sectors. After the partition boot sector, the disk contains the master file table (MFT), which usually consumes about 12.5% of the disk when it is created. As data are added, the MFT can expand to take up 50% of the disk. MFT contains information about all files located on the disk, including the system files the operating system uses. In NTFS, all files and folders have file attributes. Individual elements of a file, such as its name, security information, and even the data, are considered file attributes. Each attribute has a unique attribute type. Compared to FAT-based file systems, NTFS contains much less file slack space.

NTFS supports alternate data streams (one of the main reasons for Microsoft to support multiple stream data is for Macintosh file support), which allow data to be appended to existing files without being observed by regular applications. Attackers may use this function to hide valuable information to an existing file without other users observing it. For example, a network intruder may install backdoor (malicious) programs to a victim system as an alternate data stream to a regular application program, but the owner will not notice the existence of such malicious software. The web site (NTFSdangers) contains a detailed discussion on potential dangers of NTFS alternate data streams. When information is added to an existing file as a multiple data stream, the data stream becomes an additional data attribute of a file, and the file can be associated with different applications. For example, a stream hostfile.xxx:Stream.doc can be considered a regular WinWord document and be opened/edited by WinWord or other application software, although you can only see the hostfile.xxx file. Thus, a digital forensic investigator should also check all potential data streams in which suspects may hide valuable information. You may practice this functionality in the group project work. In the following, we use an example to illustrate how this works in Windows 2000/XP:

1. Create a command interface by running cmd (in the Start > Run window).
2. Assume that we do not have the file “file.txt” in the current directory.
3. In the cmd interface, type: “echo AnyStringThatYouWant > file.txt:AnyNameThatYouWant”.
4. Type “dir file.txt” to see that the file.txt has size 0.
5. Type “more file.txt” to examine the file content and you will not see the stream data.
6. In order to examine the stream data, you need to type “more < file.txt: AnyNameThatYouWant”.

A data stream cannot be examined via regular text editors. If we know the stream name, we can use “more” to view the stream content. If we do not know whether there are alternate stream data on a computer, the only way to find out is to examine the MFT of the hard drive. Some tools are available for this purpose. For example, freeware LADS by Frank Heyne can be used to examine all alternate data streams on a hard drive.

NTFS provides compression capabilities to compress individual files, folders, or entire volumes. During an investigation, we normally work with a bitstream image copy of a compressed disk. Thus, we may need to uncompress the data first before further investigation. For example, several forensic tools often need to search the entire image for certain key words. Without uncompressing the compressed volumes/directories/files, the search function cannot work appropriately.

Windows 2000/XP has a built-in encryption mechanism, EFS (encrypted file system), for NTFS that uses a public key cryptography to encrypt files, folders, or disk volumes (partitions). Data encrypted with a public key can only be decrypted with the corresponding private key. When a user implements EFS, a recovery certificate is generated and sent to the local Windows 2000/XP administrator’s account. The recovery key agent implements the recovery certificate, which is in the Windows 2000/XP administrator account. Windows 2000/XP administrators can recover a key through Windows interface or through an MS-DOS command prompt using the commands: cipher, copy, and efsrecvr. When analyzing an acquired drive with EFS enabled, digital investigators can use this commands to recover the user’s private key first for data decryption.

Examining Acquired Evidence

Acquired drive images can be viewed either physically or logically. That is, the raw data on the acquired drive can be viewed/examined physically using a disk editor such as Norton DiskEdit or WinHex, or the file system on the acquired drive can be viewed/examined/analyzed logically using tools such as Norton Commander. As we learned last week, WinHex is a powerful tool with examination and analysis capabilities, such as recovering all slack or unallocated space and comparing files to find any differences. For example, one can use WinHex to compare two seemingly identical Word documents created at different times to locate internal date-time stamps.

Both the logical and physical methods of viewing a disk image have limitations. For example, when searching for a keyword, a physical sector-by-sector search will not find occurrences of the keyword that are broken across non-adjacent sectors. On the other hand, a physical examination gives access to areas of the disk that are not represented by the file system, such as file slack and unallocated space. Integrated tools like EnCase and Forensic Toolkit (FTK) on Windows and the Sleuth Kit on Linux environments combine both of these methods and other features into a single tool, enabling an examiner to view a disk physically and logically at the same time.

In some cases, access to data on an acquired drive is protected with a password. In a simple case, it may be possible to use a hexadecimal editor like WinHex to simply remove the password within a file. There are also many specialized tools that can bypass or recover passwords of various files. For example, the tools at (lostpassword) provides password recovery tools for different files such as Office, Excel, Word, Outlook, and so on. NTI sells a collection of password recovery tools. Russian password crackers, LC4, and others are often useful. When performing a logical reconstruction using a restored clone of a Windows NT/2000/XP system, it may be necessary to bypass the logon password using programs such as ntpasswd.

Data Recovery and File Signatures

There are two main forms of forensic data recovery in FAT file systems: recovering deleted data from unallocated space and recovering data from slack space. Recently deleted files can often be recovered from unallocated space by reconnecting links in the chain. For example, to recover the deleted file named “UoLforensics.doc” on an acquired drive, it is often sufficient to replace the sigma “σ” with an underscore “_” in the directory it resides in. FAT-based file systems use the symbol sigma to indicate that a file is deleted. Note that the recovery process must be performed on a copy of the evidentiary disk because it requires the examiner to alter data on the disk.

Assuming that this file occupies contiguous clusters on the drive, then the above process will normally work. Recovering fragmented files is more challenging since one needs more effort and experience to locate the next link in the data chain. FTK and EnCase have utilities for automatically recovering all deleted directories and files that are contained on a FAT volume.

If you are analyzing the acquired drives on a Linux machine, you can also use tools such as fatback, the Sleuth Kit, and SMART to recover deleted files from FAT-based file systems. Note that the Sleuth Kit enables investigators to examine data at the logical and physical level; it can also be used to recover files from NTFS file systems. The Sleuth Kit can also be used to recover slack space from FAT and NTFS systems using the command “dls –s”.

This undelete software utilizes the logical structure of the file system to identify deleted files and rebuild the deleted file data structure by referencing data blocks that are not allocated to any current file. Another approach to recovering deleted files is to search unallocated space, swap files, and other digital objects for file signatures, headers, and footers. This approach relies on identifying potential deleted files or file fragments by unique signatures of targeting files. This approach is called *file carving*. Most files stored on a computer system consist of a simple structure: the file header, the file body, and the file footer. The file header contains information specific to the particular type of file. For example, in a JPEG file, the file header contains information regarding the picture resolution and a unique string “FF D8 FF E0” in hexadecimal representation (also called *magic number* or *signature*). In the case of Microsoft Word documents created with MS Office, the commonly used signature is “D0 CF 11 E0 A1 B1 1A E1 00 00 00 00”.

The undelete method based on signature analysis approach scans a block of data for the chosen signature. When an instance of the signature is found, one may decide to extract a certain size block of data. The following is a list of file signatures for commonly seen files. The table is compiled from data provided in the page (Garykessler). You can also find file signatures at (file signatures), or (file magic number):

File Type	**Extension**	**Hexadecimal Signature**
Bitmap graphics file	bmp	42 4D 00 00 00 00
Cursor	cur	00 00 02 00 01 00 20 20
Excel 2	xls	09 00 04 00
Excel 3–4	xls	09 00 06 00 00 00 10 00
GIF graphics file	gif	47 49 46 38 37 61
JPEG graphics file	jpg	FF D8 FF E0
MS Office	doc/xls/mdb	D0 CF 11 E0 A1 B1 1A E1 00 00 00 00
Microsoft Word 1	doc	9B A5
Microsoft Word 2	doc	DB A5
Microsoft Word 4–5	doc	31 BE 00 00
Netscape 3 mail	snm	23 20 4E 65 74 73 63 61 70 65
Outlook mail file	pst	21 42 44 4E
Outlook address file	pab	21 42 44 4E
PowerPoint 3.0	ppt	ED DE
Printer spool file	emf	01 00 00 00 58 00 00 00

Signature-based file recovery tools such as WinHex, FTK, EnCase, DataLifter, NTI's Graphics Image File Extractor, and Ontrack's Easy-Recovery Pro can recover many types of files including graphics, word processing, and executable files. Files with user signatures can be recovered using WinHex "file Recovery by Type" feature or EnCase e-scripts available from the EnScript library.

One of the main limitations of these tools is that they generally rely on files' having intact headers. When file headers have been modified, it may be necessary to search for other characteristics of the desired files and piece fragments together manually.

Note that another common use of file signatures in computer forensics is as a means of verifying the accuracy of each file's extension. In this approach, all files are scanned and their file signatures are matched against known signatures for their particular file type. If a file has been labeled as a data file (e.g., with extension txt) but contains an MS Office signature, then this file must have been deliberately disguised and special attention should be paid to it.

Indeed, in earlier versions of Microsoft Office (e.g., Office 97), each Word file contains the MAC address of the computer on which the file was generated (OfficeMAC). For example, if you save a file as a Word 97 document, then you can check that it contains a string like "PID_GUID_{280BAEE*-B239-11D2-A503-0000E861E4BD}". This information will certainly be useful for forensic purposes. However, due to the objection from privacy groups, this information is no longer available in newer versions of Word documents.

Experiments with Digital Forensics Tool Testing Images

In the last seminar, we mentioned that the NIST Computer Forensics Tool Testing (CFTT) Project tries to increase the public's confidence in software and hardware forensic tools by developing extensive and exhaustive tests for digital investigation tools. The goal of CFTT is to validate forensic tools. On the other hand, Digital Forensics Tool Testing Images (Images) is a project that designs small test cases by creating different disk images for testing digital forensic analysis and acquisition tools. Currently available images include Extended Partition Test image, FAT Keyword Search Test image, NTFS Keyword Search Test image, EXT3FS Keyword Search Test image, FAT Daylight Savings Test image, FAT Undelete Test image, NTFS Undelete (and leap year) Test image, JPEG Search Test image, FAT Volume Label Test image, NTFS Autodetect Test image, and Basic Data Carving Test image. These images are also valuable for beginner forensic investigators to experience different forensic tools and to understand the concepts that we have discussed previously.

Log Files

It is essential to attribute suspected computer activities to specific users in a digital investigation. Log files record which account was used to access a system at a given time. If illegal activities are found on a computer, either legitimate users of the computer or intruders with unauthorized access to the computer could be the suspects. Windows NT/2000/XP stores log files in the “%systemroot%\system32\config\” directory (most commonly, “C:\windowst\system32\config\”). Most of this log file information can be viewed from: “Control Panel”→” Administrative Tools” →”Event Viewer”. Then you can see three kinds of log information: Application, Security, and System. The system log files can contain the information about user accounts that were used to commit a crime and can show that a user account might have been stolen. The application logs also contain user activities on the system. Additionally, event logs can be correlated with file system traces to determine what occurred when a given account was logged in. Different log files can also be correlated to reconstruct the activities that have occurred (e.g., Real-time Log File Analysis Using the Simple Event Correlator (Rouillard)).

File System Traces

Each activity on a computer system leaves traces that may be used for forensic purposes. In addition to the potential system activity logs file, the file date-time stamps may be used for forensic analysis. The common date-time stamps on a file or directory include last modified date-time, last accessed date-time, and created date-time. This kind of information is very important for temporal analysis of digital evidences, which we discussed in previous weeks.

Furthermore, Microsoft Office documents contain the following metadata information that is useful for forensic analysis: the location where a file was stored on disk, the original creation date and time, and others. These metadata can be useful for locating file fragments that were generated while documents were being edited. The activity of printing documents also creates useful trails on file systems. Windows 95/98 stores information relating to printed files in C:\Windows\Spool\Printers\; Windows NT/2000/XP stores them in C:\Windows\System32\Spool\Printers\. These files can contain the name (or URL) of the printed file, application used to print, printer name, and file owner. Furthermore, the date-time stamp of these files indicates when it was printed. When printing a document in EMF mode, the associated spool file contains name of temporal files that were created during the process of printing. These temporal files essentially contain a fragment of the printed document.

Registry

Windows systems use the registry to store system configuration and usage details. Registry files in Windows 95/98 include “system.dat” and “user.dat”, which are located in the Windows installation directory. Registry in Windows NT/2000/XP contains several hive files located in “%systemroot%/system32/config/” and a “ntuser.dat” hive file for each user account.

Acquired registry files can be viewed using the Windows regedt32 command with the Load Hive option on the registry menu. You can type regedt32 in the “Open” field of “Run” command. They can also be viewed with commercial forensics tools such as EnCase. The “Save Subtree As” file menu option of regedt32 can convert the hexadecimal format of the values in some registry keys to ASCII format.

The registry tools will display the registry information in two windows. The left side contains keys (folder icons) and the right side contain values of keys. Keys can contain other keys or values. Values can be one of the three types: binary, string, or DWORD (32 bit). The first key we may want to check is HKEY_CURRENT_USER, which further contains the user profile for the user who is currently logged on to the computer. The user profile includes environment variables, personal program groups, desktop settings, network connections, printers, and application preferences. The details of this key are found at (MSFTregistry). To get a feeling for how it could help in investigation, you may check the following key folder: HKEY_CURRENT_USER/Software/Microsoft/Windows/CurrentVersion/Explorer/ComDlg32/OpenSaveMRU/ and the contents of its subfolders. Can you find any familiar information there?

Further information in the registry is the text strings used to support the functionality "AutoComplete". Starting with Internet Explorer 5, the browser provides an option to keep track of information that users have recently typed (e.g., URL addresses and fields in forms, see MSFTAutoComplete). For example, once a user enters a (string, password) pair on a web page with the AutoComplete feature turned on, that string is helpfully suggested to the user if they ever return to the page; the password is automatically filled in when the string is chosen. This information will be useful in several investigations. For example, digital investigators can use this information to recover the user's password for a remote site. Several tools are available for finding this information automatically. For example, Index Reader can recover this information and many other history data of the web browser.

Furthermore, the HKEY_CURRENT_USER/Software/Microsoft/Internet Explorer/TypedURLs/ key contains a list of URLs that the user typed into the URL location field, and the HKEY_CURRENT_USER/Software/Microsoft/Windows/CurrentVersion/Internet Settings/ contains information about the configuration of the Internet browser. For example, you may find the proxies that the suspect used to access Internet.

HKEY_LOCAL_MACHINE key stores information related to both the PC and the network. For example, the Network/Logon key contains the last username used to log onto a network.

Internet Traces

One important task for digital investigators is to investigate cyber-crimes. Surfing the Internet leaves a wide variety of information on a computer, such as web sites visited, contents viewed, and newsgroups visited. In addition, Windows maintains a log of when the wireless Internet was accessed, and some Internet service provider dial-up programs maintain detailed information of dial-up–related activities.

When users surf web pages, the browser generally caches the pages and associated elements such as images on disk. File download information (can be found in registry, external media, etc.) may provide clues to what kind of the information the suspect was interested in. Netscape maintains a database of web sites visited in a file called "netscape.hst" (deleted entries can be recovered using the method we have discussed). Internet Explorer maintains similar information in files called "index.dat" (earlier versions of Internet Explorer may also use files called MM256.DAT and MM2048.DAT). Some open source utilities have been developed to extract information from "index.dat" files and the browser's other auxiliary files (BrowserAuxiliary). In addition, the file "_CACHE_001_" created by Mozilla contains HTTP responses with web server clock-based date-time stamps that may be more accurate than the local system clock.

Most commercial web sites keep track of users' visits and interests by placing information in cookie files. Though the presence of a cookie does not necessarily prove that an individual intentionally accessed a given web site (e.g., the cookies may be placed via a third party advertisement company associated with the site the user visited), it contains lots of information that could be used for forensic purposes. Cookies for Netscape are stored in the cookies.txt file, and cookies for Internet Explorer are stored in the Windows\Cookies directory.

Web browsers with Usenet readers also keep a record of Usenet newsgroups that have been accessed by the user. For example, Netscape stores this information in a file with an ".rc" extension (similar to UNIX news reader tin). Internet Explorer stores information about newsgroup activities in the News directory (C:/Program Files/Internet Mail and News/user/). This information can help investigators to narrow their search of Usenet to a selection of groups.

In addition to web sites and newsgroups, cyber-crime investigation generally needs to examine emails. While Netscape and Eudora store email in plain text files, Outlook, Outlook Express, and AOL use proprietary formats that require special tools to read. FTK and EnCase can interpret a variety of proprietary email formats.

Instant messengers such as Yahoo pager, AOL IM, and Hotmail MSN do not retain archives of messages by default but may be configured to log chat sessions. Peer-to-peer file sharing, anonymous VoIP calls (e.g., SkyPe over the anonymous Internet service (findnot)), IRC, and other online chat systems may retain some logs but only if the user saves them. Therefore, remnants of these more transient Internet activities are more likely to be found in swap space and other areas of the hard disk. Recently, researchers from George Mason University designed specific techniques to trace anonymous and encrypted VoIP calls by inserting inter-packet delay-based watermarks.

An important component of forensic examinations is to identify any remote locations where digital evidence may be found. For example, one may store emails or web pages at Internet service provider sites. These files/emails may be managed via FTP, SSH, or other network application protocols, and traces may be left in the registry. Thus, it is important to search the registry for traces of network file management applications. Shared drives or network file systems (e.g., NFS or AFS) may also be used to store information remotely. Shared network drive information is generally found in registry files.

Windows XP/2000/NT Startup Files

We close this seminar by briefly discussing the startup process during Windows XP/2000/NT boot processes. When Windows XP starts, the NT Loader (NTLDR) reads the boot.ini file, which displays a boot menu. After the mode to boot to is selected, boot.ini runs Ntoskrnl.exe (a Windows XP kernel that is located in the %systemroot%Windows/System32 folder) and reads Bootvid.dll and the startup device drivers. Other core operating system files that Windows XP/2000/NT uses are located in %systemroot%/Windows/System32 or %systemroot%/Winnt/, including the following files: ntoskrl.exe, ntkrnlpa.exe, hal.dll (hardware abstraction layer dynamic link library), win32k.sys, ntdll.dll, kernel32.dll, advapi32.dll, user32.dll, and gdi32.dll.

Useful Links

1. FAT: General Overview of On-Disk Format http://www.microsoft.com/whdc/system/platform/firmware/fatgen.mspx
2. MS-DOS FORMAT Does Not Preserve Clusters Marked Bad http://support.microsoft.com/default.aspx?scid=kb;en-us;103548
3. Description of NTFS Date and Time Stamps for Files and Folders http://support.microsoft.com/default.aspx?scid=kb;en-us;299648
4. Keeping an Eye on Your NTFS Drives http://www.microsoft.com/msj/0999/journal/journal.aspx and http://www.microsoft.com/msj/1099/journal2/journal2.aspx
5. Detailed Explanation of FAT Boot Sector http://support.microsoft.com/kb/q140418
6. Encodings and Code Pages: http://www.microsoft.com/globaldev/getWR/steps/wrg_codepage.mspx
7. Casey, E. Practical Approaches to Recovering Encrypted Digital Evidence. IJDE 2002 1:3
8. Volume Serial Numbers & Format Verification Date/Time http://www.digital-detective.co.uk/documents/Volume%20Serial%20Numbers.pdf
9. Inside encrypting file system http://www.windowsitpro.com/Articles/Print.cfm?ArticleID=5387 and http://www.windowsitpro.com/Articles/Print.cfm?ArticleID=5592
10. Inside Win2K NTFS http://www.windowsitpro.com/Articles/Print.cfm?ArticleID=15719 and http://www.windowsitpro.com/Articles/Print.cfm?ArticleID=15900
11. Linux NTFS Project http://www.linux-ntfs.org/doku.php
12. Analysis of Reported Vulnerability in the Windows 2000 Encrypting File System (EFS) http://www.microsoft.com/technet/archive/security/news/analefs.mspx

13. INFO: Understanding Encrypted Directories http://support.microsoft.com/default.aspx?scid=kb;en-us;248723
14. FILETIME http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/filetime_str.asp
15. Recovering NTFS Boot Sector on NTFS Partitions http://support.microsoft.com/default.aspx?scid=kb;en-us;q153973

Bibliography

Casey, E., (2011) *Digital evidence and computer crime: forensic science, computers and the internet*. 3rd ed. New York: Elsevier Academic Press.

PDBLOCK http://www.digitalintelligence.com/software/disoftware/pdblock/
FAT http://home.no.net/tkos/info/fat.html and http://www.ntfs.com/fat-systems.htm
NTFS http://www.ntfs.com/ and http://www.microsoft.com/
NTFS data streams http://support.microsoft.com/kb/105763
NTFSdangers http://www.diamondcs.com.au/index.php?page=archive&id=ntfs-streams
Frank Heyne http://www.heysoft.de/Frames/f_home_en.htm
lostpassword http://Lostpassword.com
NTI password recovery tools http://forensics-intl.com/breakers.html
Russian password crackers http://www.password-crackers.com/crack.html
ntpasswd http://home.eunet.no/~pnordahl/ntpasswd
fatback http://sourceforge.net/projects/fatback
the Sleuth Kit http://www.sleuthkit.org/
SMART http://www.asrdata.com/
file carving http://linux.sys-con.com/read/117909_2.htm
Garykessler http://www.garykessler.net/library/file_sigs.html
File signatures http://www.wotsit.org/, http://magicdb.org/
File magic numbers http://www.garykessler.net/library/magic.html
DataLifter http://www.datalifter.com/
Ontrack’s Easy-Recovery Pro http://www.ontrack.com/
OfficeMAC http://scissec.scis.ecu.edu.au/publications/forensics04/Al-Zarouni.pdf
Images http://dftt.sourceforge.net/
Rouillard
http://www.usenix.org/publications/library/proceedings/lisa04/tech/full_papers/rouillard/rouillard.pdf
MSFTregistry http://support.microsoft.com/kb/256986
MSFTAutoComplete
http://www.microsoft.com/windows/ie/using/howto/customizing/autocomplete.mspx
Index Reader http://www.wbaudisch.de/IndexReader/index.html
BrowserAuxiliary http://odessa.sourceforge.net and http://www.foundstone.com/
SkyPe http://www.skype.com
findnot http://www.findnot.com

Reading Requirements

Read chapter 17.