

eForensics

Magazine

Computer

VOL.2NO.20

Windows/Mac Forensics Lab

***HOW TO PERFORM A BASIC AND
FAST FORENSIC ANALYSIS***

***ON MACINTOSH OPERATING
SYSTEMS***

***WHAT TO EXPECT WHEN YOU'RE
ENCRYPTING: CRYPTOGRAPHIC
CHOICES FOR MAC AND WINDOWS***

***THE WINDOWS FORENSIC
ENVIRONMENT***

***HOW TO PERFORM FORENSIC
ANALYSIS ON IOS OPERATING AND
FILE SYSTEMS***

Recommended

Automatically Fix Common Windows Problems for Free

Wise PC 1stAid is a trouble-shooting freeware to help fix common Windows problems in an automatic manner. With it, you can say bye to the following & further unlimited problems:

Icon errors, broken links, unable to open regedit/task manager/webpages, slow internet connections, slow startup, slow PC...



WiseCleaner

Wise PC 1stAid

- ✓ Easy to Match Problem
- ✓ Fast, Automatic & Intelligent Fix
- ✓ In-time, Unlimited & Active Enrichment
- ✓ Unlimited Technical Support



Highly Reviewed by
Professionals

Official Website for More Information:
www.wisecleaner.com/wisepc1staid.html



Support system:
Windows XP, Vista, Win7/8
(both 32-bit and 64-bit)

BOOK BY THE 31st DECEMBER 2013 AND RECEIVE UP TO 20% OFF REGISTRATION FEE

Cyber Intelligence Asia 2014

11th - 14th March 2014, Singapore

Esteemed Speaker Line-up:

- **Major General Bunjerd Tientongdee**, Deputy Director of Defense Information and Space Technology Department (DIST), **Ministry of Defence, Thailand**
- **Yurie Ito**, Chair, **Asia-Pacific Computer Emergency Response Team (APCERT)**
- **Phannarith Ou**, Head, **Cambodia Computer Emergency Response Team (CamCERT) Cambodia**
- **Budi Rahardjo**, President, **Indonesia Computer Emergency Response Team (ID-CERT)**, Indonesia
- **Khamla Sounnalat**, Deputy Head, **Lao Computer Emergency Response Team (LaoCERT)**, Lao
- **Philip Victor**, Director, Centre for Policy & International Cooperation, **IMPACT**
- **Inspector Allan Cabanlong**, Chief, Web Services and Cyber Security Division, **Philippine National Police Force**
- **Serupepeli Neiko**, Section Head, Cybercrime Division, **Fiji Police Force**
- **Dr. Mingu Jumaan**, Director, **Sabah State Computer Services Department, Malaysia**
- **Jack YS Lin**, Senior Security Analyst, **Japan Computer Emergency Response Team (JPCERT)**, Japan
- **Dr. Frank Law**, President, **High Technology Crime Investigation Association (HTCIA)**
- **Ammar Jafri**, President, **Pakistan Information Security Association (PISA)**
- **Andrey Komarov**, Chief Technology Officer, CERT-GIB, **Russian Law Enforcement Agency**
- **Senior Representative, Ministry of Internal Affairs, Russia**
- **Senior Representative, Infocomm Development Agency (IDA), Singapore**
- **Kiran Karnad**, Staff Engineer, **MiMOS, Malaysia**

Reasons to attend:

- ✓ Largest international gathering of cyber security experts in ASEAN
- ✓ Opportunity to network with the leading firms who provide defences to cyber attacks
- ✓ Analyse the latest cyber security challenges and issues in the region
- ✓ Discuss international cooperation to combat cyber-crime
- ✓ Network with the leading decision makers in the government's
- ✓ Determine the latest cyber-crimes taking place in ASEAN
- ✓ Gain a mix of policy, strategies and technical expertise in one place

Associated Workshops :

☐ Strategic Co-operation amongst CERT's

Led by: Asia-Pacific Computer Emergency Response Team (APCERT)

☐ OWASP Top 3 - Injection, Session Management and Cross Site Scripting: Hands-on with Kali Linux

Led by: MiMOS Malaysia

For more information visit – www.intelligence-sec.com**Book your place by:****Web: www.intelligence-sec.com | Email: events@intelligence-sec.com | Tel: +44(0)1582 346706**

Editors:

Artur Inderike

artur.inderike@eforensicsmag.com

Betatesters/Proofreaders:

M1ndl3ss, Salvatore Fiorillo, Kishore PV,
Richard C. Leitz Jr., Simohammed Serrhini,
Dr D. B. Karron, Andrew J. Levandoski,
Owain Williams, Alex Rams, Sir James
Fleit, JohanScholtz, Olivier Caleff, Leighton
Johnson, Derek Thomas, Martin Baader,
DANNY LAVARDERA, Jeff Weaver,
Pardhasaradhi C. H., Henrik Becker, Robert
Vanaman, Nicolas Villatte, Luca Losio,
Massa Danilo, Christopher P. Collins,
Sundarapariipurnan Narayanan.

Senior Consultant/Publisher:

Paweł Marciniak

CEO: Ewa Dudzic

ewa.dudzic@software.com.pl

Production Director: Andrzej Kuca

andrzej.kuca@software.com.pl

Marketing Director: Joanna Kretowicz

jaonna.kretowicz@eforensicsmag.com

Art Director: Ireneusz Pogroszewski

ireneusz.pogroszewski@software.com.pl

DTP: Ireneusz Pogroszewski

Publisher: Hakin9 Media Sp. z o.o. SK

02-676 Warszawa, ul. Postępu 17D

Phone: 1 917 338 3631

www.eforensicsmag.com

DISCLAIMER!

*The techniques described in our articles
may only be used in private, local net-
works. The editors hold no responsibility
for misuse of the presented techniques or
consequent data loss.*

Dear eForensics Readers!

Finally we did it!

We've just finished our special edition devoted to Windows & Mac Forensics. The main reason this issue is so important, is that iOS and Mac's are in general becoming more and more popular nowadays. I would even say it is getting kind-off mainstream...

But why do people think that Apple OS is more secure than Windows OS? What the difference between the two operating systems from a forensic investigator's perspective? Let's see!

In this issue we try to pinpoint the differences between Windows and Mac OS's, off course as applied in a forensic science discipline. Also, we would like to destroy the common myth that iOS is the most secured OS in the world. We believe this edition will be interesting for every forensicator. As usual, we collected the most practical articles regarding this subject and we trust you will find them useful.

Thanks again for your support. Don't hesitate to send us your ideas for future special editions.

Peace, love, unity.

Artur Inderike
eForensics Team

WINDOWS REGISTRY FORENSICS 101

by Jason Stradley

This article is meant to serve as a very basic introduction to the Windows Registry and its usefulness as a resource for certain types of forensic investigations. Windows 9x/ME, Windows CE, Windows NT/2000/XP/2003 store configuration data in a data structure called the Registry. The Windows Registry contains lots of information that are of potential evidential value or helpful in aiding forensic examiners on other aspects of forensic analysis. It is a central repository for configuration data that is stored in a hierarchical manner.

HOW TO PERFORM FORENSIC ANALYSIS ON IOS OPERATING AND FILE SYSTEMS

by Deivison Pinheiro Franco and Nágila Magalhães Cardoso

With Apple Operation System (iOS) design and the large amount of storage space available, records of emails, text messages, browsing history, chat, map searching, and more are all being kept. With the amount of information available to forensic analysts on iOS, this article will cover the basics to accurately retrieve evidence from this platform and build forensically analysis when applicable. Once the image logically, via backup or physically has been obtained, files of interest will be highlighted for a forensic examiner to review.

FOUR WINDOWS XP FORENSIC ANALYSIS TIPS & TRICKS

by Davide Barbato

When conducting forensics analysis of a Windows XP system, it must be taken into account some particular behaviors that can lead to misleading conclusions if not properly handled.

WINDOWS MEMORY FORENSICS & MEMORY ACQUISITION

by Dr Craig S. Wright, GSE, GSM, LLM, MStat

This article takes the reader through the process of imaging memory on a live Windows host. This is part one of a six part series and will introduce the reader to the topic before we go into the details of memory forensics. The first step in doing any memory forensics on a Windows host involves acquisition. If we do not have a sample of the memory image from a system we cannot analyze it. This sounds simple, but memory forensics is not like imaging an unmounted hard drive. Memory is powered and dynamic, and changes as we attempt to image it.

HOW TO DETECT A FILE WRITTEN TO AN USB EXTERNAL DEVICE IN WINDOWS FROM THE MRU LISTS

by Carlos Dias da Silva

Today one of the principal company asset is the digital information. The digital information can be used of a lot of methods and also can be copied using different modes. To know and to control what files were sent to out of the company is a problem nowadays and never is a little the investment to guarantee the data secure.

THE WINDOWS FORENSIC ENVIRONMENT

by Brett Shavers

The Windows Forensic Environment, also known as Windows FE or WinFE, is a Windows operating system that can be booted from external media such as a CD, DVD, or USB flash drive. Windows FE is based on Windows PE, which is a minimal Windows operating system with limited services, used to prepare a computer for Windows installation, among other tasks related to Windows. The main, and of course most important, difference between Windows FE and Windows PE, is that Windows FE forensically boots a computer system whereas Windows PE does not.

INTRODUCTION TO WINDOWS FORENSICS USING PARABEN P2 COMMANDER

by Dauda Sule, CISA

Microsoft Windows is the most widely used operating system both for business and personal use. Such popularity has made it one of the most targeted operating systems by malicious attackers. As a result, it is often used as a platform to access personal and work place data, or even to commit policy breaches assisting in the commission of criminal acts. Investigations that are based on electronic evidence stand a very high chance of being carried out on a system with one or the other version of Windows operating system. It is therefore one of the most important operating systems anyone going into the field of cyber forensics will need to know how to investigate.

08

22

34

40

46

50

60

76

HOW TO USE ENCRYPTED ITUNES BACKUPS FOR SMS HISTORY WITHOUT THE DEVICE OR JAILBREAKING

by Gouthum Karadi, CISSP, CEH, MBA

Imagine it is late Friday afternoon at Forensics, Inc. and you get a call from ABC Corp, one of your top clients. It seems that ABC had competitor XYZ cornered and agreeing to submit to a deal before a timely lunch. Yet when talks resumed after the break, XYZ began to negotiate more fiercely. The opponent began to negotiate using not only the exact tactics that ABC prepared for, but even using the exact words in some cases. How could XYZ know what ABC was planning? Someone had to have leaked the internal talking points memorandum the morning of the negotiaton.

92

HOW TO PERFORM A BASIC AND FAST FORENSIC ANALYSIS ON MACINTOSH OPERATING SYSTEMS – A QUICK START GUIDE

by Deivison Pinheiro Franco

Computer Forensics is an area that is very Windows-centric. Many tools pay lip service to Apple's Macintosh (Mac) platform, and others do not even recognize it at all. The few Mac tools available are either expensive or inadequate. Regardless, it is necessary for an investigator to know what to look for and where to look. This article is intended to give investigators a brief outline of what the file system and structure of a Mac looks like and to give a basic criteria on what to look for, as well as some generalized locations for where to look. It is far from a comprehensive forensic manual for Macintosh computers, but it does attempt to give an examiner relatively comfortable with Windows environments a place to start learning about Mac forensics.

100

HOW TO STEAL GMAIL CREDENTIALS USING SE-TOOLKIT – A CASE STUDY IN SOCIAL ENGINEERING

by Kevin M. Moker

Hacking? Why hack when you can trick someone more easily than trying to hack into his or her computer? I am talking about social engineering (SE). SE, in the context of information security, is the ability to manipulate someone to steal certain information. Using SE you can steal credit card numbers, or better yet steal someone's login credentials. With no hacking involved, you will be able to easily reroute payroll funds from an employee's account to another account before they even know the money is gone. However, with the right knowledge, a victim could thwart an adverse attack. Non-technical individuals should learn how to protect themselves when online. Non-techies should understand what SE is and how to protect themselves.

106

WHAT TO EXPECT WHEN YOU'RE ENCRYPTING CRYPTOGRAPHIC CHOICES FOR MAC AND WINDOWS

by Eric Vanderburg

There are a variety of options for encrypting data whether you are a Macintosh or Windows user. Some products work for both platforms but Apple and Microsoft have also developed their own built-in products geared towards protecting your data from unauthorized access. These encryption choices are presented here so that you can protect your data no matter which system you want to use.

112

FORENSIC APPROACH TO ANALYSIS OF FILE TIMESTAMPS IN MICROSOFT WINDOWS OPERATING SYSTEMS AND NTFS FILE SYSTEM

by Matveeva Vesta Sergeevna, Leading specialist in computer forensics, Group-IB company

All existing file browsers display 3 timestamps for every file in NTFS file system. Nowadays there are a lot of utilities that can manipulate temporal attributes to conceal the traces of file using. However, every file in NTFS has 8 timestamps that are stored in file record in MFT and are used in detecting the fact of attributes substitution. The author suggests a method of revealing original timestamps after replacement and automated variant of it in case of a set of files.

120

WINDOWS FORENSICS AND SECURITY

by Adrian Leon Mare

The world we live in today is a technologically advanced world. While on one hand, commercialization of IT (Information technology) revolutionized our modern day lifestyle, it has raised a big question mark about the confidentiality and privacy of the information shared and managed using advanced means of communication.



Looking to Secure a CISOs Attention? CDM Media's CISO Summit has the Key

Great things happen for business when you meet key decision-makers face-to-face, but you don't always get those opportunities during the everyday business routine. What if you could make these high-level business contacts inside of a world-class environment, amongst the security industry's elite, while discussing the most prominent issues facing today's information security professionals? Well, thanks to the CISO Summit, this has never been easier!

CDM Media's CISO Summit provides attendees with a 2-day agenda designed to balance intimate networking, impactful thought leadership and valuable relationship-building, all the while engaging key topics surrounding information security in the US and across the globe. Connect with a prime selection of CISOs, IT security executives and leading edge-solution providers and see how you can advance your own business and create partnerships that will last throughout your career.

Only a few spots remain, so register for the CISO Summit today at www.cisosummit.us. You can also register by contacting us at marketing@cdmmedia.com.



Dec 3 - Dec 4, 2013

**Westin Kierland
Resort & Spa**

Scottsdale, AZ, USA

CISOSUMMIT.US

WINDOWS REGISTRY FORENSICS 101

by Jason Stradley

This article is meant to serve as a very basic introduction to the Windows Registry and its usefulness as a resource for certain types of forensic investigations. Windows 9x/ME, Windows CE, Windows NT/2000/XP/2003 store configuration data in a data structure called the Registry. The Windows Registry contains lots of information that are of potential evidential value or helpful in aiding forensic examiners on other aspects of forensic analysis. It is a central repository for configuration data that is stored in a hierarchical manner.

What you will learn:

- Windows Registry Structure
- Windows Registry Forensic Analysis skills
- Windows Registry Analysis tools

What you should know:

- Basic Windows Operating System Skills
- Basic entry level forensics skills

System, users, applications and hardware in the Windows Operating System make use of the Registry to store their configuration information and it is constantly accessed for reference during operation. The Registry was introduced to replace most text-based configuration files used in Windows 3.x and MS-DOS, such as .ini files, autoexec.bat and config.sys. Due to the vast amount of information stored in the Windows Registry, it can be an excellent source for potential evidential data. For instance, the Windows Registry contains information on user accounts, typed URLs, network shares, and run command history. While the Windows Registry has many similarities across the various versions of Windows operating system, aspects discussed in this article are based solely on Windows 7 Service Pack 1 and tools that work in that environment.

REGISTRY STRUCTURE

The Windows Registry can be seen as one unified file system by invoking the Registry Editor (accomplished by typing regedit in the run window). The left-hand pane, also referred to as the key pane contains an organized listing in a folder-like structure. The five folder-like structures at the top of the hierarchy are called hives and begin with designation "HKEY" (an abbreviation for Handle to a Key).

Although five hives may be seen, only two are actually real, `HKEY_USERS` (HKU) and `HKEY_LOCAL_MACHINE` (HKLM). The other three hives are shortcuts or aliases

to branches within one of the two hives. Each of these five hives is composed of keys, containing values and subkeys. Values designate the names of certain items within a key that identify specific values relating to the operating system, or to applications that are dependent on that value.

Figure 1 shows Windows Registry logical view from Register Editor (Windows default registry editor). Each folder in the left key pane is a registry key. The right panes show the keys value. Subkey is used to show the relationship between a key and the keys nested below it. Branch refers to a key and all its subkeys. Windows uses a symbolic link (i.e. similar to file systems shortcut) to link a key to a different path which allows the same key and its values to appear at two different paths. (Russinovich, 1999).

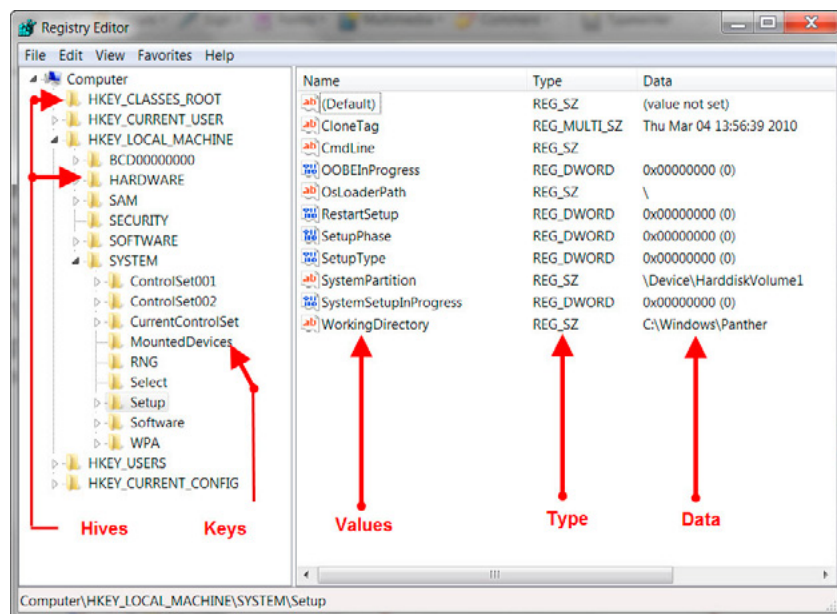


Figure 1. Windows Registry Logical View

A common way of explaining the structure of the Windows Registry is through comparison to the Windows Explorer file system, given their similarity in structure. The structure of the key pane of the Windows Registry is very similar in nature to the left-hand pane in the Windows Explorer file system.

The keys and subkeys located within the five main hives are similar to the folder structure of Windows Explorer with key values being similar to files within a folder. Using this same analogy a value name in the right-hand pane of the Windows Registry is similar to a files name, its type is comparable to a files extension, and its data is akin to the actual contents of a file.

There are 5 root keys (i.e. starting points) in the Windows registry. Table 1 shows the root keys and the abbreviation normally used to represent each of the 5 root keys.

Table 1. Root Keys

Name	Abbreviation
HKEY_CLASSES_ROOT	HKCR
HKEY_CURRENT_USER	HKCU
HKEY_LOCAL_MACHINE	HKLM
HKEY_USERS	HKU
HKEY_CURRENT_CONFIG	HKCC

Below are very basic descriptions for each of the five hierarchical hives listed in Table 1. Beside the root key is their commonly referred to abbreviation in parenthesis, which will frequently be referred to as appropriate throughout the remainder of this article.

HKEY_CLASSES_ROOT (HKCR)

This hive stores information that ensures that the correct program opens when it is executed in Windows Explorer. It also contains additional details on shortcuts, drag-and-drop rules, and information on the user interface. Alias for: `HKLM\Software\Classes`.

HKEY_CURRENT_USER (HKCU)

The HKCU hive contains configuration information for the current user of the system, including user's profile including; folders, screen colors, and Control Panel settings. The alias for a user specific branch in `HKEY_USERS`. Generic information usually applies to all users and is found in `HKU\DEFAULT`.

HKEY_LOCAL_MACHINE (HKLM)

This hive holds machine hardware-specific information that the operating system runs on. Included in that hardware-specific information is a list of mounted drives and generic configurations of installed hardware and applications.

HKEY_USERS (HKU)

THE HKU hive contains information of all user profiles on the systems, including application configurations, and visual settings.

HKEY_CURRENT_CONFIG (HKCC)

This hive stores information about the systems current configuration. Alias for: `HKLM\Config\profile`.

VALUES

Each key has one or more values. There are 3 parts in value, which are Name, Type and Data, as shown in Table 2.

Table 2. *Value Parts*

Value Parts	Description
Name	Every value has a unique name in that particular key.
Type	Value's type determines the type of data value contains. The common value types in registry for instance are: <code>REG_BINARY</code> type contains binary data; <code>REG_DWORD</code> type contains double-word (32-bit) data; <code>REG_SZ</code> type contains fix-length string data.
Data	Value's data contains data which usually relates to the value's type.

ORGANIZATION OF REGISTRY ROOT KEYS

HKLM and HKU are the only root keys that Windows physically stores on files. HKCU is a symbolic link to subkey in HKU. HKCR and HKCC are symbolic links to subkeys in HKLM. (Honeycutt, 2003).

REGISTRY HIVES

From a forensic analysis perspective, an analyst does not generally interact with the Registry through the Registry Editor. An analyst will most likely interact with Registry hive files directly, through some type of forensic analysis application, or as a result of extracting them from a file system or from an acquired image. There are a number of such tools available, several of which will be discussed later in this article. However, it is important for the analyst to know where these files exist on disk so that they can be retrieved and analyzed. The main, core system Registry hive files (specifically, SAM, Security, Software, Default & System) can be found in the `Windows\system32\config` directory, as illustrated in Table 3 below.

Table 3. *Registry Hive Files in Windows-System32-Config*

Hive	File Location
<code>HKEY_LOCAL_MACHINE \SYSTEM</code>	<code>\system32\config\system</code>
<code>HKEY_LOCAL_MACHINE \SAM</code>	<code>\system32\config\sam</code>
<code>HKEY_LOCAL_MACHINE \SECURITY</code>	<code>\system32\config\security</code>
<code>HKEY_LOCAL_MACHINE \SOFTWARE</code>	<code>\system32\config\software</code>
<code>HKEY_USERS.DEFAULT</code>	<code>\system32\config\default</code>

In addition there are some hives that don't have associated files due to their volatility. The system creates and manages these hives entirely in memory. These hives are consequently temporary in nature and are created at every system boot. Some examples of volatile hive are:

```
HKEY_LOCAL_MACHINE \HARDWARE
HKEY_LOCAL_MACHINE \SYSTEM \Clone
```

WINDOWS REGISTRY SLACK SPACE

Slack space is the colloquial term used to describe remnants of user activity, installed applications, etc. that have been deleted and left behind in Registry hive files that are not part of the active hive file itself. The ability to identify these remnants and analyze them is of great forensic significance to an analyst attempting to piece together all information relevant to an incident. Given that the methods and techniques to accomplish this are somewhat advanced, a more detailed examination of this subject is better left to another discussion.

WINDOWS REGISTRY FORENSIC ANALYSIS

Now that we have a basic understanding of the structure and layout of the Windows Registry it is time to start to discuss the meaning of all of this from the perspective of the incident responder and forensic analyst? What it means is that there is a significant amount of information in the Windows Registry that tells the operating system and applications what to do, where to put things, and how to react to certain stimulus.

While the Windows Registry is forensically significant, often it is not captured during the triage of a live system. Similarly, it is often overlooked during post-mortem examinations. On a regular basis, examiners are faced with many challenges: a lack of training to perform triage on a live system; examining multiple hard drives containing terabytes of data; dealing with pressures from management to complete an arbitrary, often unrealistic, quota of examinations per month; constantly juggling and prioritizing overwhelming caseloads; shortages of personnel; and until recently, limited tools for examining Windows Registry files. When faced with these challenges, it is easy to understand why the Windows Registry is not forensically examined to its fullest a great deal more often.

The Registry contains information that Windows constantly references such as the user profiles, the applications installed on the computer, hardware on or attached to the system, application icons, property sheet folder settings, the ports being used, and so on. From a forensic perspective, the Windows Registry is a veritable goldmine that can often provide probative information to an analyst. For instance, some of the information that can be seen in the Windows Registry includes:

- Auto run locations that list applications to automatically run when the computer is booted
- Lists of the most recently used files or applications
- URL's accessed from a system
- All USB storage devices that have been attached to the computer
- Internet Search Assistant
- Printers, Computers and People
- Remote Desktop – Connections
- MSPaint – Recent Files
- Mapped Network Drives -
- Windows Explorer searches
- WordPad – Recent Files
- Excel – Recent Files

ACCESSED URL'S

An example of the type of forensic data that can be extracted from the Windows operating system is provided through the use of Internet Explorer. Internet Explorer is the default web browser in Windows operating systems. It makes extensive use of the Registry widely in the storage of data, like many applications. Internet Explorer stores its data in the `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer` key. There are three subkeys within the Internet Explorer key that are of the paramount significance to the forensic analyst. The first is `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main`. User's settings for Internet Explorer are stored in this key, and contains information such as search bars, form settings, start pages, etc.

The second and perhaps one of the most informative important subkeys to a forensic analyst is `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs`. Figure 2 demonstrates the content of what the TypedURLs key shows.

From this data a forensic analyst could the following: (1) That someone may have deleted some entries from this subkey given that the entry numbers skip over some values (url9 – url14); (2) that they like to go to the playboy website, which may or may not be a violation of a given organization's corporate policy; and (3) the user has visited the pastebin site, which houses a lot of questionable things and is typically not allowed by many corporate browsing policies. Other things of interest include the fact that the user may have an eBay account, has accessed several financial institutions from the work system and has a LinkedIn account (Figure 2).

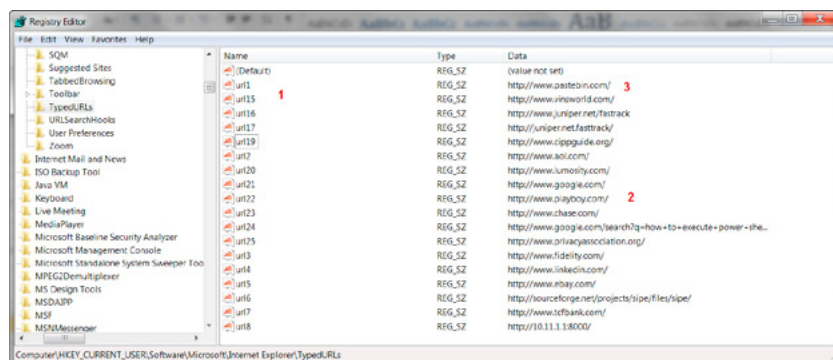


Figure 2. Example Typed URL's

The third subkey that may interest a forensic analyst is `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Download`. This key shows the last directory used to store a downloaded file from Internet Explorer, giving the analyst a clue as to the location of where the user stores their files.

COMMON TOOLS & TECHNIQUES FOR WINDOWS REGISTRY ANALYSIS

There are many automated tools both commercial and open source that have the capability to automate the retrieval of registry contents. In addition there are two basic techniques for performing analysis activities. The first is a live analysis performed on systems while they are powered on and potentially still connected to a network. The second is the classic off-line analysis where most commonly in response to some type of incident a first responder has taken the system off the network and taken disk images on which the analysis will be performed.

For the purposes of this article three open source or freeware tool sets will be examined for the ability to automate the collection of Windows Registry information in a live analysis scenario. After the live analysis scenarios have been reviewed another open source tool will show one method for analyzing Windows Registry files that have been retrieved from a system offline on another system.

The tools sets that will be reviewed for the scenarios described include:

- Microsoft PowerShell Scripting – Live Analysis (Free / Open Source)
- Autoruns – Live Analysis (Free / Open Source)
- SysInternals – Live Analysis (Free / Open Source)
- Registry Decoder – Offline Analysis (Free / Open Source)

MICROSOFT POWERSHELL SCRIPTING

Windows PowerShell is Microsoft's task automation framework, consisting of a command-line shell and associated scripting language built on top of .NET Framework. PowerShell provides full access to COM and WMI, enabling administrators to perform administrative tasks on both local and remote Windows systems. Using the previous example of looking for the TypedURLs a PowerShell command like that shown below will provide a listing of the TypedURLs. After starting PowerShell enter the command string shown below:

```
Get-ItemProperty "HKCU:\SOFTWARE\MICROSOFT\INTERNET EXPLORER\TYPEDURLS"
```


The output of that command yields a listing of the TypedURLs from that registry key as shown in Figure 3.

```

Administrator: Command Prompt - powershell
PS C:\Windows\system32> POWERSHELL
Windows PowerShell
Copyright (C) 2009 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> GET-ITEMPROPERTY "HKCU\SOFTWARE\MICROSOFT\INTERNET EXPLORER\TYPEDURLS"

PSPATH      : Microsoft.PowerShell.Core\Registry::HKEY_CURRENT_USER\SOFTWARE\MICROSOFT\INTERNET EXPLORER\TYPEDURLS
PSPARENTPATH: Microsoft.PowerShell.Core\Registry::HKEY_CURRENT_USER\SOFTWARE\MICROSOFT\INTERNET EXPLORER
PSCHILDNAME : TYPEDURLS
PSDRIVE     : HKCU
PSPROVIDER  : Microsoft.PowerShell.Core\Registry
ur11       : http://www.pastebin.com/
ur12       : http://www.aol.com/
ur13       : http://www.fidelity.com/
ur14       : http://www.linkedin.com/
ur15       : http://www.ebay.com/
ur16       : http://sourceforge.net/projects/sipe/files/sipe/
ur17       : http://www.tcfbank.com/
ur18       : http://10.11.1.1:8000/
ur115      : http://www.vinsworld.com/
ur116      : http://www.juniper.net/faqtrack
ur117      : http://juniper.net/faqtrack/
ur119      : http://www.cipguide.org/
ur120      : http://www.lumosity.com/
ur121      : http://www.google.com/
ur122      : http://www.playboy.com/
ur123      : http://www.chase.com/
ur124      : http://www.google.com/search?q=how+to+execute+power+shell+script+on+windows+7&sourceid=ie7&rls=com.microsoft:en-us:IE-Address&oe=8oe&rlz=117AURU_enUS504
ur125      : http://www.privacyassociation.org/

PS C:\Windows\system32>

```

Figure 3. TypedURLs PowerShell Output

The ability to automate the collection of registry data from multiple registry locations using PowerShell scripting is a valuable addition to any incident responder's or analyst's tool kit.

AUTORUNS

Autoruns.exe is an excellent tool written by Mark Russinovich of Microsoft, part of the SysInternals tool set. Autoruns is a great GUI tool that allows you to see a lot of the various locations on a system, where various programs can be run automatically, with little to no user interaction.

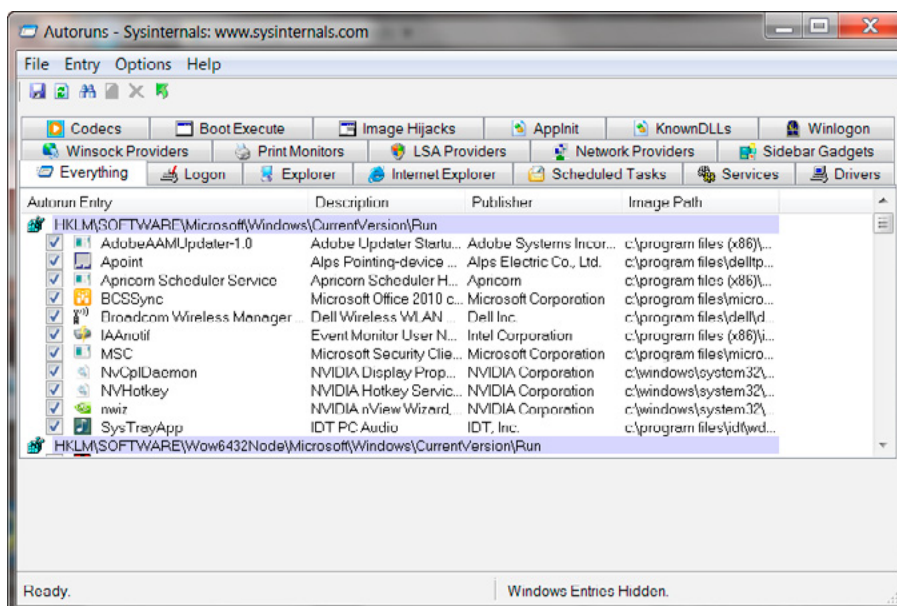


Figure 4. Autoruns, Version 11.42, GUI on Windows 7

Figure 4 shows the Autoruns GUI when the tool is run on Windows 7. The most notable addition to GUI is the available tab named SideBar Gadgets. Figure 4 also shows that there are a number of locations, many of which (albeit not all) are found in the Registry, that allow programs to start automatically, often with no more interaction from the user than booting the system or logging into the system. Autoruns is a very useful tool for troubleshooting systems, as well as for locating malware and suspicious applications, during incident response.

Autoruns comes with a command line companion tool called autorunsc.exe (note the addition of the “c” in the filename), both of which are intended to be run on live systems. Incident responders can include this tool in batch files used for collecting information from systems and gain a considerable amount of insight into what may be happening on the system.

ACCESSING THE WINDOWS REGISTRY REMOTELY

The Autoruns tool can also be deployed remotely by responders using the Psexec.exe (remote command execution tool) also available from Microsoft. As of version 10, Autoruns includes the capability to analyze off-line Registry files; the administrator simply selects the appropriate locations via the “Offline System” dialog box illustrated in Figure 5.

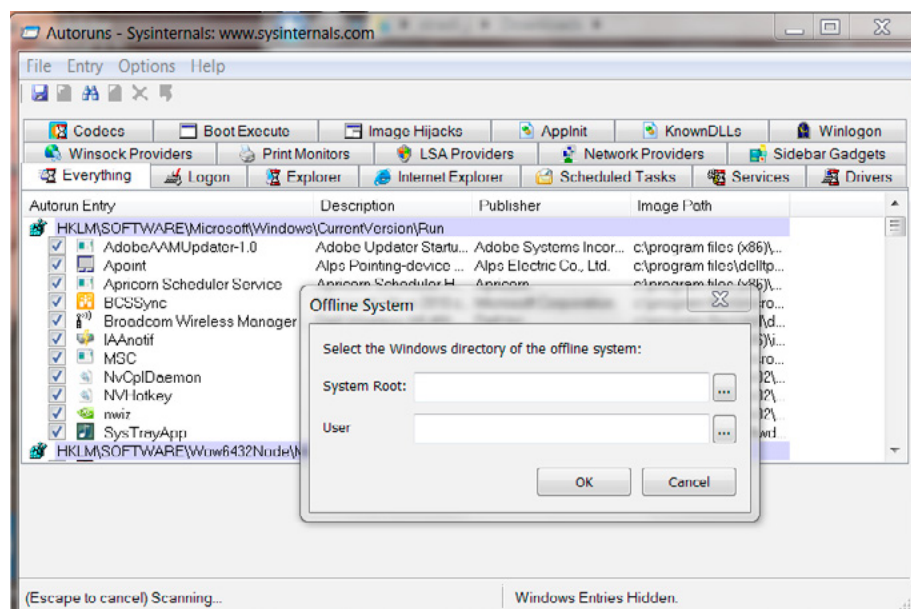


Figure 5. *Autoruns Offline System Selection*

Another method of accessing the Windows Registry of a remote system is a capability that might be very useful for an incident responder or investigator to have. It is possible to access the Windows Registry of a remote system using regedit.exe or reg.exe. After starting the regedit program select the “Connect Network Registry” as shown in Figure 6.

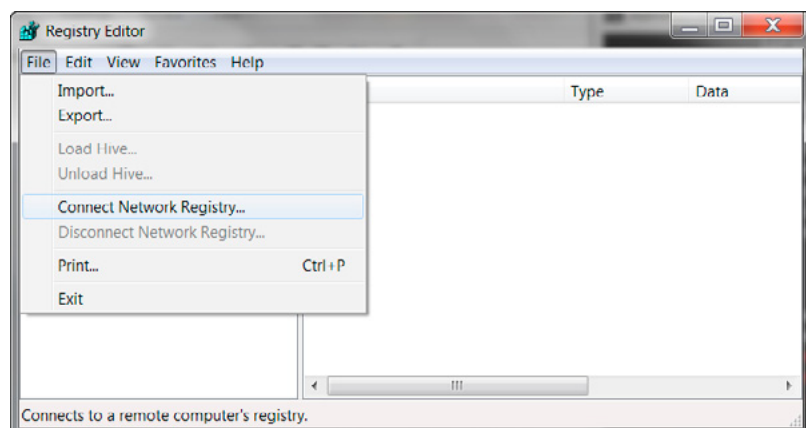


Figure 6. *Connecting to a Remote Registry with Regedit*

Regedit will prompt the user to enter the name of the remote system using host name or IP Address as shown in Figure 7.

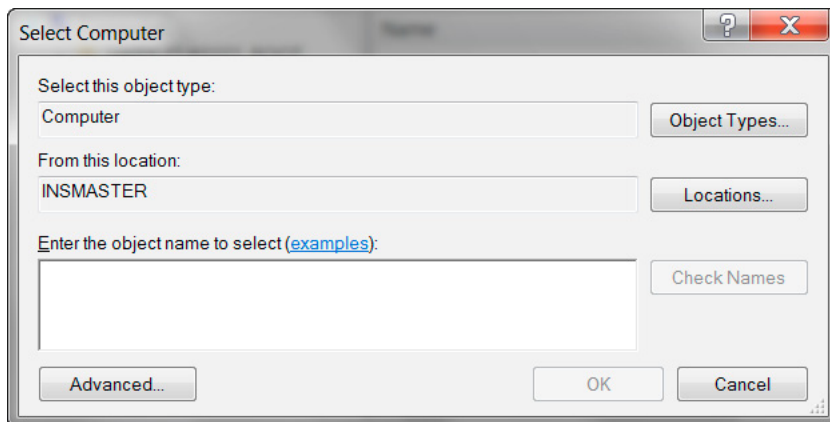


Figure 7. *Regedit Remote Login Prompt*

The user will be prompted for credentials to access the remote systems. Once those credentials are entered the Registry for that remote system will appear below the registry of the local system as shown in Figure 8.

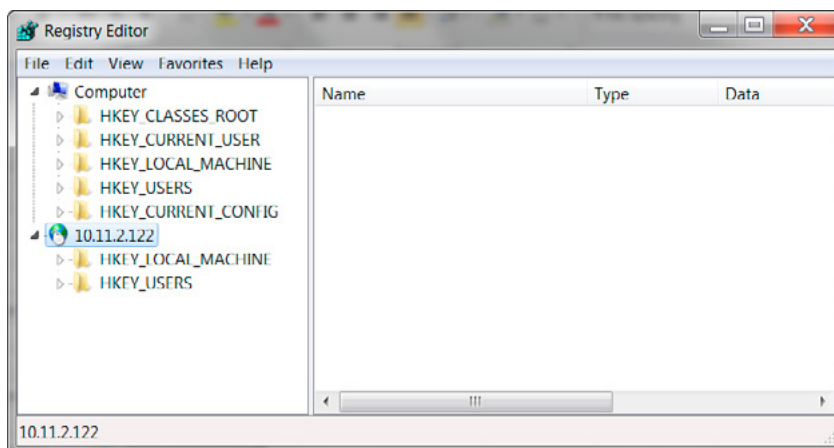


Figure 8. *Regedit Remote System Registry*

The advantage of this method is that it is included in the operating system and is fairly easy to use. The disadvantage is that not all registry information is available. As shown in Figure 8 only the `HKEY_LOCAL_MACHINE` and `HKEY_LOCAL_USERS` keys are directly accessible. The aliases or shortcuts to branches within one of the two hives are not available in the same way as when connecting to the Windows Registry locally.

Other methods to access the Windows Registry of a remote system that permit access to the entire registry include utilizing remote desktop connectivity to run, the Microsoft remote tools framework, PowerShell remote and of course WMI scripting.

REGISTRY DECODER FOR FORENSIC ANALYSIS OF THE WINDOWS REGISTRY

Registry Decoder is an open source project with funding from the *National Institute of Justice* (NIJ) and the *National Institute of Standards & Technology* (NIST). Its purpose is to help automate the acquisition, analysis and reporting of the contents of the Windows Registry. Registry Decoder consists of two components: A live data acquisition tool (Registry Decoder Live); and an offline analysis tool (Registry Decoder).

Obtaining the Windows Registry information using the *Registry Decoder Live* (RDL) tool is a relatively straightforward process. Registry Decoder Live has the ability to obtain Windows Registry data from either the current Windows Registry files on a running system or the backup Windows Registry files.

The acquisition component using the RDP tool is very simple and contains only a single form as shown in Figure 9. Analysts simply need to input a description of the case, select an empty directory in which to copy acquired files, and indicate which registry files should be acquired – the current files, backup files, or both.

The screenshot shows a window titled "Registry Decoder Live - Digital Forensics Solutions". The form contains the following fields and controls:

- Computer Description:** A text input field containing "Dell6300".
- Output Directory:** A text input field containing "C:\tmp\Test001" with a "Browse" button to its right.
- Acquire:** Two radio buttons: "Current Files" (selected) and "Backup Files".
- Acquire Files:** A button at the bottom of the form.

Figure 9. Digital Recorder Offline Case Initiation

Once the acquisition options are chosen, it is just a matter of clicking the “Acquire Files” button and waiting for acquisition to finish. The acquired registry files will be written to the chosen output directory, along with a log file that lists the selected options and acquired files, as well as an SQLite database with information on each file obtained. This directory can then be imported into the offline analysis tool.

Once that data is imported, Registry Decoder can perform an offline analysis of Windows Registry. To begin the offline analysis a new case will need to be initiated using the offline tool. To initiate a new case, just run Registry Decoder and click “Next” on the first form. This will then bring you to the case information form as shown in Figure 10.

The screenshot shows a window titled "Registry Decoder - Digital Forensics Solutions" with a "File" menu and a "Reporting" tab. The form contains the following fields and controls:

- Case Name:** A text input field containing "Test".
- Case Number:** A text input field containing "Test001".
- Investigator Name:** A text input field containing "Stradley".
- Comments:** A text input field containing "Test".
- Case Directory:** A text input field containing "C:\tmp\Test001" with a "Browse" button to its right.
- Create Case:** A button at the bottom of the form.
- Cancel:** A button at the bottom of the form.

Figure 10. Registry Decoder Case Initiation

The case information form is simple and the only field needed is the directory to which case data will be saved. A case's files include a copy of the registry files analyzed and SQLite databases that store needed information. To proceed, just fill out the form and then click "Create Case".

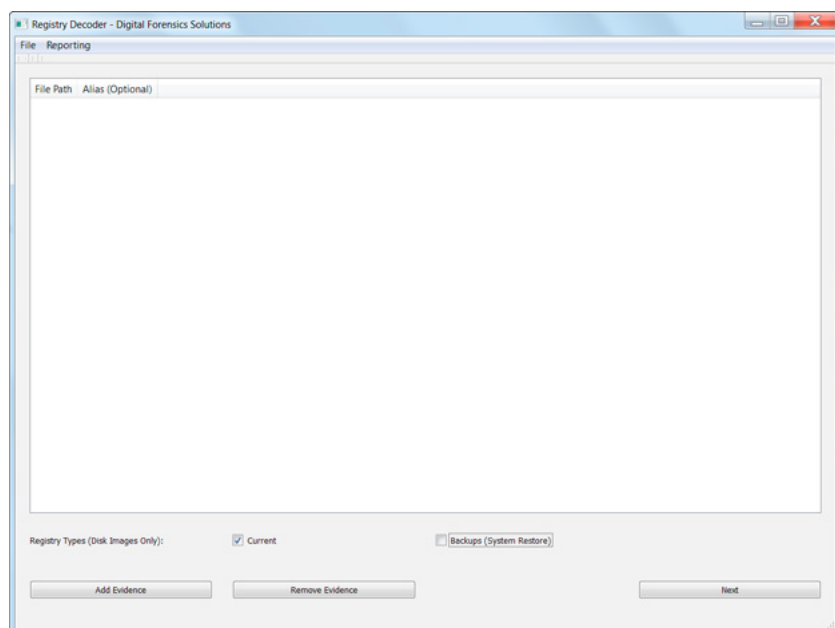


Figure 11. Add Evidence Form

The next form allows for adding of evidence to the case as shown in Figure 11 on the next page. To add evidence click the "Add Evidence" button and then choose the evidence type that is to be used for the offline analysis. Registry Decoder supports the following formats for this function:

- Databases from the live tool
- Individual (or groups of) registry files
- Raw dd disk images
- Split dd images
- Encase (E01) disk images (not the newest version)
- Encase split images

To give a certain piece of evidence an alias, such as the name of the machine from which a registry file came from or the name of the person to whom the disk image belongs, place it in the Alias field.

The "Add Evidence" has two options for disk images, the two checkboxes (current and backup) determine which files will be acquired. The Current files option includes those that would be active on the running machine:

- Everything under c:\windows\system32\config
- All ntuser.dat files

The Backups option will attempt to gather files from the Reg-Back folder of Windows 7.

When all of the evidence has been added to the case, click on the "Next" button to advance to the processing form. To start processing the evidence files click the "Starting Processing" button as shown in Figure 12 on the next page. The evidence will then process, and when completed, will open the investigation tabs. Successful processing of the imported Windows Registry data results in a view of the Analysis Tab as shown in Figure 13.

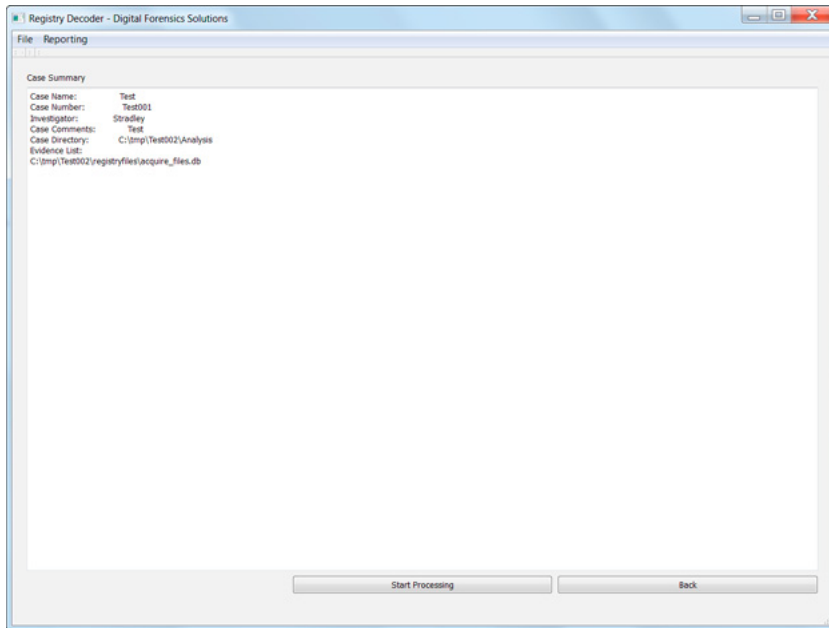


Figure 12. *Processing Form*

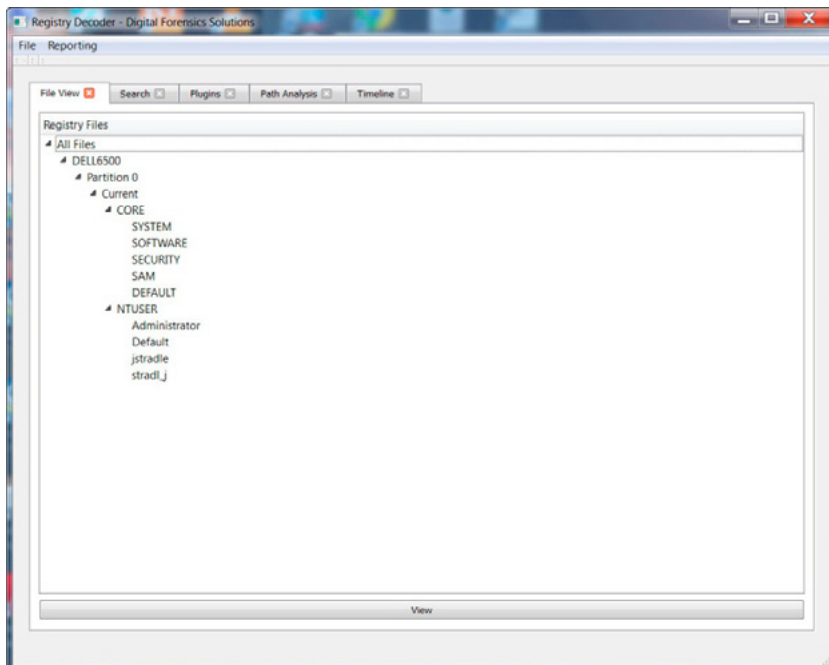


Figure 13. *Analysis Tab*

There are four general characteristics of the Analysis Tab with regard to how data is presented and its overall navigation.

- Each analysis tab shows a tree view of the loaded evidence in a given case file. Any files chosen for any type of analysis available in Registry Decoder will be selected from this tree structure. This tree structure supports a range of selection options including:
 - Selection of a single file.
 - Selection of a group of files, causing the analysis to execute on each file in the group and can include an entire disk image or groups within a disk image.
 - Selection of multiple files or groups throughout the tree structure.
 - Selection of all files will execute the analysis against each file in the case, which may produce a very large amount of data based on the number of evidence files added to the case.
- All tabs generated during analysis can be safely closed, but the initial set of tabs, as shown in Figure 10, may not be closed. Tabs can also be closed automatically using the CTRL+w shortcut.

- Backups of cases can be made through the “File” menu once a case has been loaded. The backup process will create a ZIP file with the chosen name of all files in a case directory. This directory can be later decompressed and opened on any machine running Registry Decoder.
- A case can be closed at any point in time by choosing “Close Case” from the File menu. Any case can be reopened from the initial form where you have the option to open a new case or create a new one.

Now that you have the evidence loaded into Registry Decoder a number of analyses and reporting tasks may be performed including:

- Hive Browsing
 - Similar to regedit and AccessData’s Registry Viewer®.
- Hive Searching
 - Performs full text searching across keys, values, and names
 - Creates tables of results
 - Provides automated reporting of the search term and matches
- Plugins
 - Similar to RegRipper
 - Registry Decoder currently has 30 plugins
 - Provides automated reporting of plugin results
- Hive Differencing
 - Can show the variances between two registry hives using either search or plugin results as the data source
- Timelining
 - Similar to regtime.pl from Harlan Carvey
- Path-Based Analysis
 - Allows exporting and viewing of paths and their key value pairs. Useful to identify if malware or other specific software pieces or events occurred on a computer
- Reporting
 - Searches and plugins can be individually exported to HTML, PDF, or XLS
 - “Bulk” Exports can be performed for all active analysis results tabs

CONCLUSIONS

Given the huge market share that the Windows operating system enjoys for personal and corporate use now and in the foreseeable future it is important for computer forensic analysts to understand the intricacy of the Windows Registry. The data and potential evidence that exists in the Windows Registry make it an essential forensic resource. The ability to consistently retrieve and analyze this data is crucial to any digital investigation. By understanding the fundamentals of the Windows Registry from a forensics standpoint, an analyst can develop greater precision in the accounting of what actions transpired on the given system.

This report should in no way be considered a complete guide to all of the steps and methods required to perform a complete Registry Examination. Such detail will vary based on the type of incident that initiated the examination. It does presents some explanations and examples of what types of data can be found, how it can be found, and why it may be pertinent to an examination. As long as operating systems continue to use the Registry as a configuration database, and applications continue to use that database for storage, there will continually be new locations to discover that provide forensic support in digital investigations.

BIBLIOGRAPHY

- Registry Decoder – Instructions for Offline Analysis Component. (2011, 11 27). Retrieved April 15, 2013, from Registry Decoder: <https://code.google.com/p/registrydecoder/downloads/detail?name=RegistryDecoder-Offline-Analysis-Instructions-v1.1.pdf&can=2&q=>
- Registry Decoder Live – Instructions for Online Acquisition Component. (2012, March 21). Retrieved April 18, 2013, from http://code.google.com/p/regdecoderlive/downloads/detail?name=RegistryDecoder-Online-Acquisition-Instructions-v1.1_ljp.pdf&can=2&q=
- Farmer, D. J. (n.d.). A Forensic Analysis Of The Windows Registry. Retrieved April 18, 2013, from Forensic Focus: <http://www.forensicfocus.com/a-forensic-analysis-of-the-windows-registry>
- Honeycutt, J. (. (2003). Microsoft Windows XP Registry Guide. In J. Honeycutt, Microsoft Windows XP Registry Guide. Microsoft Press.
- Russinovich, M. (1999, May). Inside the Registry. Retrieved March 18, 2012, from www.windowsitpro.com: http://www.windowsitpro.com/Articles/Print.cfm?ArticleID=5195
- Thomassen, J. (2008, April 11). FORENSIC ANALYSIS OF UNALLOCATED SPACE IN WINDOWS REGISTRY HIVE FILES. Retrieved April 16, 2013, from <http://www.google.com/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=2&cad=rja&ved=0CDsQFjAB&url=http%3A%2F%2Fwww.sentinelchicken.com%2Fdata%2FJolantaThomassenDISSERTATION.pdf&ei=TBf4UbcuiP-tAeHZglAI&usq=AFQjCNGGzG0hqgnn74NrsOLwBv0yZ4wNLA&sig2=3soTEuMU6TAXB>
- Wong, L. W. (2007, February 1). Forensic Analysis of the Windows Registry. Retrieved April 17, 2013, from Forensic Focus: <http://www.forensicfocus.com/index.php?name=Content&pid=73&page=1>

ABOUT THE AUTHOR

Jason Stradley is a recognized thought leader in the area of information security. Acknowledged as a visionary security executive he combines an entrepreneurial spirit with the ability to execute against his vision. He has worked with many organizations to develop information security solutions to solve business issues in large complex environments. His organizational expertise and thought leadership combined with his strong communication skills allow him to communicate his vision to all levels in an organization, motivating others to succeed. Mr. Stradley is a published author and frequent speaker. Some of his works have been published in CSO Magazine and the Cutter IT Journal. He has presented at venues including EC Council, SANS, MISTI, Gartner, DRJ and others. Jason currently holds the CISSP, CGEIT, CBCP, CISM, SANS GSLC, CBCP, CRISC, CCSK and C|CISO certifications as well as several solution specific certifications. Jason may be contacted at jstrad@aol.com.



**pending final confirmation*

Confirmed Speakers:

Mr. Noboru Nakatani, Executive Director, INTERPOL Global Complex for Innovation
 Mr. Anwer Yussoff, Head of Innovation and Commercialisation, CyberSecurity Malaysia
 Mr. Mohd Zabri Adil Bin Talib, Head of Digital Forensics, CyberSecurity Malaysia
 Dr. Mingu Jumaan, Director, Sabah State Computer Services Department, Malaysia
 Mr. Lauri Korts-Pärn, CTO, Cyber Defense Institute, Japan
 Mr. Jack YS Lin, Information Security Analyst, JPCERT, Japan
 Mr. Roberto Panganiban, System Administrator, Philippines News Agency
 Mr. Budi Rahardjo, Chairman, ID-CERT, Indonesia *
 Mr. Matthew Gartenberg, Chief Legal Officer, Centre for Strategic Cyberspace + Security Science *
 Mr. Adli Wahid, Manager, Cyber Security / MUFG-CERT, Bank of Tokyo
 Mr. Kislay Chaudhary, Director and Senior Information Security Analyst, Indian Cyber Army
 Mr. Leo Dofiles, Computer Crime Investigator/Computer & Cellphone Forensics Planner, National Police, Philippine
 Mr. Jairam Ramesh, IT Infrastructure, International Multilateral Partnership Against Cyber Threats (IMPACT), Malaysia *
 Mr. Ng Kang Siong, Principle Researcher, MIMOS Berhad, Malaysia

Organised by:



Sponsored by:



Supported by:



Media Partner:



Australia's Security Portal
MySecurity
 .com.au



HOW TO PERFORM FORENSIC ANALYSIS

ON IOS OPERATING AND FILE SYSTEMS

by **Deivison Pinheiro Franco** and **Nágila Magalhães Cardoso**

With Apple Operation System (iOS) design and the large amount of storage space available, records of emails, text messages, browsing history, chat, map searching, and more are all being kept. With the amount of information available to forensic analysts on iOS, this article will cover the basics to accurately retrieve evidence from this platform and build forensically analysis when applicable. Once the image logically, via backup or physically has been obtained, files of interest will be highlighted for a forensic examiner to review.

What you will learn:

- The changes in the Apple Operating System (iOS) and the addition of the App Store to the iOS environment;
- Features that iOS offers and its limitations;
- The iOS Operating and File Systems evolution;
- What iOS Operating and File Systems are and how it can have evidences for forensic analysis;
- Delve into the details of the iDevice file system in order to provide context for investigations.

What you should know:

- A basic understanding of Apple Operating System (iOS);
- A basic understanding of Apple File Systems (HFS, HFS+ and HFSX);
- A basic understanding of mobile forensics analysis.

In this article, we'll look at changes in the operating system (OS) and the addition of the App Store to the iOS environment, and then we'll delve into the details of the iDevice file system in order to provide context for investigations.

iOS, the operating system for the iPhone, iPod, and iPad, was first released with the first-generation iPhone in June 2008. This revolutionized the way cell phones would be created in the future. HTC, Motorola, and Google have since jumped into the smartphone market with their Android phones, as has Research in Motion with its Blackberry phones.

Beginning with iOS 2, Apple allowed the development of application for its App Store. The iPhone SDK gave application developers the

access they needed to write applications for all devices. For a developer to release software to the App Store, the developer had to enroll into the iPhone Developer Program, the initial interface of which is shown in Figure 1. A standard program had a cost of \$99 and an enterprise program had a cost of \$299. The developer also had to sign an extensive agreement with Apple in order to develop and add applications to the App Store. Apple also had a strict and sometimes time-consuming approval process. Over time, Apple has loosened some of its rules, and has even accommodated apps such as Google Voice and applications developed with Adobe Flash.

One of the biggest challenges that Apple has faced is the army of hackers that descended onto the iPhone.

The original hackers of the iPhone justified their actions by virtue of the fact that the iPhone and iOS didn't allow certain functions (e.g., MMS, tethering, customization) or third-party applications other than those available from the App Store. Some hackers also took the stance that the iPhone was insecure, and they wanted to show Apple the flaws that it had. Some of the more notorious groups were the iPhone Dev Team and the Chronic Dev Team. Some of their more maverick members have splintered to develop jailbreaks to further their own ambitions and fame.

The modus operandi of all these hackers was notoriety – becoming known to the masses – which became an intoxicating motivation. By late 2009, other hackers had developed viruses and exploits to jailbroken iPhones. These exploits invaded the provider's network to seek out and find jailbroken iPhones. This was a concern that Apple addressed in its counter to the Electronic Freedom Foundation's claim to allow jailbreaking as an exception to the DMCA (*Digital Media Copyright Act*).

The Library of Congress decided that jailbreaking your phone was an exception. However, the deciders of this policy didn't take into account the increase of threats that would invade AT&T and Apple. So Apple and AT&T would have to protect their networks and OS. Since the release of the first Apple mobile device, Apple and the hackers have played a cat-and-mouse game. The first jailbreaks were crude and were prone to crashing the phone and making the iPhone nonfunctional, otherwise known as "bricking" the phone. Some of the jailbreaks and unlocks had the following monikers: Pwnage, Qwkpwn, RedSn0w, YellowSn0w, iLiberTy, Purplera1n, Blackra1n and Greenpois0n.

All circumvented the security measures of the iPhone by either replacing the OS with one engineered on user-created firmware, or just patching the kernel and/or bootrom, which allowed the device to run unsigned code.

THE IOS FILE SYSTEM

HFS+ FILE SYSTEM

In 1996, Apple developed a new file system that would accommodate storing large data sets. As physical disk size was increasing at breakneck speed, a file system had to be developed to support the growing need for storage. Hence, Apple developed the *Hierarchical File System* (HFS). The structure of HFS can be complicated to understand. At the physical level, the disks formatted with HFS are in 512 – byte blocks. These are similar to Windows-based sectors. There are two types of blocks on an HFS system: logical blocks and allocation blocks. The logical blocks are numbered from the first to the last on a given volume. They are static and are the same size as the physical blocks, 512 bytes. Allocation blocks are groups of logical blocks used by the HFS system to track data in a more efficient way. To reduce fragmentation on an HFS volume, groups of allocation blocks are tied together as clumps. This organization is shown in Figure 2.

- The first 1024 bytes: Reserved for boot blocks;
- Volume header: The next 1024 bytes are for the volume header, which contains information in regards to the structure of the HFS volume. There is a backup volume header at the last 1024 bytes of the HFS volume. There are also volume header signatures. HFS plus the volume header signature is seen as "H+." For HFSX it is "HX";
- Allocation file: The allocation file simply tracks which allocation blocks are in use by the file system;
- Extents overflow file: This tracks all the allocation blocks that belong to a file's data forks. The contains a list of all extents used by a file and the associated blocks in the appropriate order;

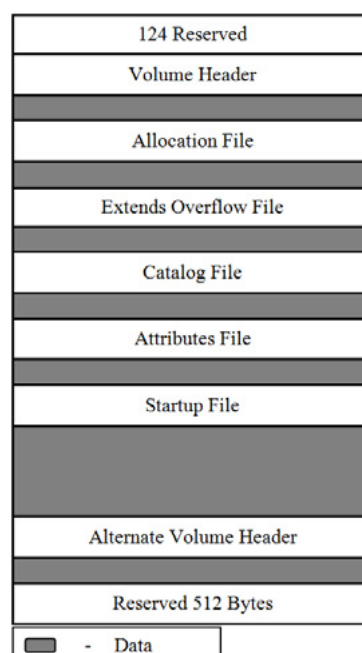


Figure 2. The structure of an HFS+ file system

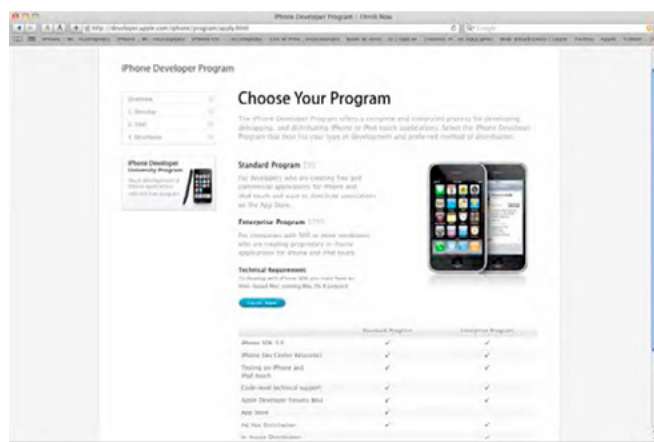


Figure 1. iPhone developer program

- **Catalog file:** The HFS+ file system uses a catalog file system to maintain all the information in regards to files and folders within a volume. These are in a hierarchical system of nodes: Header node (The location of the Header node is tracked in the volume header. Within that, the catalog ID number is stored as well. This number is assigned by the catalog file, which gets the next number from the volume header that tracks the last number assigned. The catalog file will increment that number by one and assign it to that file, and is in turn store in the Header node. Attributes file: This file is reserved for future use of data forks; Startup file: This file was designed to assist in booting a system that did not have built-in ROM support; After the startup file: Where all the data in a volume is stored and tracked by the file system; Alternate volume header: A back-up of the volume header and is primarily used for disk repair; The last 512 bytes: Reserved.), Index node, Leaf nodes and Map nodes.

In terms of date and time, Apple has used absolute time, otherwise known as local time. UNIX time is used as well. The iOS system utilizes both of these

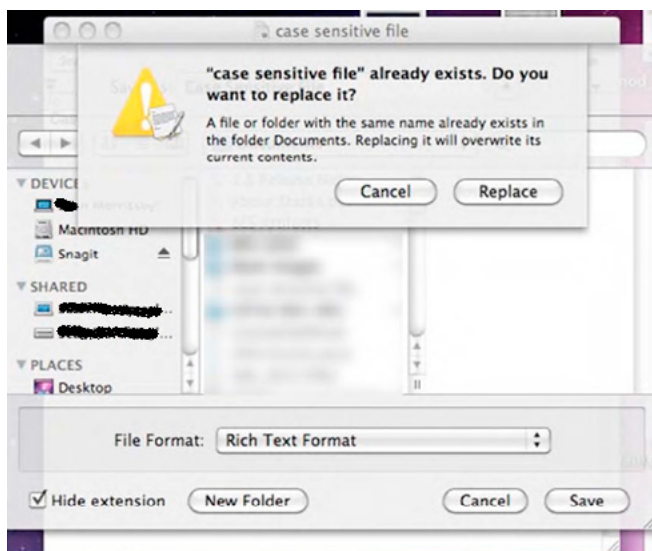


Figure 3. Error message from saving on an HFS+ system

AppleTV "OSBoot" Partition 900MB	iPhone OS Partition Partition 300-500MB
AppleTV "Media" Partition 35.5 – 155.5 GB	iPhone Data Partition Partition 7 – 31 GB

Figure 4. The similarities between the iPhone and Apple TV

time schemes. Since absolute time does not take into account the differences in time zones, one must be cognizant to identify the location of the system to understand actual the data and time of artifacts.

Data within the HFS file system utilizes a catalog file system or B*tree (balanced tree) to organize files. This balanced tree uses a catalog file and extents overflows in its organization scheme. B*trees are comprised of nodes. These nodes are grouped together in linear fashion, which makes data access faster. When data is added or deleted, the extents are constantly balanced to keep its efficiency. Each file that is created on an HFS file system is given a unique number – a catalog ID number. The HFS volume header tracks the numbering of the catalog ID and will increment by one each file added. These numbers can be reused, but this is tracked by the HFS volume header.

Typically, the reuse of catalog ID numbers is mainly seen in server environments, where large numbers of files are created. This number is consistently used to bind each node together in a file.

THE HFSX FILE SYSTEM

All Apple mobile devices use HFSX as the file system. HFSX is a variation of HFS+ with one major difference. HFSX is case sensitive. This means that two files on the file system can have the exact same name – but the case sensitivity is what allows the file system to differentiate between the two. For example: Case sensitive.doc / Case Sensitive.doc

Both of these files can exist on a HFSX file system. On OS X on a desktop or laptop, the following error occurs when the two file names with different cases are attempted to be saved. If the same were attempted on an HFS+ system, the following error will be seen, as shown in Figure 3.

IPHONE PARTITION AND VOLUME INFORMATION

The partition and volumes of the iPhone also have some history to them. Apple TV, another product of Apple, also came out with a scaled-down version of OS X. It had only one user and two partitions – an OS and data partition. Like the iPhone, Apple TV was designed to hold multimedia and access the Internet and iTunes. AppleTV appears to be a project test bed for HFSX for Apple and the use of a jailed system. Today the new AppleTV now utilizes the HFSX and jailed system of iOS 4. Figure 4 demonstrates the similarities between the iPhone and Apple TV.

IPHONE PARTITION INFORMATION ACQUISITION

Using two tools on the Mac from the command line, we can see the partition structure of the iPhone. Hdiutil

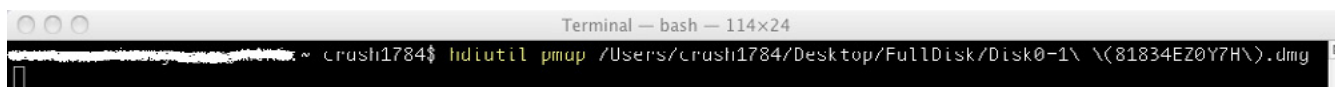


Figure 5. Steps to acquire partition information on the iPhone

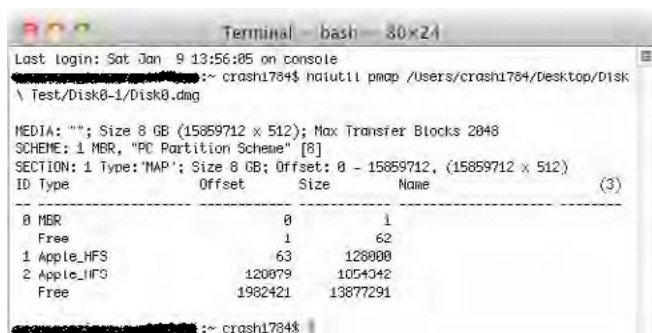


Figure 6. Output of the partition acquisition

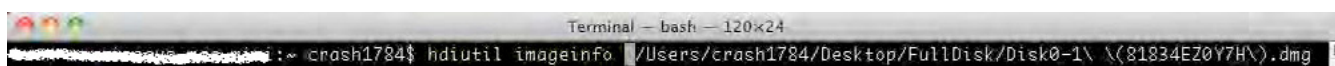


Figure 7. Command hdiutil imageinfo

is a command-line binary that is already on the Mac, and there are the following switches, pmmap and imageinfo, which can give the picture of the iPhone.

Hdiutil is a great program for looking at the structure of an iOS system. Hdiutil with the option pmmap gives an overall view of the partitioning scheme on a device. Hdiutil with the option imageinfo gives a granular look at each partition and information in regards to each. To acquire iPhone partition information:

- Open the Terminal application;
- Navigate to /Applications/Utilities/Terminal. From the command line, type: hdiutil pmmap, and then

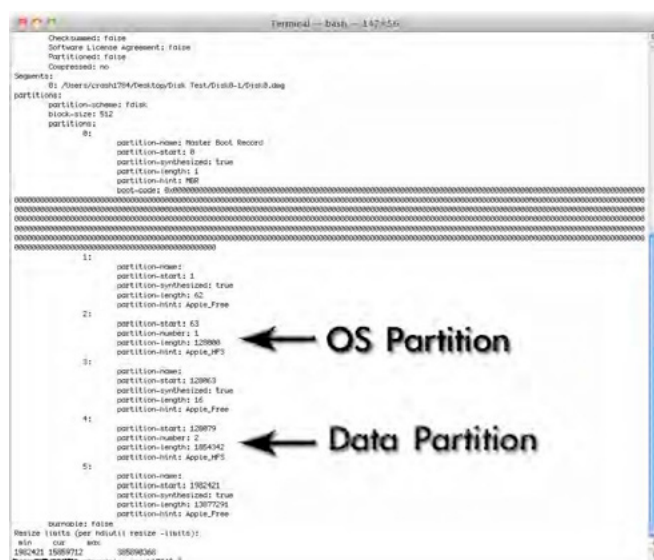


Figure 8. Output of the command hdiutil imageinfo

ID	Type	Offset	Size	Name	(3)
0	MBR	0	1		
	Free	1	62		
1	Apple_HFS	63	128000		
2	Apple_HFS	128079	1854342		
	Free	1982421	13877291		

Figure 9. Hdiutil reports as the start of each partition

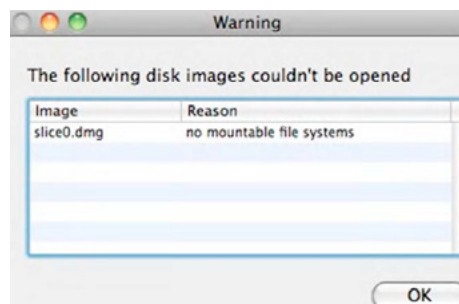


Figure 10. Error produced when a RAW image of Disk0 is in the mounting process

drag and drop an image of the iPhone from the finder to the terminal and press Enter, as depicted in Figure 5. You'll see the output shown in Figure 6.

- Next, from the terminal, type the command: hdiutil imageinfo, and then drag and drop a raw disk image or .dmg and press Enter, as shown in Figure 7. You'll see the output shown in Figure 8.

The previous two images show the partition scheme of the Apple iPhone OS. However, the information from hdiutil is incorrect. If the image were correct, Mac OS would be able to mount the iPhone image. If we look at what hdiutil reports as the start of each partition, as shown in Figure 9, the answer becomes clear.

When OS X attempts to mount this volume it sees the first HFS volume at sector 63 and the second HFS volume at 128079. The actual starting sector is as follows: the OS volume header is at sector 504 and the data volume header is at sector 1024632. It is because of the offsets of these volumes that even a Mac cannot mount a Disk0 (the complete raw image of the physical disk) image properly. The disk utility can mount images of either the OS partition (Disk0s1) or data partition (Disk0s2) themselves, without any errors. When a raw image of Disk0 is in the process of mounting, the following error shown in Figure 10 occurs.

However, if the gathered .dmg of the whole raw disk was copied, the offsets can be corrected and the image can be mounted properly. Creating a plug-in for MacFUSE can assist in allowing the Mac OS to properly mount the complete Disk0. Information in regards to creating a plug-in can be found at <http://code.google.com/p/macfuse>.

OS PARTITION

The OS partition is a read-only volume. This can be seen by following the path located at `private/etc/fstab`. Open the `fstab` file with TextEdit, and the following information is then shown in Figure 11.

As on all Macs, the partitions are divided into disks and slices. The RAW disk is “Disk0.” There is only one disk on the iPhone, hence you see Disk0. The OS partition is “Disk0s1” and the Data partition is “Disk0s2.” Next you see both partitions from Figure 11, and the `/dev/disk0s1` and then `/hfs` denoting an HFS volume after that. Next to `hfs` is `ro`. This means that the volume is read-only. The data partition `/dev/Disk0s2` is a read/write HFS volume. Due to the fact that the system partition is read-only, all the data that is on this volume is usually non-evidentiary unless the phone has been jailbroken. The relevance of this file is that if you see `/dev/disk0s1 /hfs rw`, the system has been jailbroken. This is a good artifact to use to validate if an imaging process has tampered with the UNIX jail of the iDevice system.

IOS SYSTEM PARTITION

The system partition shown in Figure 12 is of the iOS device described in Table 1. The contents of this partition are usually non-evidentiary; however, sometimes an examination could be necessary.



Figure 11. Opening the `fstab` file in TextEdit

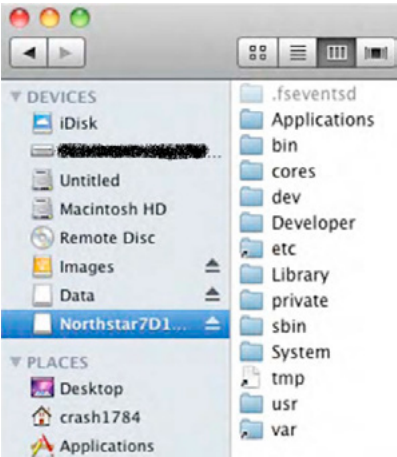


Figure 12. iOS system partition



Figure 13. The password `Alpine`

The path `private/etc/passwd` is the password file of the OS. Tools like John the Ripper, which can be downloaded at www.openwall.com/john/, allow for cracking the root and mobile passwords. The root and mobile passwords are encrypted using a DES algorithm that requires a 2 character salt key and an 8 character text password, which yields an 11

Table 1. System partition of the iOS device

Directory	Description
Application	Has symbolic links that point to the <code>/var/stacsh</code> directory
Etc	Has a symbolic link to <code>/private/etc</code>
Tmp	Has a symbolic link to
User	Has a symbolic link
Var	Has a symbolic link to <code>/private/var</code>
Damaged files	Can contain artifacts of a previous jailbreak
Bin	Contains one command-line binary, <code>launchctl</code>
Cores	Empty
Dev	Empty
Developer	Empty
Library	As with any OS X system, contains system plug-ins and settings: Application support: Bluetooth models and PIN codes Audio: Contains the audio plug-in Cashes: Empty File systems: Empty Internet Plug: Empty LaunchAgents: Empty LaunchDaemons: Empty Manager Preferences: Contains a symbolic link to Mobile Printers: Empty Ringtones: Contains system-installed ringtones Updates: Empty Wallpaper: Contains numerous PNG files and thumbnails (non-evidentiary)
private	Contains the <code>Etc</code> and <code>Var</code> folders: <code>Etc</code> : Contains <code>fstab.master.passwd</code> , <code>passwd</code> files (both master and <code>passwd</code> : same) <code>Var</code> : Empty
sbin	Contains command-line binaries
System	Library folder that contains system preferences and settings; includes <code>/System/Library/CoreService/SystemVersion.plist</code> : Firmware Version
Usr	Contains more command-line binaries and time zone data

character value. With jailbroken iPhones, a more advanced user can change these passwords. A password for root that has never changed since the first iPhone is “Alpine,” as shown in Figure 13.

Due to the design of the iPhone, there are procedures that can break the phone or use copyrighted software to bypass the security measures in order to image an iPhone. As will be discussed in this article, there are numerous areas of investigation that will maintain the integrity of the evidence and

Table 2. *iOS version and corresponding volume name*

iOS Version	Volume Name	iOS Version	Volume Name
1.00	Alpine 1A420	3.1.2	Northstar 7D11
1.0.0	Heavenly 1A543a	3.1.3	SUNorthstarTwo 7E18
1.0.1	Heavenly 1C25	2.00	Big Bear 5A345
1.0.2	Heavenly 1C28	2.00	Big Bear 5A347
1.1.1	Snowbird 3A109a	2.0.1	Beg Bear 5B108
1.1.2	Oktoberfest 3B48b	2.0.2	Big Bear 5C1
1.1.3	Little Bear 4A93	2.1	Sugar Bowl 5F136
1.1.4	Little Bear 4A102	2.2	Timberline 5G77
2	Big Bear 5A347	2.2.1	SUTimberline 5H11
2.0.1	Big Bear 5B108	3.00	Kirkwood 7A341
2.0.2	Big bear 5C1	3.0.1	Kirkwood 7A400
2.1	Sugar Bowl 5F136	3.1	Northstar 7C144
2.2	Timberline 5G77	3.1.2	Northstar 7D11
2.2.1	SYTimberline 5H11	3.1.3	SUNorthstarTwo 7E18
3	Kirkwood 7A341	3.2	Wildcat7B367
3.0.1	Kirkwood 7A400	4.0	Apex8A306
3.1	Northstar 7C144	4.1	Baker8B177

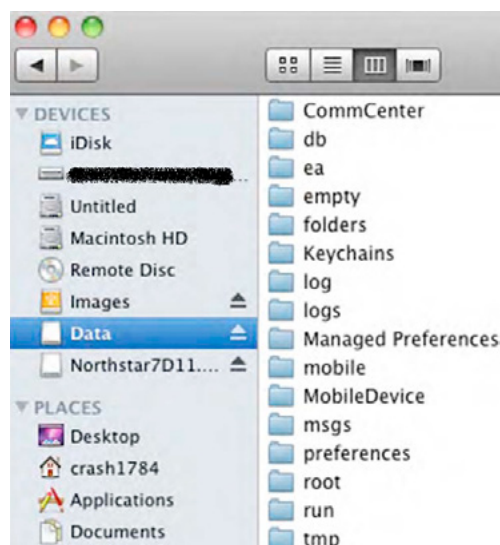


Figure 14. *Data partition directory structure*

still locate valuable artifacts and secure convictions. For each firmware version, the OS partition has volume names that correspond to the iOS version. Table 2 shows the iOS version (from 1.00 to 4.1) and the corresponding volume name of the OS system partitions.

IOS DATA PARTITION

Over the years, there has been little change in the makeup of this data partition. You can see some of the changes in the file system from logical acquisitions. The bulk of the evidence that can be acquired from this device comes from the read/write partition, also known the data partition, as shown

Figure 15. *The ROWID, address, date, text and flags*

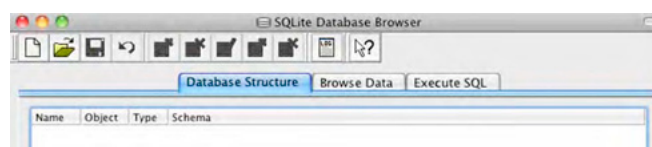


Figure 16. *The Interface of the SQLite Database Browser Application*

Table 3. *Directories and corresponding items of interest*

Directory	Items of Interest
CommCenter	No artifacts
Dhcpclient	One plist that contains the last IP address and router information for that device
db	No artifacts
Ea	Empty
Folders	Empty
Keychains	Keychain.db, which contains user passwords from various applications
Log	Empty
Logs	General.log: The OS version and serial number Lockdown.log: Lockdown daemon log
Manager Preferences	Empty
Mobile	Bulk of the user data
MobileDevice	Empty
Preferebces	System configuration: Network artifacts backed up
Root	Caches: GPS location information Lockdown: Pairing certificates Preferences: No artifacts
Run	System log
tmp	Manifest.plist: plist backup
Vm	Empty

in Figure 14 and the Table 3 shows the directories and accompanying items of interest. The data partition is riddled with a lot of information that will assist in any investigation. When an Apple device

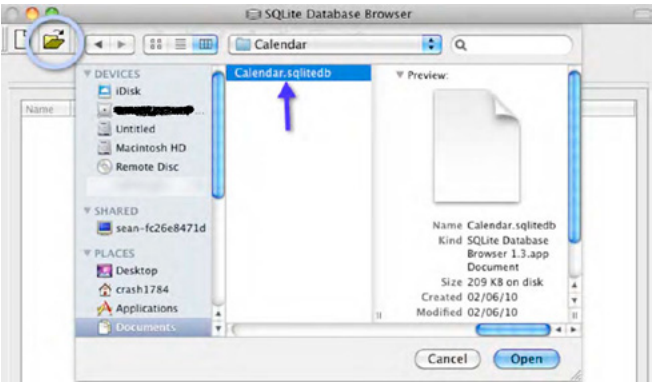


Figure 17. Adding SQLite database browser

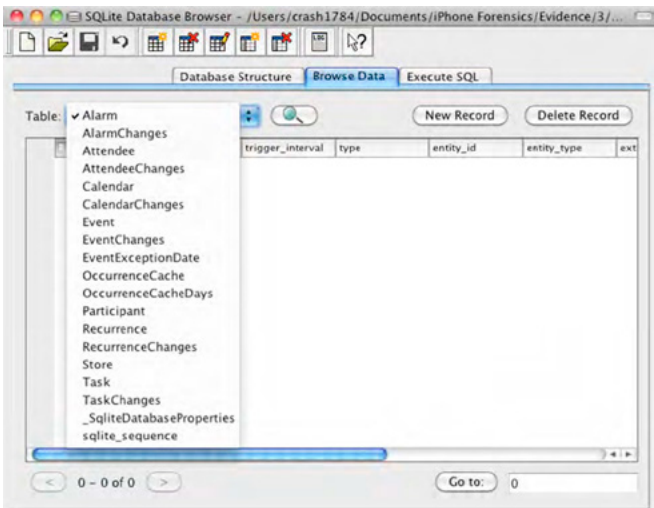


Figure 18. Moving to the browse data tab and picking the table to review

Table 4. Artifacts organized by directory and whether they are in backup

Directory	In Backup	Artifact
Mobile/Application	+	Plists, SQLite databases
Library/AddressBook	+	Contacts and images
Library/Caches		SQLite database: MapTiles
Library/Calendar	+	SQLite database: Events
Library/CallHistory	+	SQLite database: Call logs
Library/Carrier Bundles		Carrier information
Library/Caches/Com.apple.itunesstored		iTunes purchase information
Library/ConfigurationProfiles	+	Plist password history
Library/Cookies	+	Plist: Internet cookies
Library/DataAccess	+	E-mail account information
Library/Keyboard	+	.dat file: Dynamic text
Library/Logs	+	Log files
Library/Mail	+	In Logical Data, no artifacts
Library/Maps	+	Plist: Bookmarks, directions, history
Library/MobileInstallation	+	Applications that use Locations
Library/Notes	+	SQLite database: Notes
Library/Preferences	+	Plist System and user settings
Library/RemoteNotification	+	Plist: Apps that have push notification
Library/Safari	+	Plist: Bookmarks, history
Library/SafeHarbour		Location of where app data is stored
Library/SMS	+	SMS and MMS data
Library/Voicemail	+	.arm files: Voice messages
Library/Webclips		
Library/Webkit	+	SQLite databases: Gmail account info, caches e-mail messages
Media/DCIM	+	iPhone camera photos
Media/PhotoData	+	Additional photo information and thumbnails
Media /iTunes _Control		Music and video from iTunes
Media/Books		Books from the iBookstore and synced PDFs

gets backed up from iTunes, it gathers information from the Mobile directory. Table 4 shows all the artifacts that are acquired logically and items that are also stored as backups on a Mac or PC.

SQLITE DATABASES

The iPhone OS uses the SQLite database format to store information on the phone. An examination of the logical extraction shows numerous SQLite databases for the operation of the phone and by developers of applications. The iPhone also uses these databases to cross-reference information from one database to the other, which gets displayed on the UI. These databases interact with each other to give the user an informative experience. The big three databases are the Address Book, SMS, and Call History databases.

Table 5. *The address book database*

Table	Relevant Data
AB Group	Group information
ABGroupChanges	Non-evidentiary
ABGroupMembers	Contacts associated each group
ABMultiValue	When a contact has multiple values, phone numbers, e-mail address books, company URLs, etc.
ABMultiValueEntry	Street addresses for contacts
ABMultiValueEntryKey	Non-evidentiary
ABMultiValueLabel	Non-evidentiary
ABPerson	Name, organization, department, notes, etc.
ABPersonChanges	Non-evidentiary
ABPersonMultiValueDeletes	Non-evidentiary
ABPersonSearchKey	Non-evidentiary
ABPersonSearchKey	Non-evidentiary
ABPhoneLastFour	Non-evidentiary
ABRecent	Recently used e-mail addresses
ABStore	Non-evidentiary
FirstSortSectionCount	Non-evidentiary
FirstSortSectionCount	Non-evidentiary
_SqliteDatabase Properties	Non-evidentiary
Sqlite _ sequence	Non-evidentiary (but contains good information on the structure of the database)

ADDRESS BOOK DATABASE

This database has 18 tables. Table 5 provides the information that would be relevant in an investigation.

Table 6. *The tables and relevant data of the SMS database*

Table	Relevant Data
_SqliteDatabase Properties	Contains database properties (non-evidentiary)
Group _ member	Assigns an incoming text a group ID that then will pull all the text messages from the iPhone owner and the party having the conversation
Message	Contains the content of the message, date and time, and whether the message was sent or received; also lists the associated group ID
Msg _ group	Gives the group ID and ID of the last message in that group
Msg _ Pieces	Tracks all MMS messages
Sqlite _ sequence	Provides a sequential list of all tables in the database

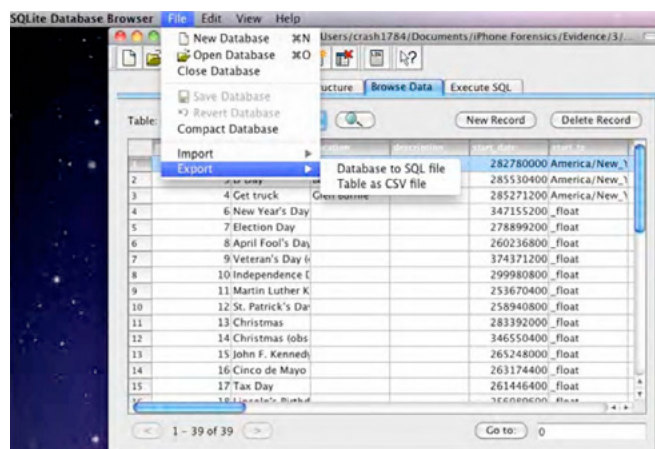


Figure 19. *The CSV format can be opened in other Applications*



Figure 20. *The Froq interface*

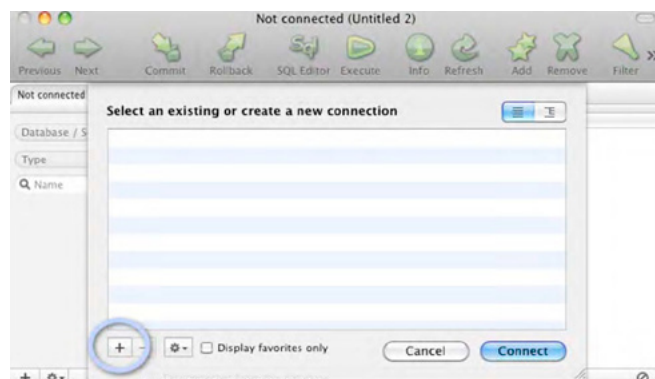


Figure 21. *Creating a new connection*

SMS DATABASE

The SMS database is the container that keeps records of text messages sent and received by the Messages application. Table 6 shows the tables that make up this database.

In Figure 15, you can see the ROWID (row identification), which is a number for the message, the address (the phone number that the text came from), and the date and time of the text. The date and time values are in Unix time and can be converted using several free tools. The flags are for sent and received text messages.

Table 7. Tables and relevant data artifacts

Table	Relevant Data
SqliteDatabase Properties	
Call	Contains phone numbers, date and time info, and the duration of the call; also flags incoming, outgoing, and missed calls, and calls that have voicemails
Data	Tracks the number of bytes the iPhone has sent and received
Sqlite_sequence	Contains a sequential list of tabled in the database

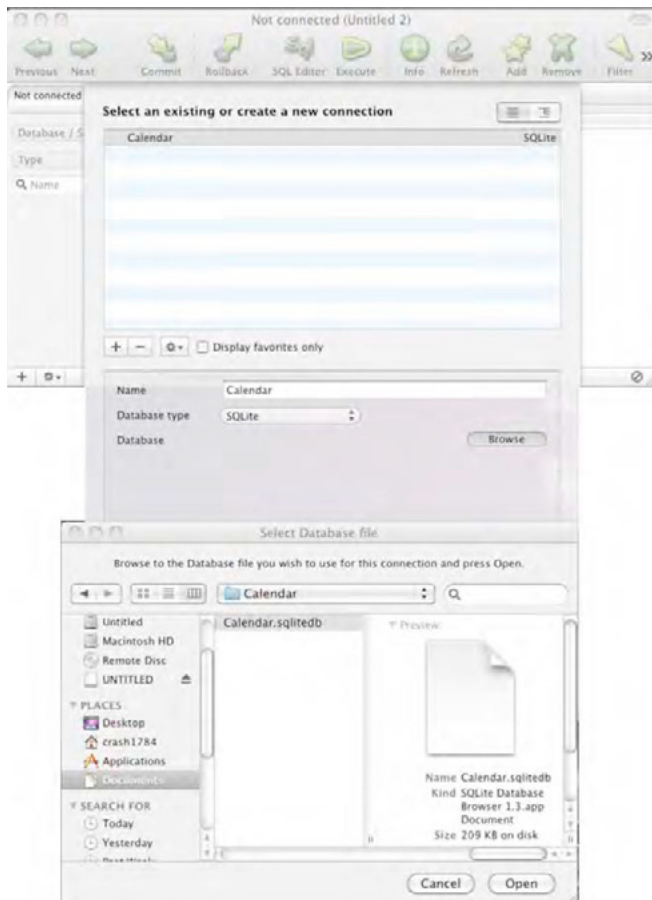


Figure 22. Selecting SQLite as the database type and browsing to the relevant one

CALL HISTORY DATABASE

The Call History database is a simpler database, and the only one that has restrictions – it will only hold 100 calls. The Address Book database is the hub of a lot of other applications on the iDevice. A lot of data correlation occurs between this database and others. For example, the Call History database correlates the numbers from the sent and/or received call with the names associated with those numbers in the Address Book database. Table 7 describes the tables and artifacts of relevance.

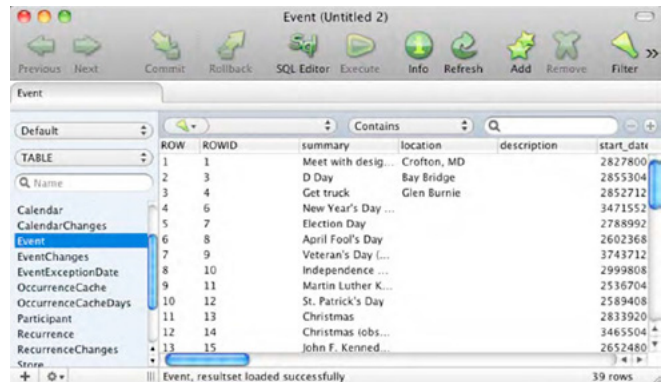


Figure 23. Database brought into Froq for analysis

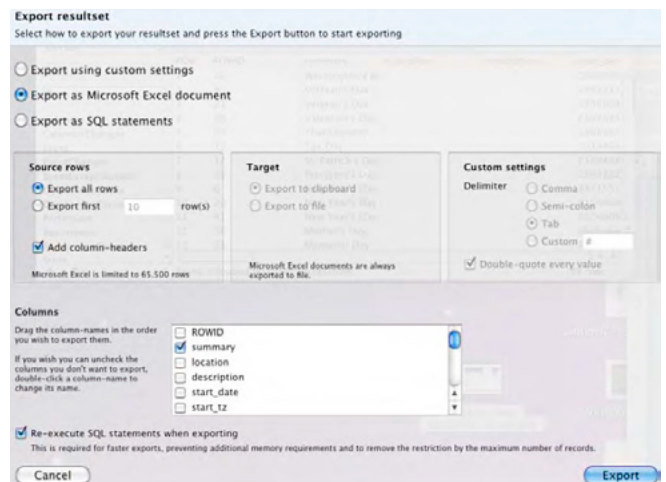


Figure 24. "Export Resultset" screen

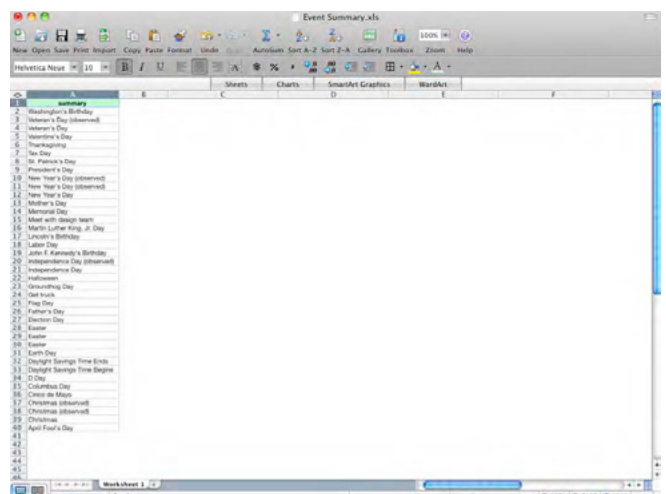


Figure 25. Exported data viewed in Excel

RETRIEVING DATA FROM SQLITE DATABASES

There are applications that can assist in extracting data from SQLite databases that can be used in other applications or tools. One of these SQLite database applications is SQLite Database Browser. The interface of this application is shown in Figure 16.

To add SQLite Database Browser, click the Open icon and navigate to the relevant database, as shown in Figure 17.

After the relevant database is brought into SQLite Database Browser, one can browse through the tables in the database. First move to the Browse Data tab and then pick the table to review from the Table drop-down list. This is shown in Figure 18.

The data can be exported from SQLite Database Browser to a CSV (Comma Separated Value) format, which in turn can be opened with applications such as Microsoft Excel, as shown in Figure 19.

Another application worth mentioning is Froq, developed by Alwin Troost. This application is pro-

prietary and can be purchased at www.alwintroost.nl/?id=82. This application has a lot of functionality and is an excellent tool for viewing the tables of a database and exporting the portions of the database needed for a given investigation. The interface of Froq is shown in Figure 20. To view a database of interest, perform the following steps:

- Go to the Froq menu bar and select connect | connect;
- The next box will ask you to select an existing connection or create a new one. Select a new connection by clicking the +, as shown in Figure 21;
- In the expanded window, give the connection a name – for example, Calendar;
- For the database type, select SQLite;
- From the Browse tab, navigate to the relevant database. (Steps 4 and 5 are shown in Figure 22);
- Then the database will be brought into Froq for analysis. The tables can be selected from the left pane, and the data can be seen in the right pane, as shown in Figure 23. To export data from this application, return to the top toolbar;
- Select Resultset | Export;

Table 8. Property lists and relevant data directory property lists and artifacts

Directory	Property Lists and Artifacts
Db	
Keychain	
Managed preferences	Com.apple.sprongboard.plist: Add artifact
Mobile/Library/Cookies	Cookies.plist: Web-related artifacts
Mobile/Library/Mail	Accounts.plists: E-mail accounts Metadata.plist: Dates and times of e-mail puuls
Mobile/Library.Maps	Bookmarks.plist: Map bookmarks created by the user History.plist: All routes and searches
Mobile/Library/Preferences	Com, apple.BTserver,airplane.plist: Shows that airplane mode was initiated on the device for Bluetooth Com.apple.commcenter.plist: Stores ICCID and IMSI numbers Com.apple.maps.plist: Recent map searches and last latitude and longitude of last map tile seen Com.apple.mobilephone.settings.plist: Call-forwarding numbers Com.apple.mobilephone.speeddial.plist: All favorite contacts for speed dial Com.apple.mobilesafari.plist: Recent Safari searches Com.apple.MobileSMS.plist: Any unset SMS messages Com.apple.mobiletimer.plist: List of world clocks used Com.apple.preference.plist: Keyboard language last used Com.apple.springboard.plist: Lists of apps that are shown in the interface, password protection flag, wipe enable settings, last system version Com.apple.weather.plist: Cities for weather reports, date and time of last update Com.apple.youtube.plist: URLs of all videos bookmarked, history of all video watched, videos searched by user
Library/Safari	Bookmarks.plist: all Internet bookmarks – created and standard History.plist: Web browsing history Suspendedstate.plist: Web page title ans URL of all suspended web paged that are held in the background so that users can jump from one page to another easily (a maximum of eight pages can be saved at one time)

- There are three types of settings: Custom, export as an excel spreadsheet, or as SQL statements;
- Under the columns, you can be as granular as necessary for the data that is required. For example, select “Export as Microsoft Excel document.” Then select the “Export all rows” radio button from the “Source rows” section, and select the columns needed;
- Then select “Export.” The resulting screen is shown in Figure 24;
- After the data is exported, it can be viewed in Excel, as shown in Figure 25.

PROPERTY LISTS

Property lists are XML files that are commonly seen in standard OS X systems. Since iOS is a modified OS X system, it stands to reason that we will also see property lists within the directory structure. The iOS data partition is riddled with property lists that can contain valuable information. Table 8 shows the property lists that contain data of relevance.

VIEWING PROPERTY LISTS

Apple has given examiners a free tool to view property lists, the Property List Editor (also known

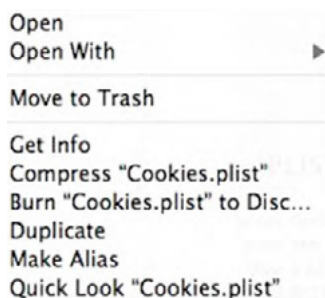


Figure 26. Select “Get Info” from this drop-down menu

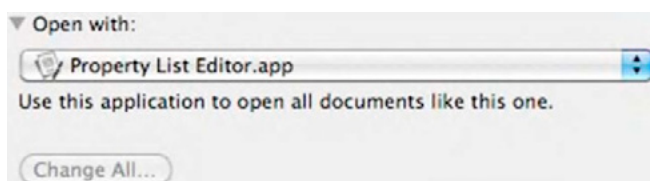


Figure 27. Expand the “Open with” portion of the window

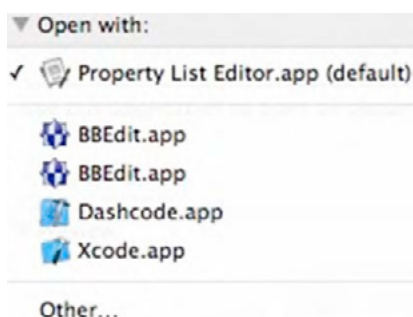


Figure 28. Select other

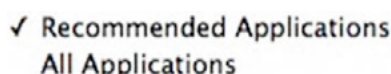


Figure 29. Change “Recommended Applications” to “All Applications”

as the plist) The Property List Editor is part of the developer tools, and is an optional install on the OS X installation disk. The newest versions can be downloaded from the Apple Developers web site, at <http://developer.apple.com/technologies/tools>. The Property List Editor can display these XML-formatted files in a readable manner, similar to how they are viewed on a Windows system (i.e., not in their raw form). Once the Property List Editor has either been installed from the OS X disk or downloaded from the Internet, the following steps can be followed to view a given property list:

- Navigate to /Developer/Applications/Utilities/Property List Editor;
- Double-click the application;
- From the Property list file menu, select Open;
- Next, navigate to the location of the plist you wish to view;
- Select the plist;
- Press the Open button;
- View the artifacts from the plist editor interface.

The one thing that detracts from this free tool is the way it reports the artifacts. One can grab screenshots of the relevant data and add those images to a report. There is another application, OmniOutliner 3, an app bundled with OS X 1.4 (Tiger). It is a for-pay app, and it’s available at www.omnigroup.com/products/omnioutliner. You can use this tool to view plists easily bring them into an existing report. The following describes how to view and report plists with OmniOutliner 3.

First you have to set up your Mac so that you can automatically open all plists with OmniOutliner.

- From Finder, find any plist on your volume (Library/Preferences is a good choice);
- Right-click the plist;

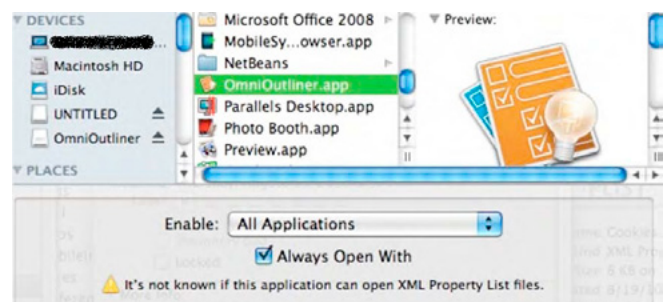


Figure 30. Select “Always Open With”

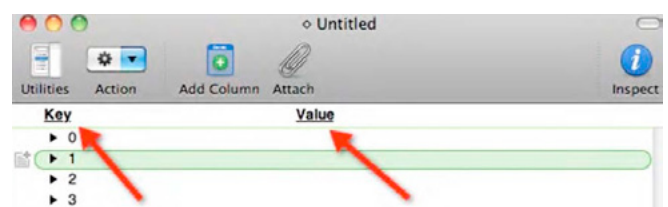


Figure 31. Separate key and value columns

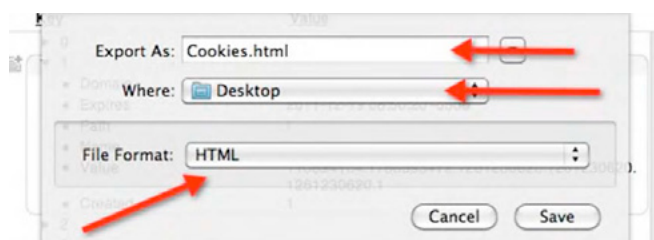


Figure 32. Choose a file name, where to save the file and the file format

- Select Get Info, as shown in Figure 26;
- From the Get Info dialog box, expand the “Open with” portion of the window (shown in Figure 27);
- Now click the drop-down list and select Other, as shown in Figure 28;
- The next window will be another finder window in the application directory. You will have to change Recommended Applications to All Applications, as shown in Figure 29;
- Then locate OmniOutliner and highlight the application;
- Then select the Always Open With box, and click the Add button, as shown in Figure 30 (all property lists will automatically open with OmniOutliner instead of the Property List Editor. If you wish to switch back to the Property List Editor, repeat the same steps, but select

Property List Editor instead. Now that you have switched to OmniOutliner, the next steps will go through using OmniOutliner);

- Select a property list to examine and double-click the file. OmniOutliner will automatically open the plist.
- The values are separated into Key and Value columns, as shown in Figure 31.
- To expand all the keys, go to the menu bar and select View | Expand All. Now you'll be able to view all the keys and values.
- To report data from Omni Outliner
 - Either expand all or just the items of relevance;
 - Then go to the menu bar and select File | Export;
 - Enter a file name, where you want the file saved, and what format to export it in, as shown in Figure 32.

CONCLUSIONS

The iOS operating and file systems have changed since its introduction in 2007. Since then the Apple device family has expanded and changed the way we communicate and now how we compute, it is important to understand the inner workings of the devices to intelligently articulate some of the processes that are accomplished to facilitate artifact extraction. As shown in this article, there can be a mountain of data that can be captured from the devices. In this article, we reviewed the history of the iOS operating and file system, and artifacts that reside in the system and data partitions. We also looked at tools that can examine many of the artifacts that are on any iDevice. As we saw, most of the evidence on the iDevice is stored in SQLite databases and property lists.

REFERENCES

- Elmer-Dewitt, P. (2008, May, 16). iPhone Rollout: 42 Countries, 575 million potential customers. Fortune. Retrieved March 30, 2009 from <http://apple20.blogs.fortune.cnn.com/2008/05/16/iphone-rollout-42-countries-575-million-potential-customers/>
- Farley, T. (2007). The Cell-Phone Revolution. American Heritage of Invention and Technology. Retrieved March 24, 2009, from www.americanheritage.com/events/articles/web/20070110-cell-phone-att-mobile-phone-motorola-federal-communications-commission-cdma-tdmagsm.shtml.
- Fletcher, F. E., & Mow, L. C. (2002). What's happening with E-911? The Voice of Technology. Retrieved April 2, 2009, from www.drinkerbiddle.com/files/Publication/d6e48706-e421-411c-ab6f-b4fa132be026/Presentation/PublicationAttachment/fdb0980a-7abf-40bf-a9cd-1b7f9c64f3c7/WhatHappeningWithE911.pdf
- Hafner, K. (2007, July 6). iPhone futures turn out to be a risky investment. The New York Times, p. C3.
- Henderson, S. (2006). Learning from all fifty states: how to apply the fourth amendment and its state analogs to protect third party information from unreasonable search. The Catholic University Law Review, 55, 373.
- Kerr, O. (2004). The fourth amendment and new technologies: constitutional myths and the case for caution. Michigan Law Review, 102, 801.
- Krazit, T. (2009). Apple ready for third generation iPhone. Retrieved March 30, 2009, from <http://news.cent.com/apple-ready-for-third-generation-of-iphone/>
- Morrissey, Sean. (2010) iOS Forensic Analysis: for iPhone, iPad and iPod Touch. New York, NY: Apress.
- Roberts, M. (2007, July 25). AT&T profit soars: iPhone gives cell provider a boost. Augusta Chronicle, p. B11.
- Stillwagon, B. (2008). Bringing an end to warrantless cell phone searches. Georgia Law Review, 42, 1165.
- Walsh, D., & Finz, S. (2004, August 26). The Peterson trial: defendant lied often, recorded calls show, supporters mislead about whereabouts. San Francisco Chronicle, p. B1.

ABOUT THE AUTHOR



Devison Pinheiro Franco is Graduated in Data Processing. Specialist in Computer Networks, in Computer Networks Support and in Forensic Sciences (Emphasis in Forensic Computing). Security Analyst of Bank of Amazônia. Professor at various colleges and universities of disciplines like: Computer Forensics, Information Security, Systems Audit, Computer Networks, Computer Architecture and Operating Systems. Computer Forensic Expert, IT Auditor and Pentester with the following certifications: CEH – Certified Ethical Hacker, CHFI – Certified Hacking Forensic Investigator, DSEH – Data Security Ethical Hacker, DSFE – Data Security Forensics Examiner, DSO – Data Security Officer and ISO/IEC 27002 Foundation.

ABOUT THE AUTHOR



Nágila Magalhães Cardoso is graduated in Computer Networks Technology and Specialist in Computer Security. Certified in network administration and technical in computer installation, maintenance and installation of computer networks. Panelist and professor of free computer courses in the areas of information technology and computer networks, with special knowledge in computer security and forensics.

FOUR WINDOWS XP FORENSIC ANALYSIS TIPS & TRICKS

by Davide Barbato

When conducting forensics analysis of a Windows XP system, it must be taken into account some particular behaviors that can lead to misleading conclusions if not properly handled.

What you will learn:

- Specific Windows XP behaviors
- A basic knowledge of Windows LNK file structure

What you should know:

- A basic understanding of NTFS structure
- A basic understanding of Windows XP registry
- How to create and read timeline

Even if most of Windows based PCs and notebooks are shipped with Windows 7 or Windows 8, you could happen to deal with an old Windows XP operating system.

To an untrained eye, it could appear that Windows XP is just another Windows operating system family: It behaves completely different, and could lead to misleading conclusions if you are not familiar with XP. Think about a case in which you need to know if a user views a document or a folder, or opened a document and trashed them: Windows XP has different behavior in respect to Windows 7 and this need to be addressed.

NTFS DISABLE LAST ACCESS UPDATE

First of all, let's talk about the file system: even if Windows XP is really old, it's not so old to be shipped with FAT32 file system, so in this article we can assume that we are dealing with NTFS file system.

Based on that assumption, it is important and critical to remember that Windows XP, every time it reads a file or a directory, it changes the access time of \$SI object, updating on the time the system is accessing the object. This means that even listing the content of a directory will update the \$SI access time, losing the previous last access time.

This behavior can be avoided adding a Registry key, under *HKLM\SYSTEM\CurrentControlSet\Control\FileSystem*, named *NtfsDisableLastAccessUpdate* and setting its value to 1.

Some scenarios presented in this article will deal with that behavior, trying to show how and when the access timestamp is updated.

FOUR WINDOWS XP FORENSIC ANALYSIS TIPS & TRICKS

5936	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Programmi/Windows Messenger.lnk
5937	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Programmi/Windows Movie Maker.lnk
5938	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Windows Update.lnk
5939	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Catalogo di Windows.lnk
5940	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Menu Avvio/Programmi/Assistenza remota.lnk
5941	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Menu Avvio/Programmi/Outlook Express.lnk
5942	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Programmi/Adobe Reader 8.lnk
5943	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Menu Avvio/Programmi/Windows Media Player.lnk
5944	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/All Users/Menu Avvio/Impostazioni accesso ai programmi.lnk
5945	06/03/13 12:52:32	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Menu Avvio/Programmi/Internet Explorer.lnk
5946	06/03/13 12:52:34	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Programmi/File comuni/Adobe/Acrobat/ActiveX/pdfshell.dll

Figure 1. User clicks the Start icon

6066	06/03/13 12:56:21	UTC	NTFS \$MFT	\$\$I [MAC.] time	-	MALWARETESTENV	/WINDOWS/system32/wbem/Logs/wmiprolog
6067	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/synpress.pdf.lnk
6068	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/VM_Share_Folder su 'yboxsv' (E).lnk
6069	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/index.html.lnk
6070	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/html.lnk
6071	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/Read Me.txt.lnk
6072	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/MyHook_1.2.lnk
6073	06/03/13 12:57:20	UTC	NTFS \$MFT	\$\$I [A.] time	-	MALWARETESTENV	/Documents and Settings/mw/Recent/~credits.afx.txt.lnk

Figure 2. Recent menu folder

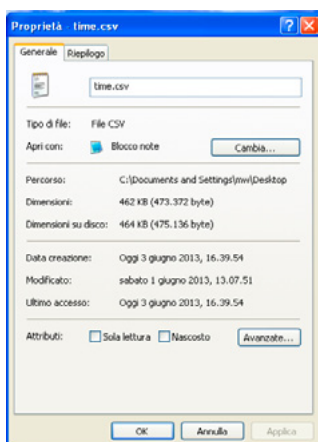


Figure 3. time.csv property

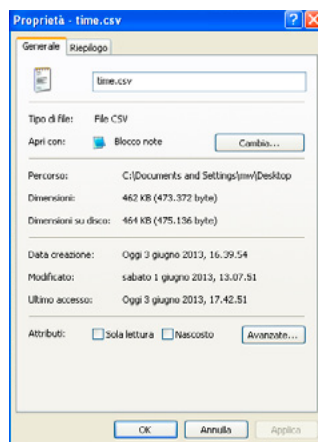


Figure 4. time.csv last access updated

01

OPENING WINDOWS MENU

What happens when a user clicks on the Start icon? What filesystem changes occurred? Here an excerpt of a timeline, presenting only the lines in which the user clicks the Start menu icon: Figure 1.

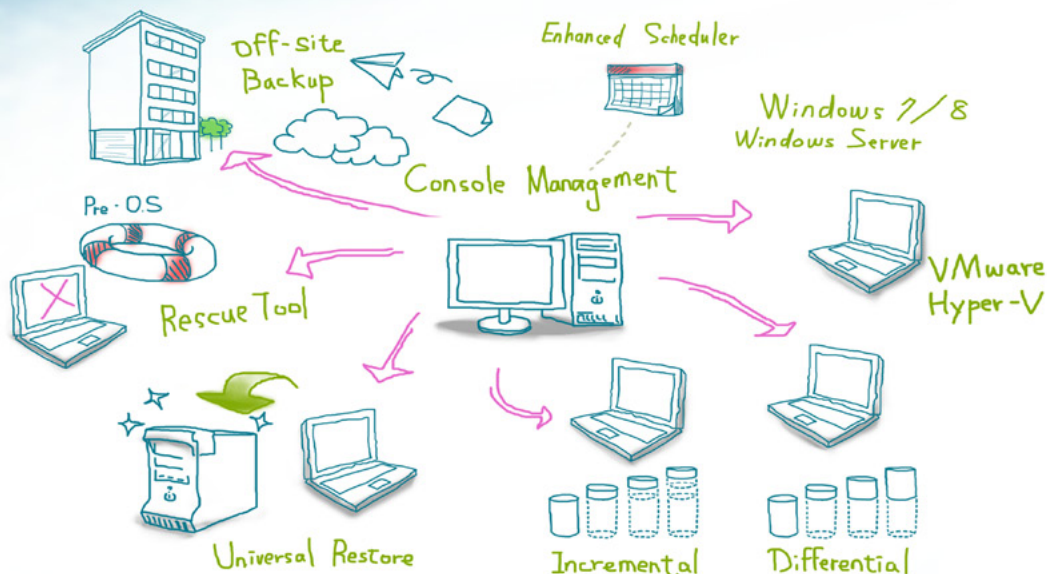
As you can see, Windows opens the menu directories (named "Menu Avvio") and reads its content, updating the \$\$I access timestamp.

Let's take a look at the Recent folder, under the start menu: when showing its content, Windows

a d v e r t i s e m e n t

FARSTONE®
Total Backup Recovery®

We make it easy for you.



updates the \$SI access timestamp, the same way as the menu items, listed above (Figure 2).

If you show the metadata information about the Ink files, you will see the access timestamp changed to 3 June 2013 at 12:57 UTC.

06/03/13 15:17:44	UTC	FILE	NTFS \$MFT \$SI [A..] time	MALWARETESTENV	/Programmi/Internet Explorer/ieplorer.exe
06/03/13 15:17:46	UTC	FILE	NTFS \$MFT \$SI [A..] time	MALWARETESTENV	/WINDOWS/AppPatch/aclayers.dll
06/03/13 15:31:00	UTC	FILE	NTFS \$MFT \$SI [A..] time	MALWARETESTENV	/Documents and Settings/All Users/Desktop/Scelta del browser.lnk
06/03/13 15:42:51	UTC	FILE	NTFS \$MFT \$SI [A..] time	MALWARETESTENV	/Documents and Settings/mw/Desktop/time.csv
06/03/13 15:43:43	UTC	FILE	NTFS \$MFT \$SI [A..] time	MALWARETESTENV	/WINDOWS/system32/wuaueng.dll

Figure 5. time.csv timeline

1553	06/04/13 20:38:25	NTFS \$MFT	\$FN [A..] time	/RECYCLER/S-1-5-21-854245398-1409082233-725345543-500/Dc1.doc
1554	06/04/13 20:38:25	NTFS \$MFT	\$SI [A..] time	/Programmi/Windows NT/Accessori/wordpad.exe
1555	06/04/13 20:38:25	NTUSER key	Last Written	Software/Microsoft/Windows/ShellNoRoam/MUICache
1556	06/04/13 20:38:26	NTUSER key	Last Written	Software/Microsoft/Windows/ShellNoRoam/BagMRU
1557	06/04/13 20:38:29	NTFS \$MFT	\$FN [C..] time	/RECYCLER/S-1-5-21-854245398-1409082233-725345543-500/Dc1.doc
1558	06/04/13 20:38:29	NTFS \$MFT	\$SI [C..] time	/Programmi/Windows NT/Accessori/wordpad.exe
1559	06/04/13 20:38:29	UserAssist key	Time of Launch	UEME_RUNPATH:C:/Programmi/Windows NT/Accessori/WORDPAD.EXE
1560	06/04/13 20:38:29	UserAssist key	Time of Launch	UEME_RUNPATH
1561	06/04/13 20:38:29	UserAssist key	Time of Launch	UEME_UISCUT
1562	06/04/13 20:38:30	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/spool/drivers/w32x86/3
1563	06/04/13 20:38:30	NTFS \$MFT	\$SI [MACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
1564	06/04/13 20:38:30	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
1565	06/04/13 20:38:30	NTFS \$MFT	\$SI [A..] time	/RECYCLER/S-1-5-21-854245398-1409082233-725345543-500/Dc1.doc
1566	06/04/13 20:38:30	NTFS \$MFT	\$SI [MACB] time	/Documents and Settings/Administrator/Recent/privatefile.doc.lnk
1567	06/04/13 20:38:30	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Recent/privatefile.doc.lnk

Figure 6. privatefile.doc opening

1593	06/04/13 20:38:30	Internet Explorer	Last Access	visited file:///C:/Documents%20and%20Settings/Administrator/Desktop/privatefile.doc
1594	06/04/13 20:38:30	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/ExtensibleCar
1595	06/04/13 20:38:30	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/ExtensibleCar
1596	06/04/13 20:38:30	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/ExtensibleCar
1597	06/04/13 20:38:30	RecentDocs key	File opened	Recently opened file of extension: .doc - value: privatefile.doc
1598	06/04/13 20:38:32	NTFS \$MFT	\$SI [MAC.] time	/Documents and Settings/Administrator/Impostazioni locali/Temp
1599	06/04/13 20:38:34	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/stdole2.tlb
1600	06/04/13 20:38:38	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/Prefetch/WORDPAD.EXE-20E16A4D.pf
1601	06/04/13 20:38:38	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/Applets/Wordpad
1602	06/04/13 20:38:38	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/Applets/Wordpad/RecentFileList
1603	06/04/13 20:38:46	NTFS \$MFT	\$SI [C..] time	/RECYCLER/S-1-5-21-854245398-1409082233-725345543-500/Dc1.doc
1604	06/04/13 20:38:46	NTFS \$MFT	\$SI [MAC.] time	/Documents and Settings/Administrator/Desktop
1605	06/04/13 20:38:46	NTFS \$MFT	\$SI [MAC.] time	/RECYCLER/S-1-5-21-854245398-1409082233-725345543-500
1606	06/04/13 20:38:46	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/Explorer/CLSID/{645FF040-5081-101B-9F
1607	06/04/13 20:38:46	\$Recycle_bin	File deleted	DELETED C:/Documents and Settings/Administrator/Desktop/privatefile.doc

Figure 7. privatefile.doc opened and deleted

1	date	time	timezone	sourcetype	type	short
2	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A.C.] time	/Documents and Settings/Administrator/Desktop/very important.doc
3	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Documents and Settings/All Users/Dati applicazioni/desktop.ini
4	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Documents and Settings/All Users/Documents/Video/Desktop.ini
5	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Documents and Settings/All Users/Documents/Musica/Desktop.ini
6	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Documents and Settings/All Users/Documents/Immagini/Desktop.ini
7	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [M.C.] time	/Documents and Settings/Administrator/Recent
8	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/desk
9	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A.C.] time	/Programmi/Windows NT/Accessori/wordpad.exe
10	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Programmi/Windows NT/Accessori/msword8.wpc
11	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [MACB] time	/Documents and Settings/Administrator/Recent/very important.doc.lnk
12	06/10/13	17:19:05	UTC	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Recent/very important.doc.lnk
13	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [MACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
14	06/10/13	17:19:05	UTC	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
15	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/Programmi/Windows NT/Accessori/msword6.wpc
16	06/10/13	17:19:05	UTC	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
17	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [ACB] time	/Documents and Settings/Administrator/Impostazioni locali/Cronologia/Hist
18	06/10/13	17:19:05	UTC	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/msftedit.dll
19	06/10/13	17:19:05	UTC	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/Ext
20	06/10/13	17:19:05	UTC	UserAssist key	Time of Launch	UEME_RUNPATH:C:/Programmi/Windows NT/Accessori/WORDPAD.EXE
21	06/10/13	17:19:05	UTC	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/Ext
22	06/10/13	17:19:05	UTC	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/Ext
23	06/10/13	17:19:05	UTC	RecentDocs key	File opened	Recently opened file of extension: .doc - value: very important.doc
24	06/10/13	17:19:05	UTC	UserAssist key	Time of Launch	UEME_RUNPATH
25	06/10/13	17:19:05	UTC	XP Prefetch	Last run	WORDPAD.EXE-20E16A4D.pf: WORDPAD.EXE was executed

Figure 8. Opening of Desktop/very important.doc and file link creation

ated on 1st June 2013, at 13:07 CET. If after 60 minutes we try to show the file property again, we will see that: Figure 3 and Figure 4. The field “Ultimo accesso”, translated into “Last Access”, was updated to 3 June 2013 at 17:42 CET.

The timeline follows: you can see the updating of \$SI access timestamp of *time.csv* (Figure 5).

So, at last, think about that: during an investigation, you find a powered on PC running Windows XP. You see a highly interesting file on the user desktop, and the law enforcement are looking for just that file. In that case, it is not uncommon to take a first look at the file, maybe just to know about the creation date and time, or know just the last access time.

So, you just right click on that file and see the property, getting the right values. But if the file was created at least 60 minutes before your right click action, then the second time you right click or analyze the file with your preferred tool, you will get the wrong date, that is, the time you right clicked the file. No more original last access time that you firstly saw.

03

OPENING DOCUMENT FILE

Let's take a look on what happened when opening a document file on Figure 6.

419	06/10/13 17:38:25	MACB	SetupAPI Log	Entry written	DriverContextual information. Contextual information. Contextual information. Information msg 022. Information
420	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/Driver Cache/i386
421	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/config/systemprofile/Dati applicazioni/Microsoft/SystemCertificates/My/CTLs
422	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/config/systemprofile/Dati applicazioni/Microsoft/SystemCertificates/My/CRLs
423	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/config/systemprofile/Dati applicazioni/Microsoft/SystemCertificates/My/Certificates
424	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/ServicePackFiles/ServicePackCache/i386
425	06/10/13 17:38:28	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/drivers/USBSTOR.SYS
426	06/10/13 17:38:29	MACB	SetupAPI Log	Entry written	DriverContextual information. Contextual information. Contextual information. Information msg 022. Information
427	06/10/13 17:38:30	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/syssetup.dll
428	06/10/13 17:38:30	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/storprop.dll
429	06/10/13 17:38:31	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/drivers/fasfat.sys
430	06/10/13 17:38:37	MAC.	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/system32/config/default.LOG
431	06/10/13 17:38:40	MAC.	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/setupapi.log
432	06/10/13 17:38:40	MACB	SetupAPI Log	Entry written	DriverContextual information. Contextual information. Contextual information. Information msg 022. Information
433	06/10/13 17:38:41	MAC.	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/Prefetch/NTOSBOOT-B00DFAAD.pf
434	06/10/13 17:38:42	MACB	NTFS \$MFT	\$SI [MACB] time	/WINDOWS/Prefetch/RUNDLL32.EXE-3BA74C30.pf
435	06/10/13 17:38:42	MACB	NTFS \$MFT	\$FN [MACB] time	/WINDOWS/Prefetch/RUNDLL32.EXE-3BA74C30.pf
436	06/10/13 17:38:42	MAC.	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/Prefetch
437	06/10/13 17:38:46	MACB	UserAssist key	Time of Launch	UEFI UISCUT
438	06/10/13 17:38:46	MACB	UserAssist key	Time of Launch	UEFI RUNPATH:[My Computer] VIRTUAL
439	06/10/13 17:38:49	A..	NTFS \$MFT	\$SI [A..] time	/WINDOWS/system32/shell32.dll
440	06/10/13 17:38:49	..C.	NTFS \$MFT	\$SI [..C.] time	/WINDOWS/explorer.exe
441	06/10/13 17:38:49	MAC.	NTFS \$MFT	\$SI [MAC.] time	/WINDOWS/Prefetch/RUNDLL32.EXE-451FC2C0.pf
442	06/10/13 17:38:49	MACB	MountPoints2 key	Drive last mounted	{8f312cea-d1f4-11e2-bf41-000c29a2309b} volume mounted

Figure 9. USB plugging events

480	06/10/13 17:38:54	..C.	NTFS \$MFT	\$SI [..C.] time	/Programmi/Windows NT/Accessori/wordpad.exe
481	06/10/13 17:38:54	..C.	NTFS \$MFT	\$SI [..C.] time	/System Volume Information/ restore(3EDD201A-674D-4829-AFD2-565941
482	06/10/13 17:38:54	MAC.	NTFS \$MFT	\$SI [MAC.] time	/Documents and Settings/Administrator/Recent/very important.doc.lnk
483	06/10/13 17:38:54	MAC.	NTFS \$MFT	\$FN [MAC.] time	/Documents and Settings/Administrator/Recent/very important.doc.lnk
484	06/10/13 17:38:54	MACB	NTFS \$MFT	\$SI [MACB] time	/Documents and Settings/Administrator/Recent/testdir.lnk
485	06/10/13 17:38:54	MACB	NTFS \$MFT	\$FN [MACB] time	/Documents and Settings/Administrator/Recent/testdir.lnk
486	06/10/13 17:38:54	MACB	UserAssist key	Time of Launch	UEFI RUNPATH
487	06/10/13 17:38:54	MACB	RecentDocs key	Folder opened	Recently opened file of extension: Folder - value: testdir
488	06/10/13 17:38:54	MACB	RecentDocs key	File opened	Recently opened file of extension: .doc - value: very important.doc
489	06/10/13 17:38:54	MACB	UserAssist key	Time of Launch	UEFI RUNPATH:C:/Programmi/Windows NT/Accessori/WORDPAD.EXE

Figure 10. Opening very important.doc from a USB device

The really interesting part of that experiment is on rows 1553, 1557, 1565 and 1603: when you open a document file, in our case *privatefile.doc* (that was opened by *WordPad* program), a file was created under *RECYCLERS* folder.

It's important to say that you will not find the \$SI and \$FN birth timestamp set because the file has the creation time set up as the original creation time, so you will find only the MFT change and file access timestamps set, as you can read on rows 1553 and 1557.

On rows 1566 and 1567 you can see the creation of a link file under the Recent folder, showed under the Start menu (Figure 6).

The file is named *Dc1.doc*: D stands for Deleted, c is the logical drive which the file belongs to, and 1 is a sequential number.

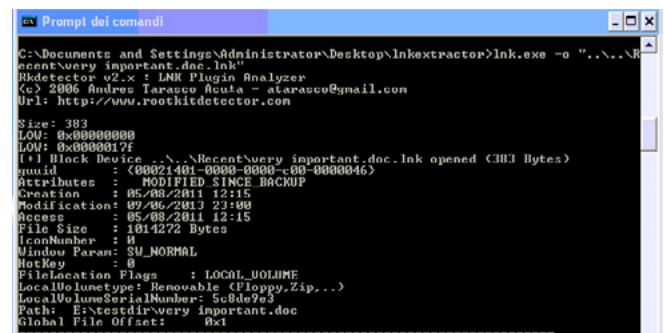


Figure 11. Inkeextractor in action

Figure 7 shows the start of *WORDPAD.EXE*, recorded by the Windows prefetch feature (row 1600), and then, on row 1603, the *Dc1.doc* meta-data changed to reflect the action done on row 1607: the file was deleted, that is, moved into recycler bin (in this case, the DEL keystroke was typed) (Figure 7). The key concept to keep in mind is: when you open a document file, with WordPad or Microsoft Office Word, a file is created under the *RECYCLERS* folder, so you can keep tracks of its changes and, if deleted, you can recover some basics information about it.

04

READING RECENTS FILE FOLDER

This is an interesting behavior of Windows XP, the scenario is: you view a file, say, on your Desktop folder. Windows creates a link file on the Recent folder, as saw on the previous paragraph. Later, you open a file with the same name but on a different location or folder: Windows XP does not create a new link file on the *Recent* folder, instead it updates the old one, so you lose the previous infor-

```

C:\Documents and Settings\Administrator\Recent>dir /TC
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 000F-071D

Directory di C:\Documents and Settings\Administrator\Recent

22/05/2012  16.26           532 22 maggio.txt.lnk
23/05/2012  12.06           893 Case Information.lnk
16/03/2012  09.26           404 fhchats.lnk
16/03/2012  09.27           507 fbstatus.lnk
22/05/2012  15.06           704 IEF - mag 22 2012 09.37.14.lnk
23/05/2012  12.06           570 IEF - mag 22 2012 14.07.42.lnk
23/05/2012  12.16           575 IEF - mag 22 2012 16.56.33.lnk
16/07/2012  08.46           570 IEF - mag 22 2012 17.07.11.lnk
23/07/2012  21.08           642 IEF - mag 22 2012 10.00.32.lnk
22/05/2012  16.44           699 IEF - mag 22 2012 21.20.00.lnk
22/05/2012  10.54           1.016 IEFCase.lnk
03/05/2013  15.01           657 Immagini campione.lnk
03/05/2013  12.41           360 imputazioni.doc.lnk
27/05/2012  00.39           807 index.lnk
31/05/2013  15.19           187 Install Ubuntu Gnome (E).lnk
03/05/2013  15.01           911 Inverno.jpg.lnk
31/05/2013  15.19           283 Mandiant.pdf.lnk
04/06/2013  21.38           522 privatefile.doc.lnk
27/05/2012  08.39           544 Report 2012-05-27 08-35-31.lnk
03/05/2013  16.22           246 test.lnk
10/06/2013  10.30           254 testdir.lnk
03/05/2013  15.02           918 Tramonto.jpg.lnk
10/06/2013  18.19           383 very important.doc.lnk
                23 File             13.184 byte
                0 Directory 54.799.622.144 byte disponibili

```

Figure 12. Link creation time

```

C:\Documents and Settings\Administrator\Recent>dir /TC
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 000F-071D

Directory di C:\Documents and Settings\Administrator\Recent

22/05/2012  16.26           532 22 maggio.txt.lnk
16/07/2012  08.46           893 Case Information.lnk
16/03/2012  09.26           404 fhchats.lnk
22/05/2012  10.53           507 fbstatus.lnk
22/05/2012  15.06           704 IEF - mag 22 2012 09.37.14.lnk
23/05/2012  12.06           570 IEF - mag 22 2012 14.07.42.lnk
23/05/2012  12.16           575 IEF - mag 22 2012 16.56.33.lnk
16/07/2012  08.46           570 IEF - mag 22 2012 17.07.11.lnk
23/07/2012  21.08           642 IEF - mag 22 2012 10.00.32.lnk
22/05/2012  16.44           699 IEF - mag 22 2012 21.20.00.lnk
03/05/2013  15.02           1.016 IEFCase.lnk
22/05/2012  16.27           657 Immagini campione.lnk
22/05/2012  10.56           360 imputazioni.doc.lnk
31/05/2013  15.19           807 index.lnk
03/05/2013  15.01           187 Install Ubuntu Gnome (E).lnk
31/05/2013  15.19           911 Inverno.jpg.lnk
04/06/2013  21.38           283 Mandiant.pdf.lnk
22/05/2012  10.56           522 privatefile.doc.lnk
03/05/2013  16.22           544 Report 2012-05-27 08-35-31.lnk
10/06/2013  10.30           246 test.lnk
03/05/2013  15.02           254 testdir.lnk
10/06/2013  18.38           918 Tramonto.jpg.lnk
                23 File             13.184 byte
                0 Directory 54.799.622.144 byte disponibili

```

Figure 13. Ink files modification timestamp

mation. Let's see how this happens. On Figure 8 you can see the opening of *very important.doc*, on row 2 you can see the click action on the file, resident on the user desktop, and at rows 11 and 12 you can see the creation of the link file in the *Recent* folder. The remaining highlighted rows shows the files involved in the file opening process. Now, let's plug in a USB device (Figure 9) and click another file named *very important.doc*, the same name as the one on the user desktop. Figure 10 shows this action: as you can see, on rows 482 and 483, the link file *very important.doc.lnk* is updated (MAC timestamp) to reflect our action, plus a new link created, *testdir.lnk*, newly created. To better understand what happened, let try to parse the link file with *Inkextractor*: Figure 11 shows the *very important.doc.lnk* file information and metadata. The timestamps shown refers to the object file, as is, *very important.doc*. The important fields to take a look at are *LocalVolumetype*, telling us what kind of device the link refers to, *LocalVolumeSerialNumber*, as the name says, it's the logical volume serial number of the file location, *Path* is the folder where *very important.doc* resides. To double check our test, have a look at the *Ink* timestamps. Figure 12 shows *very important.doc.lnk* file creation: it is set to 18.19 (UTC+1), the time when *Desktop\very important.doc* was opened (see Figure 8, row 11). The line "Numero di serie del volume", translated into Local Volume Serial Number, referring to the C: partition where *Desktop\very important.doc* resides, differs from the one recorded into *very important.doc.lnk*. Figure 13 shows the modified timestamp of *Ink* files.

CONCLUSIONS

In this article we have shown some Windows XP specific behavior that can must be taken into account when conducting forensics analysis of a Windows XP system.

We used often a timeline because it is an invaluable technique to know was happening and when on our system, but it is also important to cross-check the information and results gathered with timeline and other tools.

ABOUT THE AUTHOR



Davide Barbato has 10 years of IT experience, the last three in Digital Forensics and Incident Response. He is currently employed in an important DFIR national firm, SSRi di Lorenzo Laurato S.a.s., in which his works as Chief Security Officer and DFIR analyst. He is also a teacher and speaker at national meetings and universities about Digital Forensics, IT Security and IT Privacy. davide.barbato@ssrilab.com



The **only** existing System of its kind,
IncMan Suite has already been adopted
by a host of corporate clients worldwide

The Ultimate Forensic Case Management Software

Fully automated Encase Integration

Evidence tracking and Chain of Custody

Supports over 50 Forensic Software and third parties

Training and Certification available

Special discount for LEO, GOV and EDU customers



SPECIAL PROMO 15% OFF
single user perpetual license

<http://www.dimmodule.com>

promo code **E-FORNCS13**

DF Labs DIM is a forensic case management software that coherently manages cases, data input and modifications carried out by the different operators during Digital Evidence Tracking and Forensics Investigations.

It is part of the IncMan Suite, thus it is able to support the entire Computer Forensics and Incident Response workflow and compliant with the ISO 27037 Standard.

www.digitalinvestigationmanager.com

WINDOWS MEMORY FORENSICS & MEMORY ACQUISITION

by Dr Craig S. Wright, GSE, GSM, LLM, MStat

This article takes the reader through the process of imaging memory on a live Windows host. This is part one of a six part series and will introduce the reader to the topic before we go into the details of memory forensics. The first step in doing any memory forensics on a Windows host involves acquisition. If we do not have a sample of the memory image from a system we cannot analyze it. This sounds simple, but memory forensics is not like imaging an unmounted hard drive. Memory is powered and dynamic, and changes as we attempt to image it.

What you will learn:

- An introduction to Memory acquisition and imaging
- Memory analysis reasoning
- Why we image and analyse memory

What you should know:

- You should have a basic understanding of forensics and incident handling
- Understand system imaging
- Basic windows processes

This means it is not a repeatable process. Not that there is a requirement at all times for the results of a forensic process to provide the same output; in this it is not necessary to be able to repeat a process and obtain exactly the same results. It does not mean we cannot use a variable process in a forensic investigation. What it does mean is we have a set of steps that will allow us to image memory but that every time we do those the results will change.

INTRODUCTION

Although the results obtained in a forensic analysis of memory will vary with no two memory images being able to display the same hash value, this does not mean the process does not follow with a scientific rigor. If the same investigator uses the same process to obtain and acquire an image

of the system memory on the same computer twice in a row both images will vary significantly. The reason for this is that computer memory changes during the imaging process.

Parts of the physical memory are mapped to hardware devices. The majority of mapped and allocated hardware memory cannot be easily imaged and an attempt to do so will result in the image process crashing the system. So for all these differences and variations in the acquisition of a system's memory we have a process that can be followed but results that will vary each time it is used. Some forensic practitioners see this as a problem. That however is far from the truth. If we take medical forensics as an example, the practice of forensic autopsies has been followed for over 100 years. Yet in this

practice it is not possible for another surgeon or coroner to return the organs to the body and repeat the process. What they can do is follow a set process that will gain similar results if they are not the same.

In this article we will discuss what you should know about imaging computer memory. You will learn the fundamentals of memory imaging on a Windows system. Further follow-up articles to this one we will look at using specific tools and imaging processes.

WHERE DO WE START

Like any good forensic practice we need to follow repeatable processes. One of the best guidelines for doing this is the Internet engineering task force request for comment 3227 (<http://www.ietf.org/rfc/rfc3227.txt>) – RFC 3227, “Guidelines for Evidence Collection and Archiving”.

Like all standards and checklists, this document is far from perfect and needs to be modified to suit many environments. It is however a starting guide that should be considered. Anytime you deviate from a well-known checklist such as this it is important to justify and document your reasons.

The first thing to note is that memory is volatile evidence. It changes rapidly and unlike a hard drive evidence can quickly disappear. For this reason it is necessary to acquire an image of the system memory whenever possible as early as possible into the acquisition process. Each time you run a command on the system we are changing evidence. In doing this we are potentially overwriting areas of memory that may contain valuable information necessary for a case. The quicker we gain access to the memory and image it the less likely it is we will lose that evidence.

The best forensic method is always the one that achieves the results we are seeking most economically, but more importantly with the fewest changes to the system. In this article we will not be discussing the more disruptive and potentially damaging methods (including the Cold Boot Method) that can be used in systems where access is not available to image memory.

RFC3227

RFC 3227 provides us with some good guidelines on what we should image first. This is listed in order below:

- registers, cache
- routing table, arp cache, process table, kernel statistics, memory
- temporary file systems

- disk
- remote logging and monitoring data that is relevant to the system in question
- physical configuration, network topology
- archival media

In our case, the capture of non-hardware assigned memory will grab the majority of the system registers, cache routing tables etc. Though it is not possible to capture everything unaltered – it is highly unlikely that this will ever be achieved in any incident handling process.

MEMORY IMAGING AND FORENSICS

Memory imaging differs markedly from many other forms of digital forensics. As we have already noted, memory imaging differs significantly from disk imaging. When we image a hard drive we generally do not have to skip areas and the same process can be run multiple times without altering any evidence. To that extent hard drives are not terribly volatile source of evidence

The process of running a memory imager requires that we load the process into memory. This process of course results in changes to the memory we are attempting to image. This is why the result is not repeatable in a way that will produce the same hash value each time we enact it. The worst part of all this is that we cannot even determine whether the program has correctly imaged the memory in all cases. Being that we can expect different results each time we run a memory imager we cannot accurately determine if a particular section of memory was missed or incorrectly copied.

Memory imaging is not an instantaneous process. The result of this is that a program or other data in memory can change from a portion of memory that has not been read to one that the imager has already copied as the process is run. Consequently, it is possible to miss copying selected areas of memory. This does not invalidate the forensic value of a memory image. What we need to understand is not that the collected evidence is invalid, but that we only have a subset of the entire memory from the machine we are seeking to image. What we do have is an accurate copy of what is on the machine. This is where the forensic value is gained. At the same time however, we may not have a complete copy of all of the evidence, and it could be further evidence that the evidence of an event or incident is missing from our investigation.

Cyber criminals are rational [1]. When they create malicious code they consider the economic constraints and opportunities [2] that are associated

with producing and managing malicious code. As a result, malicious code authors have created ways for their programs to bypass many memory capture processes. They specifically seek to evade memory imaging. There are reasons for this – if malicious code can evade detection, it can manage to remain undiscovered and hence active for longer periods of time. In doing this, the cybercriminals can maximize the economic returns that they gain from the creation of this malicious code. I have discussed some of the methods used by malicious code authors and penetration testers (*Extending Control, API Hooking*) in penetration testing articles published in Hakin9 (<http://hakin9.org/buffer-overflow-exploiting-software-052012/>) amongst others. In some instances the attacker creates code that uses processes such as API hooking to link into system processes and kernel functions. Some of the more sophisticated Malware will recognize the name of an imaging program or the system calls that such a program makes and will intentionally alter its behavior. This could involve changing the location of the malicious code in memory as the system is imaged and it could even extend to feeding false data to the memory imager.

DEVICE MEMORY

If we open up the Windows “Device Manager” and select “Resources by connection” (see Figure 1) we can have a look at the memory devices on a

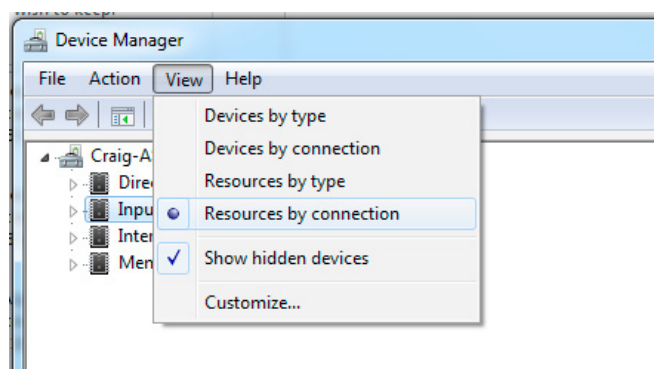


Figure 1. Viewing Windows Memory

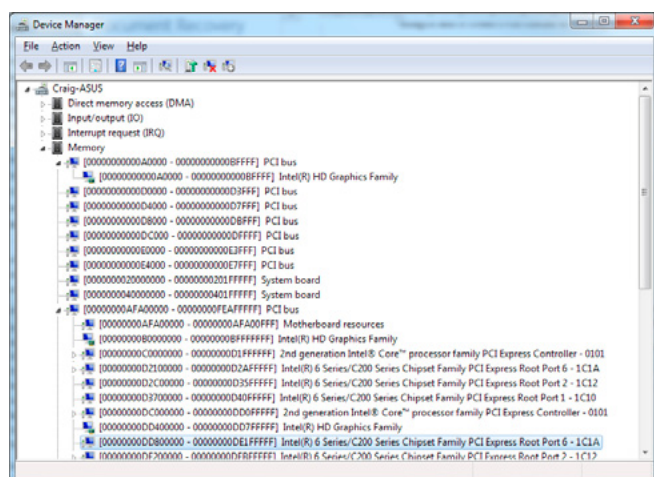


Figure 2. Windows Hardware Memory Locations

Windows system. Under the Windows kernel object, `\Device\PhysicalMemory` we have the means to obtain direct access to the memory on a Windows system.

We can see (Figure 2) that some of the physical memory is allocated to hardware devices. These areas are ones we need to avoid when imaging the memory as any request to these memory locations is written to the hardware device. This could crash the system. These are known as mapped memory locations. These points are important to note when working on Windows systems. Each tool will have different specialties which require different privileges and have different advantages across different operating systems. Before we select which tool will be deployed in a particular imaging engagement, we need to consider the particular operating system we wish to image.

A particular problem comes from practicing with a tool on one operating system and then migrating the same processes to another. What works on Windows XP for instance may not work, or may even crash the system in Windows 7. In particular, it is important to practice on the various different systems you will engage with. If you are working in an environment with multiple operating systems it is important to practice on each of them. This means gaining an understanding of the following:

- the required system privilege levels
- the various system architecture (such as 32-bit versus 64-bit)
- the differences in operating systems including patching levels
- any difference where data is written or called from.

CAPTURE TOOLS

In this article we will not address any of the commercial products. In later articles following this one we will continue with details on the use of particular tools that are available freely. It is wise to become familiar with a wide range of tools depending on the circumstances you work within. Mandiant distributes two free tools for memory capture and analysis:

- Redline (<http://www.mandiant.com/resources/download/redline>)
- Memoryze (<https://www.mandiant.com/resources/download/memoryze>)

We will look at a free tool from MoonSol in this article.

MOONSOLS DUMPIT

MoonSols provides a free Windows memory dump kit (<http://www.moonsols.com/ressources/>). As it states on its website you can do the following:

- This utility is used to generate a physical memory dump of Windows machines. It works with both x86 (32-bits) and x64 (64-bits) machines.
- The raw memory dump is generated in the current directory, only a confirmation question is prompted before starting.
- Possible to deploy the executable on USB keys for quick incident response needs.

It is simple to run DumpIt. The program runs when you extract it from the file and it can be run from an external device such as a USB. In figure 3 we see it running with the default destination for a saved image. You do require Administrative privileges on the host. Running in default mode (such as double-clicking) saves the image which is named based on the time, the system ID and with the default extension of .raw.

The default location can be changed but happens to be the location where you run the program from. You will also note that the required size of the image is noted (Address space size) and that the available space on the destination drive is listed (Free space size). It is of course essential to ensure that there is sufficient free space on the drive to be able to complete the imaging process.

Starting a memory image capture is simple from the prompt noted in Figure 3 we just select “y” in order to image the drive or “n” to end the program.

Once the image capture is completed, it will be stored in the destination directory as shown in Figure 4. At this point we have taken volatile memory and created a forensic image that we can analyze later without fearing further data loss. Always ensure that the image copy is made to an external device and not the primary hard drive (Figure 4).

There are no command line options built into DumpIt. You either need to change the location or hook data into the program to change its running state. For this reason it can be considered a one step memory imaging program.

PAGE FILE

The Windows page file is one of the simpler ways of analyzing memory. The location can vary but the file “pagefile.sys” is not too difficult to find even on an imaged hard drive. This creates a less volatile form of memory analysis. Another opportunity comes from analyzing the Windows hibernation file (Hiberfil.sys). One of the best ways of capturing memory is when a virtual machine is in use. System snapshots capture and save memory as well as hard drive-based information and evidence.

I will not list the general location of the page file as this does vary and more importantly a Windows system can have up to 16 different locations across different drives for storing page files. One of the most important reasons to capture a page file is that idle processes can be “paged” out of active memory when they are in the background. Simply imaging the systems memory could thus result in missing critical information.

Capturing a page file should be done separately to the complete imaging of the hard drive as the page file will change far more rapidly than the hard drive itself. It may be less volatile than system memory but it is still volatile evidence. To capture the page file will require access to the raw drive, as a direct copy cannot be made.

VIRTUAL IMAGES

Another source of memory information that we can obtain comes from virtual images. Programs such as VMWare, Windows virtual PC and many others allow us to take a snapshot of the system. Sometimes we can run these directly, saving the captured virtual image and running it as a machine where we can interact and experiment. In addition files such as the “.vmem” file in VMWare contain information that we can extract with a tool such as Volatility.

When we take a virtualized machine image, the suspended file is not volatile at all. This file is a serialized memory image and Malware cannot hide in this environment. This gives an advantage to serv-

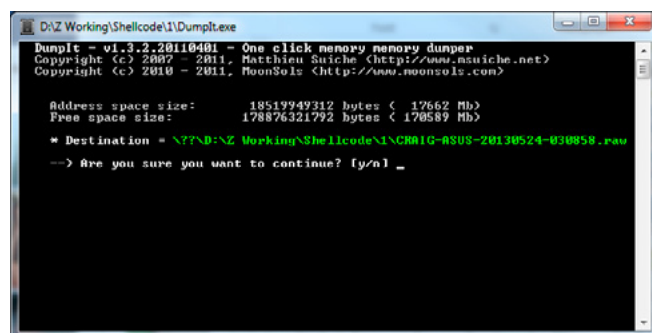


Figure 3. MoonSols DumpIt

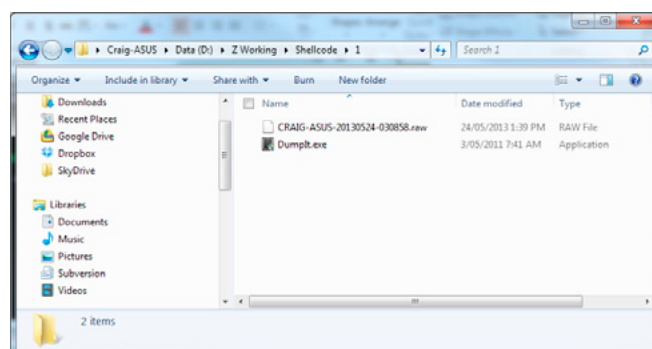
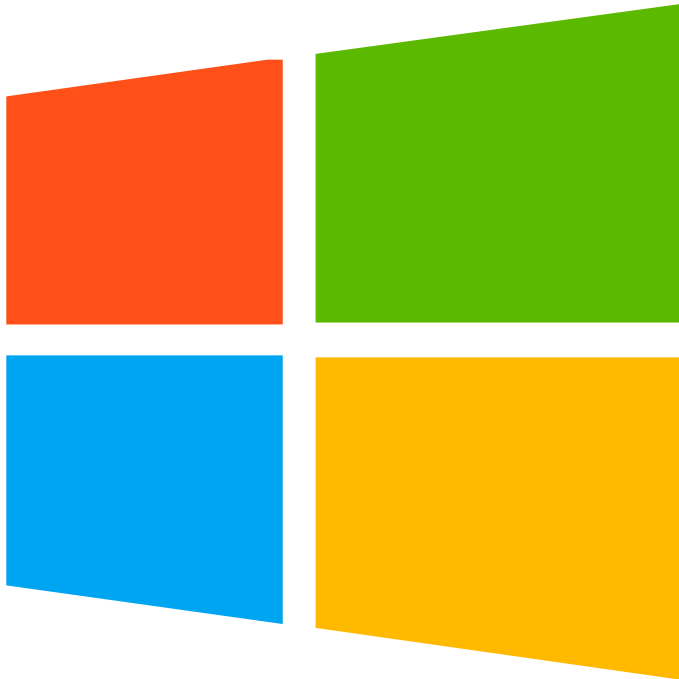


Figure 4. The memory image

REFERENCES

- [1] Wright, C. S. (2011). Criminal Specialization as a corollary of Rational Choice. Paper presented at the ICBIFE, HK, China.
- [2] Wright, C. S. (2012). Territorial behaviour and the economics of botnets. Paper presented at the SECAU Perth, WA.



TM

Practice,

Practice,

Practice,

And when you're done doing that...

Practice some more...

ers and workstations that run in a virtualized environment. These systems can be analyzed completely. In some instances they can be analyzed as the machine is still running.

TO CONCLUDE...

In the next article, we will start analysing the image we have captured.

Memory is volatile evidence and as such needs to be acquired early in the process. Perhaps more critical is the difficulty associated with acquiring a memory image. As memory imaging is going to change in results every time we enact the procedure, but memory imaging is not robust. By its very nature, memory is fragile and if you attempt to access many areas of device memory you can crash the system. The results of this would be a complete destruction of all evidence. To ensure that this does not happen to you it is important to always practice using the tools you intend to image a live system with.

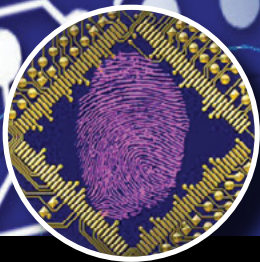
There are some ways to access system memory that are less volatile. These include hibernation files, page files and virtual machine images. When analyzing a system, always remember that you should collect as much evidence as you can in the time that is available. Also remember to document the process that you have followed and to practice this before imaging a live system.

If you walk into a forensic engagement and start by crashing the system very few people will take your evidence to be reliable. So remember...

ABOUT THE AUTHOR



Dr Craig Wright (Twitter: Dr_Craig_Wright) is a lecturer and researcher at Charles Sturt University and executive vice –president (strategy) of CSCSS (Centre for Strategic Cyberspace+ Security Science) with a focus on collaborating government bodies in securing cyber systems. With over 20 years of IT related experience, he is a sought-after public speaker both locally and internationally, training Australian and international government departments in Cyber Warfare and Cyber Defence, while also presenting his latest research findings at academic conferences. In addition to his security engagements Craig continues to author IT security related articles and books. Dr Wright holds the following industry certifications, GSE, CISP, CISA, CISM, CCE, GCFA, GLEG, GREM and GSPA. He has numerous degrees in various fields including a Master's degree in Statistics, and a Master's Degree in Law specialising in International Commercial Law. Craig has just completed working on his second doctorate, a PhD on the Quantification of Information Systems Risk and is mad enough to be planning his third doctorate.



Burgess Consulting and Forensics

Data Recovery Experts

Saving Data for Decades

**We can find what you
thought was lost forever!**

We pioneered the field of data recovery in 1985 and have successfully recovered data for thousands of clients since then.

From spilled frappuccinos to fires, floods and just plain drive crashes – count on us to **save your computer's data**, whatever disaster befalls it.

From personal laptops to smart phones and corporate databases, we pride ourselves on finding data that others can't – on all types of digital media.

With a 90% **success** rate, chances are we can save **your** data too.



Since 1985 we have extracted data from **more than 15,000** hard disks and digital devices.

We can save your valuable data from Windows, Macintosh, Linux, cameras, smart cards, smart phones and most other digital media.

In 2004, the Pine Grove School library in Orcutt, California **burned to the ground**.

We **recovered** all of the insurance and inventory **data**, enabling the school to rebuild.



Let us save your data.

*Computer Forensics
Expert Witness Services
Data Recovery*

Office: 805-349-7676
Fax: 805-349-7790
info@burgessforensics.com
1010 W. Betteravia Rd., Ste. E
Santa Maria, CA 93455 USA

HOW TO DETECT A FILE WRITTEN TO

AN USB EXTERNAL DEVICE IN WINDOWS FROM THE MRU LISTS

by Carlos Dias da Silva

Today one of the principal company asset is the digital information. The digital information can be used of a lot of methods and also can be copied using different modes. To know and to control what files were sent to out of the company is a problem nowadays and never is a little the investment to guarantee the data secure.

What you will learn:

- How to use MRU Lists for detect files written to an USB drive;
- How to use the Regripper for mount the Windows registry keys;
- How to use the Encase Imager and FTK Imager for to navigate, export and analyze structure file systems;
- How to detect a file written to an USB external device

What you should know:

- Familiarity with the Encase Imager and FTK Imager;
- Familiarity with the Windows' Registry.

A lot of files are copied and accessed in external storage devices in a company, further when the BYOD concept became a normal practice. In a forensic work, to know what files was accessed from USB devices can help very much. This information can be discovering with the exam of MRU list and some keys of Windows Registry. This article will teach you how to do this.

WHAT IS MRU LISTS

MRU are Most Recently Used lists that most applications and even the operating system uses indiscriminately and in different ways. The original purpose of the MRU was to enable users to keep track of where files landed and to re-visit them for editing. It is less and less useful when the Windows Indexer enables you to find recently modified/created/read (CMR) directory lists.

HOW THIS CAN HELP US

Windows is a nosey operating system. But it is not Microsoft's fault (entirely); it tries to support users by not letting them loose files they show an interested in. Loosing files is a big problem; there are so many places users can put them. Windows tries to restrict the locations to removable devices, and the portion of the local file system tree user the `\user\<USERNAME>\documents` tree (In Windows 7+8). An extension to this is that users generally can write to external visiting file systems such as a USB thumb drive, USB writable CD drive, or a USB portable giant 3TB hard drive. To determine what files have been copied to external devices is not an easy task. The Windows operating system has no consistent policy to record these activities. Many of these records are made by applications in their own registry space in the Windows

Registry. Window Explorer tries to keep a MRU (Most Recently Used) documents list in Explorer's application scratch directories in various places, how for example at directories \users\<USERNAME>\AppData\Roaming\Microsoft\Windows\Recent\ - \users\<USERNAME>\AppData\Roaming\Microsoft\Office\Recent. All of this information is persistent and discoverable in the windows registry and in files Explorer keeps.

In Microsoft Windows systems we demonstrate technique that relies on the exam of the MRU lists along with the exam of the Windows Registry detect which files were written to in removable devices. Similarly we can discover if files from the network or server computer were copied to USB devices.

This tutorial will require the download of tree free tools, Encase Imager, FTK Imager and Regripper, which can be found in the following websites:

- www.guidancesoftware.com/Order-Forensic-Imager.aspx
- www.marketing.accessdata.com/acton/form/4390/0119:d-0002/0/index.htm
- www.regripper.wordpress.com

This tutorial starts from the examining stage, and previous knowledge is required on evidence collection and keeping. The following use case was developed to improve the understanding of the tutorial's final purpose.

USE CASE

The company's monitoring software showed that 3 confidential documents were downloaded from the

company's server to a computer of the internal network, as follows: MyDocument 01.docx, MyDocument 02.docx and MyDocument 03.docx.

The company's forensics team assigned to check the situation made a bit stream copy of the suspected computer hard disk and conducted technical examinations to identify what was done with the confidential documents downloaded from the company's server.

PART 01 OF EXAMINATIONS

Open the forensic image of the disk using the tool Encase Imager to have access to the files structure (Figure 1).

After opening the forensic image in Encase, the first step is to find the moved files. In order to do that, click on the listbox to show all the files of the disk, as seen in Figure 2 and Figure 3.

All files of the directory structure will be listed in the right-side table of Encase. Double click the column "Name" to arrange the files list by name, as seen in Figure 4.

Add Evidence

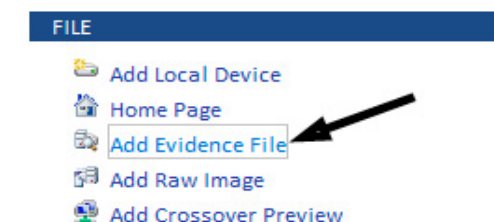


Figure 1. Opening an evidence with Encase Imager

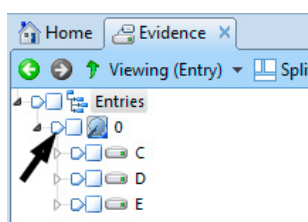


Figure 2. Evidence on Encase Imager

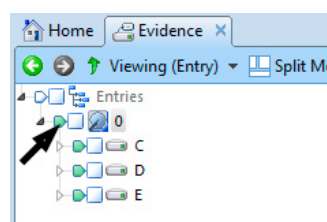









Figure 3. Listing all files on Encase Imager

Table		Selected 0/187799
	Name	
<input type="checkbox"/>	1	Check Your PC Performance.url
<input type="checkbox"/>	2	##1859A58683C270F3
<input type="checkbox"/>	3	#173.236.242.101
<input type="checkbox"/>	4	#acfe.com
<input type="checkbox"/>	5	#admin.brightcove.com
<input type="checkbox"/>	6	#apps.sistema.radio.br
<input type="checkbox"/>	7	#cdn.zopim.com


Figure 4. Classification files by name on Encase Imager


Name	Last Accessed	File Created
MyDocument 03.docx.Ink	15/08/13 11:29:17	15/08/13 11:29:17
MyDocument 01.docx.Ink	15/08/13 11:29:08	15/08/13 11:29:08
MyDocument 02.docx.Ink	15/08/13 11:29:13	15/08/13 11:29:13


Figure 5. Searching files on Encase Imager


Table			
    Selected 0/187799			
		Name	Last Accessed
<input type="checkbox"/>	1	 MyDocument 03.docx.Ink	15/08/13 11:29:17
<input type="checkbox"/>	2	 MyDocument 01.docx.Ink	15/08/13 11:29:08
<input type="checkbox"/>	3	 MyDocument 02.docx.Ink	15/08/13 11:29:13


Fields

 Report

 Zoom In

 Zoom Out

 100%

 Previous Item


 Next Item

Figure 6. Files' report on Encase Imager

After the classification, click on the panel and type the name of one of the files that need to be inspected, in this case, we will start with “MyDocument 01”. Scroll down the files until you find them, as seen in Figure 5. It can be seen that files with the extension “Ink” were found. These “Ink” files are automatically created by the operating system when a certain file is opened. They serve as a shortcut to the original file, and are stored in the directory “Systemroot\user\AppData\Roaming\Microsoft\Windows\Recent\” in Microsoft Windows 7 systems.

An “Ink” file has the path to the original file, to which it provides the shortcut, i.e., with the file path it is possible to identify in which device it was stored. To obtain a report on the “Ink” file, click on any of them to select it and then on the tab “Report” of Encase Imager, as seen in Figure 6.

A report on the “Ink” file will be shown, with the following metadata and relevant information:

Table 1. Details on metadata

Metadata	Description
Last Accessed	Date of the last access to “Ink”
File Created	Date on which the “Ink” file was created, this date indicates the date on which the original file was first open
Last Written	Last change in the “Ink” file
Volume Name	This name is given by the operating system or manually to the storage unit established in the system.
Base Name	Directory path of the original file

Name	MyDocument 01.docx.Ink
File Ext	Ink
Logical Size	568
Category	None
Last Accessed	15/08/13 11:29:08
File Created	15/08/13 11:29:08
Last Written	15/08/13 11:29:08
Link Data	
IDList Size	319
Link Flags	Has Link Target ID List Has Link Info Has Known Folder Tracking
File Attributes	Archive
Volume Name	PORTABLE
Serial Number	7E93-2134
Drive Type	2
Base Name	G:\UTIL\Documents\MyDocument 01.docx
Working directory	G:\UTIL\Documents
Property Storage Size	40
Property Storage Data	Id(0)

Figure 7. File’s report on Encase Imager

The relevant information and metadata on the file “MyDocument 01.docx.Ink” are the following: Figure 7.

Accordingly, the shortcut (Ink) to the file “MyDocument 01.docx” allowed us to obtain the following information:

- The file “MyDocument 01.docx” was last accessed on 08/15/2013 at 11:29:08;
- The volume that stores this file is named “PORTABLE”;
- The file “MyDocument 01.docx” is stored on the directory structure “G:\UTIL\Documents\”.

This first part of examinations has indicated us that it is necessary to be certain about what the “G:” unit named “PORTABLE” refers to, given that it is in this unit that one of the confidential files is stored. In order to do that, we will sort the records of the operating system by exporting them through the tool FTK Imager. We will use the FTK Imager

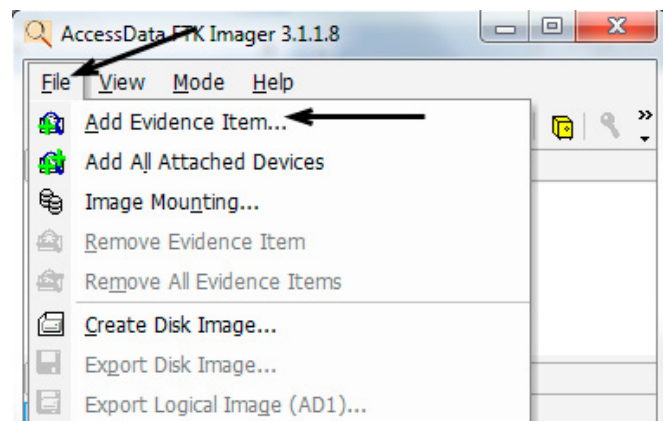


Figure 8. Opening an evidence with FTK Imager

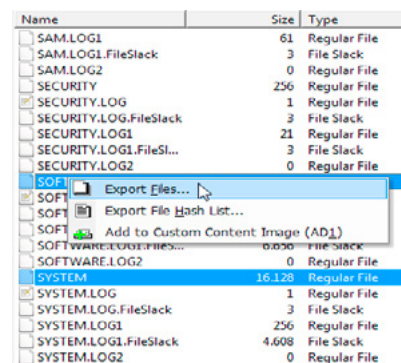


Figure 9. Export files on FTK Imager

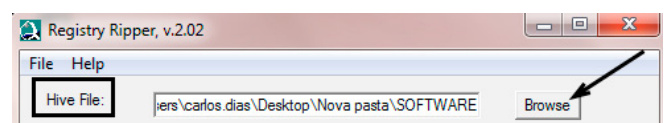


Figure 10. Mounting file on Regripper

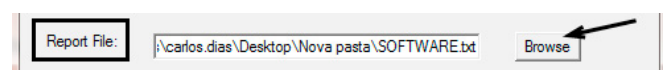


Figure 11. Mounting file on Regripper

in case the Encase Imager does not allow exporting the files.

PART 02 OF EXAMINATIONS

Open the forensic image in the FTK Imager: Figure 8. After assembling the image in the FTK, access "C:\Windows\System32\Config" and export the files "System" and "Software", as follows: Figure 9.

Open the tool RegRipper to assemble the exported files, according to the following steps.



Figure 12. Mounting file on Regripper



Figure 13. Mounting file on Regripper

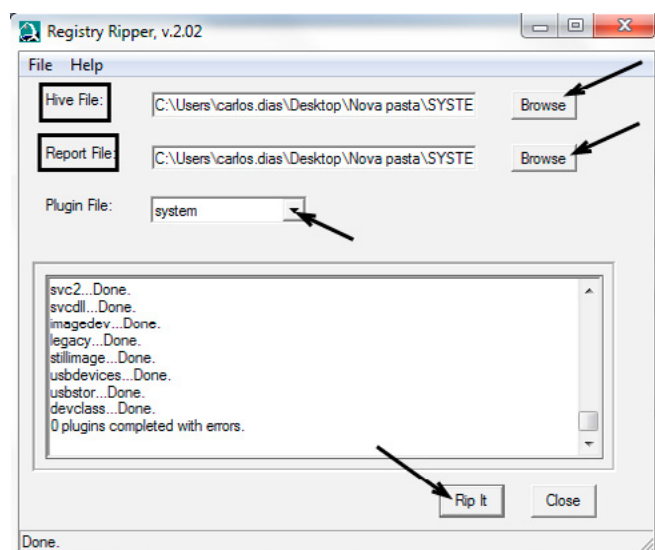


Figure 14. Mounting file on Regripper

```
Device : DISK\VEN SANDISK&PROD_CRUZER_BLADE_REV_1.26
LastWrite : Thu Aug 15 11:17:56 2013 (UTC)
SN : 4C532006840524104580&0
Drive : PORTABLE
```

Figure 15. Registry key

```
Device: ?? USBSTOR\Disk\Ven_SanDisk\Prod_Cruzer_Blade\Rev_1.26\4C532006840524104580&0
\\?\Volume{b1bc55b0-ac80-11e2-80a0-20689d5ecfab}\DosDevices\G:
\DosDevices\G:
```

Figure 16. Registry key

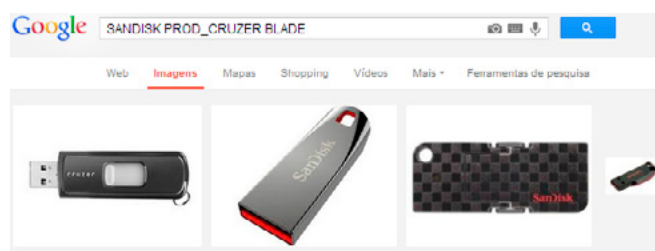


Figure 17. Google search for SanDisk pendrive

ON THE WEB

- Details about Ink files <http://www.forensicswiki.org/wiki/LNK>
- www.guidancesoftware.com/Order-Forensic-Imager.aspx
- www.marketing.accessdata.com/acton/form/4390/0119:d-0002/0/index.htm
- www.regripper.wordpress.com

Click in the "Browser" button of the field "Hive File" and point the exported file "Software" (Figure 10). Then click on the "Browser" button of the field "Report File" and point where you want the record report to be created (Figure 11). Select the file "Software" in the field "Plugin File" (Figure 12). Click in the "Rip it" button to create the report on the Software record (Figure 13). Follow the same procedure for the file System (Figure 14). Now that the record files are assembled, we will examine them in order to find the required answers.

In the file "SOFTWARE" there is the registry key "Microsoft\Windows Portable Devices\Devices". Locate this key and examine the portable devices that were connected to the computer. It can be seen that the device "SANDISK&PROD_CRUZER_BLADE" named "PORTABLE" was connected to the computer in the same day and time of the accesses of the confidential files "Ink" (that were accessed at 11:29) (Figure 15). This device has a serial number and it is possible to identify from that number which unit letter the operating system has attributed to this device. In order to do that, open the assembled file "System" and search for the serial number of the device. The result will be similar to the following: Figure 16. When searching for the device SanDisk Cruzer Blade, we found that it is a flash drive, as follows: Figure 17. Examining the other Ink files referring to the remaining confidential files, we found that both point to the same "G:" unit, which proves that they are stored in a flash drive.

CONCLUSION

The conclusion of this analysis is that files with the same name of the files downloaded from the company's server were found inside a flash drive of SanDisk. It was not possible to compare the files because there are only indications to the external device. The analysis of the flash drive is required to prove the deviation of information.

ABOUT THE AUTHOR



Carlos Dias is a systems analyst specialized in Digital Forensics. He has 10 years of experience in technology plus 5 years of experience in Digital Forensics. Nowadays he is coordinator at BDO Brazil conducting Digital Forensics projects in litigation, intellectual property, frauds and cyber crimes. Certifications: ACE, ISFS

THE WINDOWS FORENSIC ENVIRONMENT

by Brett Shavers

The Windows Forensic Environment, also known as Windows FE or WinFE, is a Windows operating system that can be booted from external media such as a CD, DVD, or USB flash drive. Windows FE is based on Windows PE, which is a minimal Windows operating system with limited services, used to prepare a computer for Windows installation, among other tasks related to Windows. The main, and of course most important, difference between Windows FE and Windows PE, is that Windows FE forensically boots a computer system whereas Windows PE does not. What makes WinFE different from non-Windows based forensic boot systems is that with WinFE, the forensic examiner can use almost all of their favorite Windows based software tools, rather than Linux applications.

What you will learn:

- The differences between Windows PE and Windows FE.
- How to build Windows FE.
- How to use Windows FE in common and uncommon scenarios.

What you should know:

- Fundamentals of digital forensics regarding preservation of original electronic evidence.
- Operation of the forensic tools you wish to use in the WinFE bootable system.
- How to boot a computer system to external media and not the evidence hard drive through BIOS configuration changes.

How would you like to boot an evidence machine to Windows to forensically image, triage, or even examine it using your favorite Windows based forensic applications? I mean, literally, boot the evidence machine to Windows. Not to Linux. Not to DOS. Not having to remove the hard drives and connect to a hardware write blocking device. Simply boot to Windows and go to work. That is the power of the Windows Forensic Environment.

Troy Larson, of Microsoft brought his idea of a Windows forensic operating system to me in 2008. Troy asked me to build a WinFE from instructions he provided, and let him know what I think. I confess, at the time I was extremely busy and did not immediately put the effort to try. In fact, after reading the instructions, I assumed that it would take too much time spend on a Windows PE that only modified two registry changes. According to Microsoft's website.

"Windows Preinstallation Environment (Windows PE) 2.0 is a minimal Win32 operating system with limited services, built on the Windows Vista kernel.

It is used to prepare a computer for Windows installation, to copy disk images from a network file server, and to initiate Windows Setup.

Windows PE is not designed to be the primary operating system on a computer, but is instead used as a standalone preinstallation environment and as an integral component of other setup and recovery technologies, such as Setup for Windows Vista, Windows Deployment Services (Windows DS), the Systems Management Server (SMS) Operating System (OS) Deployment Feature Pack, and the Windows Recovery Environment (Windows RE)” (What is Windows PE?, 2013).

That definition alone did not encourage me sufficiently to consider WinFE as the next best thing in forensic tools. Boy was I wrong. I did not fully understand the genius behind this simple modification to a Windows PE, that would provide forensic examiners worldwide, a great tool until after I spent some time to test it myself. The impetus to get me going was a presentation Troy gave in Seattle about WinFE (Larson, 2009). This one presentation made me run directly to the office and start building my first WinFE. Even in 2009, Troy had developed a process to use WinFE, and access both Shadow Copies and Bit-locked drives with a forensic boot disc. How neat is that!

Before getting into how to build WinFE, let’s talk a little about where it stands now. The current result of WinFE is seen below in Figure 1. As you can see, it looks like Windows for one reason...it is Windows. In Figure 1 below, it is Windows 7. WinFE can just as easily be built using Windows XP, Vista, or Windows 8; in both 32bit and 64bit. You can also see that since it is Windows, you can run your Windows based forensic tools such as Accessdata’s FTK Imager (<http://www.accessdata.com/support/product-downloads>), Guidance Software’s Encase Forensic (<http://www.guidancesoftware.com>), X-Ways Forensics (<http://www.x-ways.net>), or any number of your favorite tools.

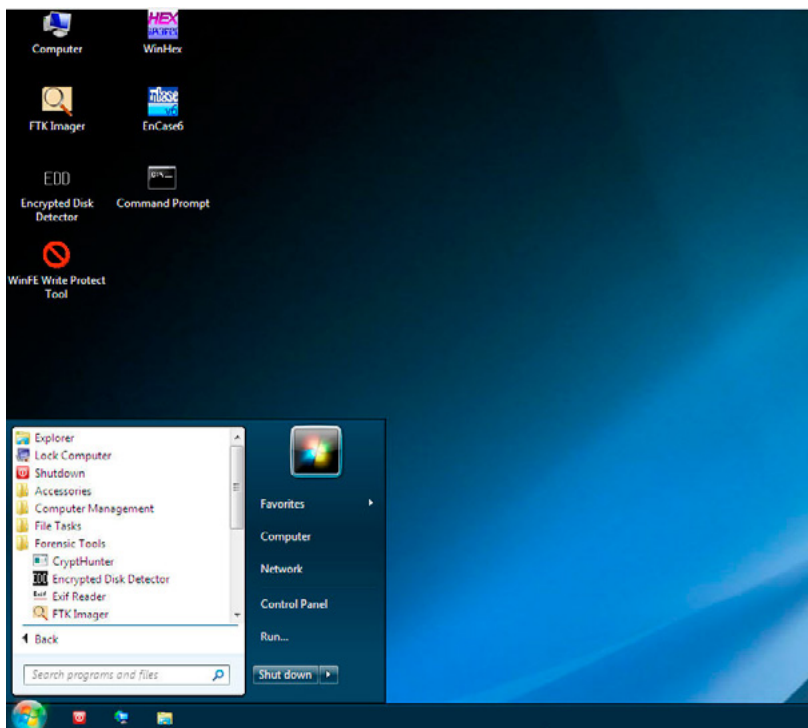


Figure 1. Windows FE

Besides adding to efficiency of analysis, being able to use the same tools you use daily in forensic examinations – ones that you can use on a forensic boot disc – is just plain neat. I also use Linux forensic boot systems, but my first choice is most always WinFE. Like many forensic examiners, I am personally more comfortable using Windows based tools in the acquisition of evidence; for the primary reason that I most always use Windows based tools for analysis. Also, the majority of commercial forensic applications are Windows based; thus, making the most common operating system, naturally Windows.

METHODS OF BUILDING YOUR OWN

Just the thought of 'building your own operating system', is enough to frighten the most hardened forensic analyst. However, it is so much easier than it sounds. Actually, since the creation of WinFE in 2008, there have been a few different methods of building WinFE, resulting in different end results of appearances and applications able to run on each type of build. The manner in which you choose to build a WinFE is solely dependent upon your needs.

Your needs can vary between building a minimal WinFE, that can boot older systems, to a more full-featured WinFE to conduct triage or a complete forensic analysis. As most examiners carry different Linux forensic versions of boot discs, examiners can have different build versions of WinFE to approach systems of unknown hardware. Older systems typically may have less RAM, requiring a WinFE that does not require more RAM than the evidence computer provides. In most systems today, WinFE will have more RAM than necessary to operate without problems.

BASIC WINFE BUILD

The basic build is the original build method developed by Troy Larson. The two registry modifications to Windows PE to create Windows FE are:

```
HKLM\WindowsFE8 %1\WindowsFE8\mount\Windows\System32\config\SYSTEM
HKLM\WindowsFE8\ControlSet001\Services\MountMgr /v NoAutoMount /t REG_DWORD /d 1 /f
HKLM\WindowsFE8\ControlSet001\Services\partmgr\Parameters /v SanPolicy /t REG_DWORD /d 4 /f
HKLM\WindowsFE8\ControlSet001\Control\FileSystem /v DisableDeleteNotification /t REG_DWORD /d 1 /f
HKLM\WindowsFE8
```

Essentially, these modifications to the registry prevent any drives to be automatically mounted when Windows PE (now "FE") boots. Specially, all internal disks are booted offline and any attached storage, such as external USB drives are placed online at boot. The user can mount, and unmounts drives as needed through command lines in DISKPART.

Figure 2 shows the interface to a Basic WinFE build. As you can see, there is not a *start button* or *menu*. There is only a command shell; however, many of your Windows based forensic applications will be able to run from the command line.

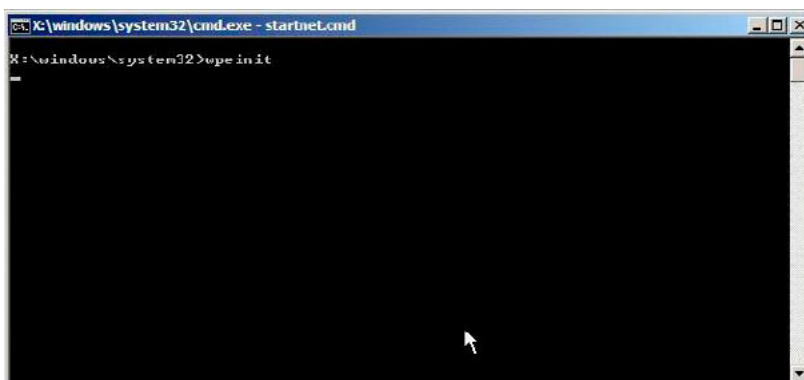


Figure 2. WinFE Basic Build

By drilling down to the directory of any installed forensic apps, simply run the executable of your application to operate the software. Figure 3 shows FTK Imager running in its GUI in the Basic WinFE, even though the application must be started through the command shell.

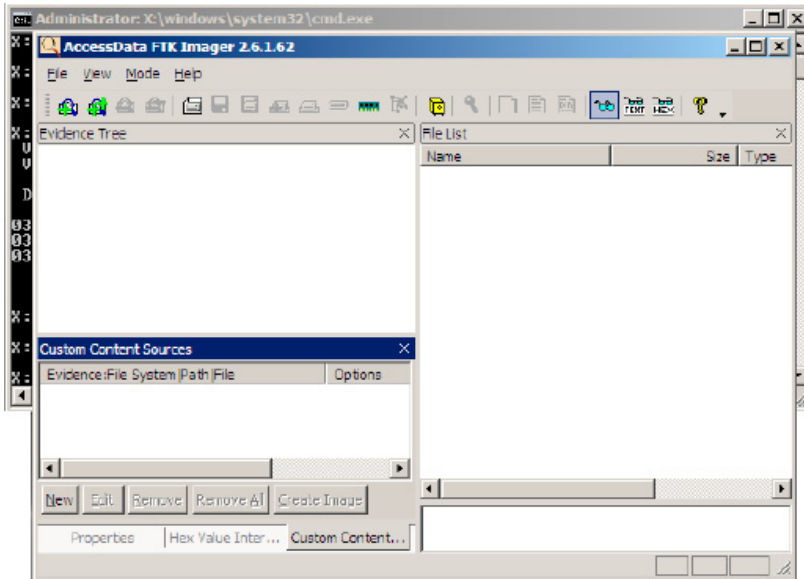


Figure 3. FTK Imager in the Basic WinFE

Building a Basic WinFE requires the Windows Automated Installation Kit (AIK), freely available from Microsoft's website. With the AIK installed, the build is made using DISKPART by opening a command shell with administrative privileges and running DISKPART. Through a series of typed commands, or running a pre-made batch file, a WinFE ISO file is created which is then placed onto your boot media (burned to a disc or installed to a USB device).

The commands to create a Basic WinFE are:

```
call copy.cmd x86 %1:\WinFEx86
Dism /Mount-Wim /WimFile:%1:\WinFEx86\winpe.wim /index:1 /MountDir:%1:\WinFEx86\mount
REG LOAD HKLM\WinFEx86 %1:\WinFEx86\mount\Windows\System32\config\SYSTEM
REG ADD HKLM\WinFEx86\ControlSet001\Services\MountMgr /v NoAutoMount /t
REG_DWORD /d 1 /f
REG ADD HKLM\WinFEx86\ControlSet001\Services\partmgr\Parameters /v SanPolicy /t REG_DWORD /d 3 /f
REG UNLOAD HKLM\WinFEx86
Dism /image:%1:\WinFEx86\mount /Add-Package /PackagePath:"C:\Program
Files\Windows AIK\Tools\PETools\x86\WinPE_FPs\winpe-scripting.cab"
Dism /image:%1:\WinFEx86\mount /Add-Package /PackagePath:"C:\Program
Files\Windows AIK\Tools\PETools\x86\WinPE_FPs\winpe-wmi.cab"
del %1:\WinFEx86\ISO\boot\bootfix.bin
xcopy "%2:\WindowsFETools\Desktop\WinPe.bmp"
"%1:\WinFEx86\mount\windows\system32\" /H /Y
md %1:\WinFEx86\mount\Apps
xcopy %2:\WindowsFETools\Applications %1:\WinFEx86\mount\Apps /S /H /Y
Dism /image:%1:\WinFEx86\mount /Add-Driver
/Driver:%2:\WindowsFETools\Drivers /recurse
"%2:\WindowsFETools\WindowsFE_FilesList.log"
Dism /Unmount-Wim /MountDir:%1:\WinFEx86\mount /Commit
move %1:\WinFEx86\winpe.wim %1:\WinFEx86\iso\sources\boot.wim
oscdimg -n -m -o -b%1:\WinFEx86\etfsboot.com %1:\WinFEx86\ISO
"%1:\WinFEx86\WindowsFE.iso"
```

As you can see, there is a lot going on that needs to be typed as commands. A better, and error free method, is using a batch file for this process for speed, ease of modification, and reduction of user errors. Rather than having to make a batch file from scratch, several are freely available on the WinFE blog at <http://winfe.wordpress.com>. Also, the batch files from the WinFE blog detail the specifics of each command used, in order for you to know what is going on under the hood during the build process.

You may also be able to guess that since the Basic WinFE is operated via a command shell and that it is a minimal build, not every program you want to run in WinFE, will be able to do so. With this build, you are limited in the applications unless you make extra effort to install or copy dependencies of each program you want to use. At the most basic need for imaging, most imaging applications will run without issue. Therefore, this Basic WinFE is able to boot the vast majority of systems, and be used for forensic imaging. A video of this build can be seen on YouTube (Creating a Windows FE ISO, 2010).

But if you want WinFE to do more with an easier build method, consider the following methods.

WINFE LITE BUILD

WinFE Lite, developed by Colin Ramsden can be considered a step in the direction of an easier build, with a much easier method of toggling drives on and offline. Remember, with the Basic WinFE build, the user mounts and unmounts drives using DISKPART with the command line. Within the DISKPART commands, there are similar commands, such as *clear* and *clean*, which can have a disastrous effect if the wrong command is given. It is possible to “clean” your evidence when you meant to “clear” the mounting status. Cleaning a drive is just what you think it is – not what you want to do with evidence...

To make handling hard drives easier, and less prone to operate error, Colin developed a write protection application for WinFE that is a simple to use, yet highly effective graphical application. Figure 4 shows Colin’s write protection application. Upon booting WinFE Lite, the write protection application automatically starts and presents a warning dialog box. After acknowledging the warning dialog, the write protection application dialog is presented.

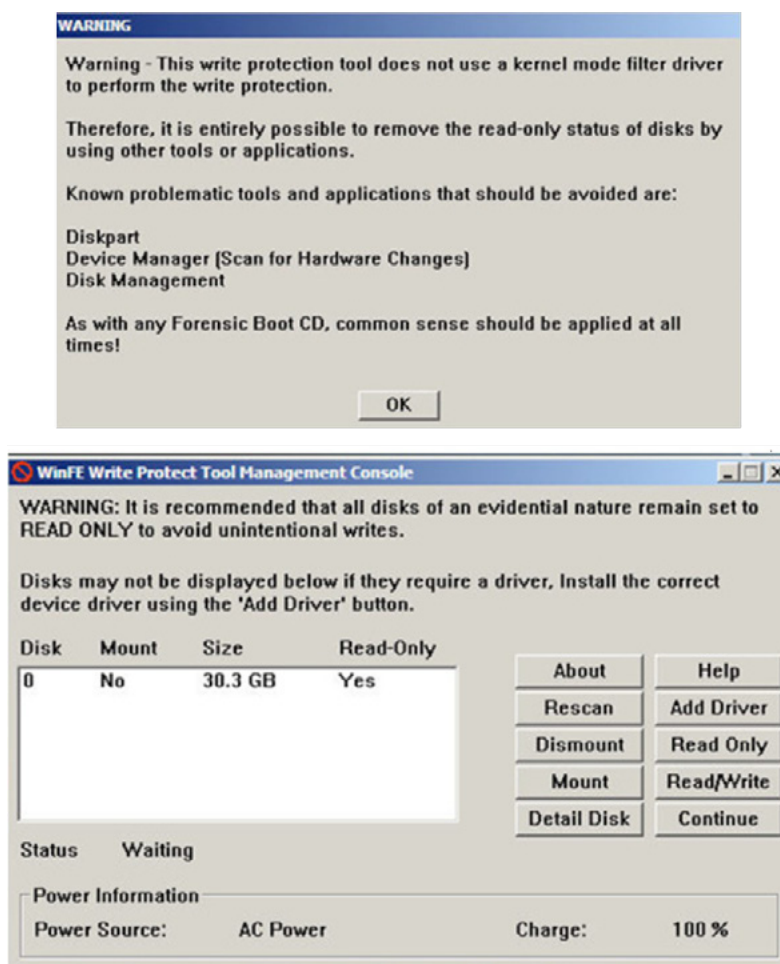


Figure 4. Write Protection Application and Warning Banner

The operation of the write protect application is as easy as it appears. Disk mounting and un-mounting is accomplished by selecting the drive, and then selecting the push button needed. Drives can be placed online, offline, in a *read only* or *read/write* mode. Using any forensic boot media requires concentration

on the status of all attached drives, internal and external, no matter if the forensic boot OS is Linux or Windows FE. The difference between *read only* and *read/write*, is the difference between a forensic bit for bit capture of evidence, or a capture of evidence that is not.

Drivers are also easily added 'on the fly' through the *Add Driver* button. This is especially helpful after booting a machine to WinFE, and later realizing you need a driver added to address a hardware issue. Without having to shutdown, create a new WinFE with the driver, and then reboot the machine, you can add the driver live for that session only.

Figure 5 shows WinFE Lite. WinFE Lite, is a visual step up from the Basic WinFE, includes several features that are easily accessible through a menu bar. The most impressive difference is the write protection application since this feature is what makes *WinPE* a *WinFE*. The write protection application also addressed disk mounting issues that exist in the Basic WinFE, for at times, external drives may have difficulty in being set online or offline.

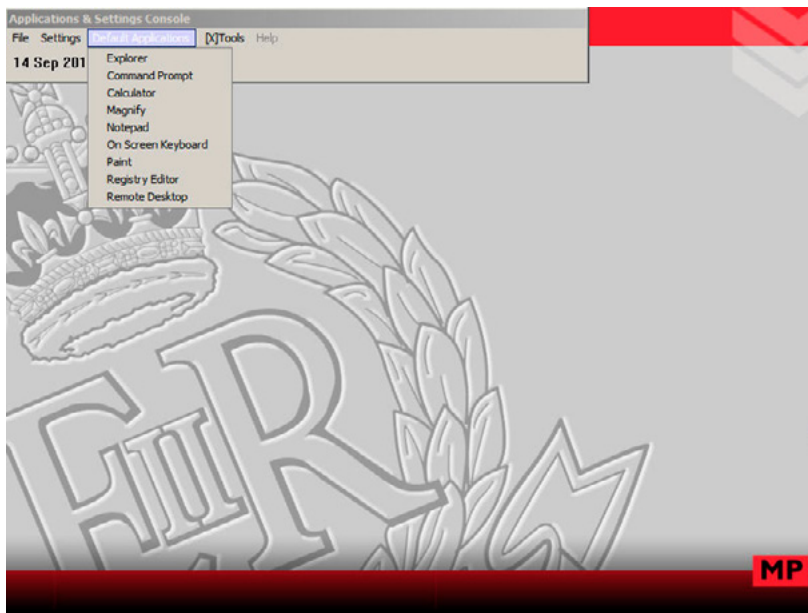


Figure 5. WinFE Lite

Detailed instructions and everything you need for the WinFE Lite build can be found on Colin's website at <http://www.ramsdens.org.uk/>.

FULL-FEATURED BUILD

Taking WinFE a step even further, is a full-featured, yet not a full Windows installation build. There are several applications developed to create a customized Windows PE, which can be easily used to create a Windows FE. One particular application used by many to create a WinFE is Winbuilder (<http://www.reboot.pro>). WinBuilder has been developed to build customized WinPEs; however, with the help of several people, scripts and plugins were written for WinBuilder to build a WinFE rather than a WinPE. The ease to which this is accomplished cannot be described enough; however, the end result, as seen in Figure 6, is almost a work of art, as any forensic examiner can fully customize their WinFE to fit their needs, and changed as needed.

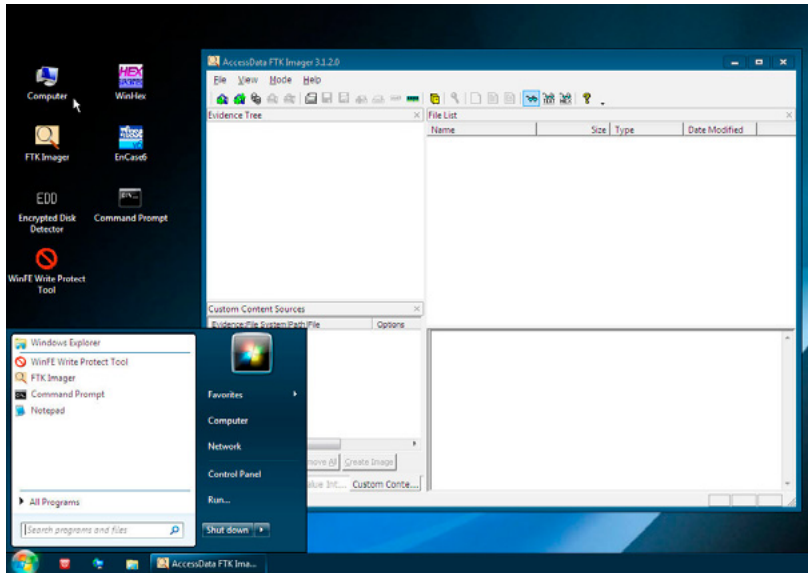


Figure 6. Full-Featured WinFE Build

Not to get ahead of ourselves, but as you can see in Figure 6, WinFE has all the makings of a Windows operating system for ease of use. To create a WinFE using WinBuilder, simply download and extract the WinBuilder zip from <http://www.reboot.pro>. You will need a Windows installation disc, which WinBuilder will use to build your WinFE. The operation of WinBuilder (Figure 7) is very intuitive, easy to use, and allows for a multitude of features. HOWEVER, the fewer features added, the better the build will be. You most likely will not need audio support, or every driver available, or network support, or even most of the available features. The more you add, the heavier your WinFE will be, thereby requiring more RAM when used.

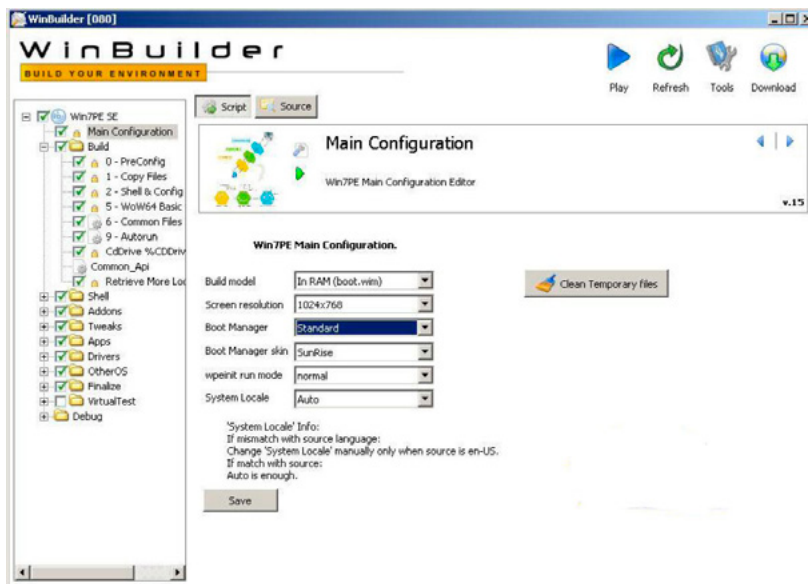


Figure 7. WinBuilder

This doesn't mean you should never use any of these features, only that when needed, add them to your build. Some features will be talked about later to increase the usefulness of WinFE. As Figure 7 shows a multitude of checkboxes, features, modifications, and customizations, it may seem overwhelming. This isn't the case since once you go through the basic needs, you can use the same settings the next time you need to build a WinFE. This method is seen in a video in YouTube (WinFE Tutorial, 2013).

If you are looking for an easier and quicker method, the latest version of WinBuilder, seen in Figure 8, may suit your needs soon. It might look like we are going backwards (or back to the command line...), which we are; but with only two or three commands, this version is faster and easier to use. All files needed

to build your WinFE are automatically downloaded for your WinFE ISO in a matter of minutes. Compared to the prior version of WinBuilder, I have seen a drastic decrease in the amount of time needed to build a WinFE, in as little as five minutes. As this build has not been completely tested for WinFE yet, I am hoping that by the time this article is in print, this newest version of the WinBuilder WinFE build will be out of beta.

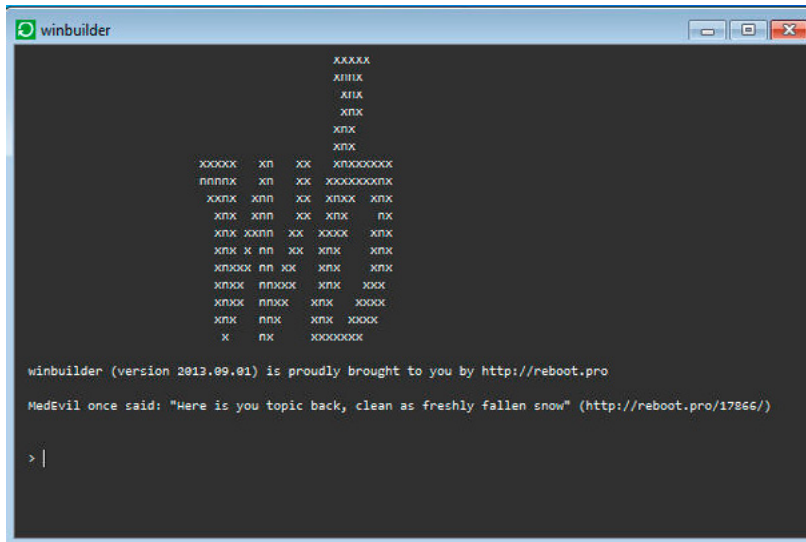


Figure 8. Latest WinBuilder

Details on these build methods can be found on the <http://winfe.wordpress.com> blog, as well as other online sources noted in the references to this article.

USING WINFE

For any person who uses Windows as their daily use operating system, the instruction in using WinFE is simple. It's Windows, just like what you are using on your forensic workstation, except for a few minor differences...

- No storage drives are mounted by default (you can choose to mount specific drives)
- The operating system is not a 'full build' of Windows, and limited in certain respects
- Not every Windows based software will run successfully due to missing dependencies in WinFE
- WinFE is booted from an external media device, such as a CD, DVD, or USB drive

Using WinFE can be as simple as booting an evidence machine to image the hard drives. Or it can involve triaging evidence machines onsite to determine which machines, if any, may contain evidence. It can be used to triage, or preview, seized machines to prioritize which one needs analyzing first. If needed, WinFE can be used onsite, on the evidence machine, to conduct as much of a forensic analysis as the examiner needs; time permitting of course. This may be extremely useful in missing person investigations, where investigative time is at a premium.

First responders can learn to use WinFE faster, rather than being trained in a different operating system. Most forensic examiners can use any type of operating system, but this cannot be expected of first responders whose job does not involving computing. By creating a customized WinFE boot system with triage applications, first responders, such as patrol officers or parole officers, are able to boot an evidence machine and conduct a triage. Of course, for any system that is to be seized subject to a search warrant or civil court order, onsite triage is not necessary. But for consent searches, or triages of multiple devices, onsite triage with WinFE using Windows based tools makes it a much easier task for first responders.

A WARNING!

I have talked about the important of knowing the status of all connected drives. WinFE is not different from a Linux forensic boot system, because with any forensic boot system, the user can intentionally or unintentionally alter the hard disks. And just like a Linux forensics boot system, drives must be placed online in a read/write mode (to write an image to), or volumes placed online in a read only mode, to allow access by certain software. An example of using read only mode would be to allow a software application that cannot see the physical drive, be able to see the logical drive. Several triage tools are only capable of viewing the logical drive, while others can view both the logical and physical.

Here is the warning when handling disks.

Setting a disk to **read only** does **NOT** alter the disk.

Setting the volume to **read only DOES** alter the disk (but only with a very small byte change)

When it is need to place a volume in *read only* mode, I would suggest only doing so when forensic analysis has not been decided, or there is no cause to seize the system. This type of triage is done to at least partially examine (triage) potential evidence devices when they would have otherwise been ignored. When using WinFE, or any forensic boot system that you plan to use, test it first; then test it again! Make sure you know what you are doing. Digital forensic started with boot discs (floppies...), and we are no less 'forensic' today with boot systems than in the beginning. We are however, much more proficient and efficient.

REFERENCES

- <http://winfe.wordpress.com>
- <http://www.ramsdens.org.uk/index.html>
- <http://www.slideshare.net/ctin/ctin-windows-fe-1256287>
- <http://praetorianprefect.com/archives/2010/04/winpe-3-0-forensics/>

WORKS CITED

- Creating a Windows FE ISO. (2010, June 30). Retrieved September 12, 2013, from YouTube: Creating a Windows FE ISO
- Product Downloads. (2013). Retrieved September 12, 2013, from Accessdata: <http://www.accessdata.com/support/product-downloads>
- What is Windows PE? (2013). Retrieved September 13, 2013, from What is Windows PE?: <http://technet.microsoft.com/en-us/library/cc766093%28v=WS.10%29.aspx>
- WinFE Tutorial. (2013, March 18). Retrieved September 12, 2013, from YouTube: <https://www.youtube.com/watch?v=Dy27R34MDkE>
- Larson, T. (2009). CTIN Windows FE. Retrieved September 13, 2009, from CTIN Windows FE: <http://www.slideshare.net/ctin/ctin-windows-fe-1256287>

ABOUT THE AUTHOR

Brett Shavers is a digital forensics expert and author. As both a former law enforcement officer and detective, Brett has investigated most types of crimes. As a private consultant, he has been retained by law firms for digital forensics analysis and has taught digital forensics at the University of Washington. He is also the author of two digital forensics books; Placing the Suspect Behind the Keyboard and The X-Ways Practitioner's Guide.

THE ONE!



The Most Powerful Forensic Imager in the World



Provides the broadest drive interface support

Built-in support for SAS, SATA, USB 3.0 and Firewire. Supports IDE and other interfaces with adapters included with Falcon

Processes evidence faster than any other forensic imager

Image from 4 source drives up to 5 destinations

Perform up to 5 imaging tasks concurrently

Image to/from a network location

Imaging speeds of up to 20GB/min

NEW FEATURES AVAILABLE NOV 2013

- NTFS Support
- TrueCrypt Support
- Drive Spanning
- The fastest E01 imaging speed available



Visit our website today to see why Falcon is The One!
www.logicube.com

INTRODUCTION TO WINDOWS FORENSICS USING PARABEN P2 COMMANDER

by Dauda Sule, CISA

Microsoft Windows is the most widely used operating system both for business and personal use. Such popularity has made it one of the most targeted operating systems by malicious attackers. As a result, it is often used as a platform to access personal and work place data, or even to commit policy breaches assisting in the commission of criminal acts. Investigations that are based on electronic evidence stand a very high chance of being carried out on a system with one or the other version of Windows operating system. It is therefore one of the most important operating systems anyone going into the field of cyber forensics will need to know how to investigate.

What you will learn:

- Basic introduction to Windows operating system
- Use of Paraben P2 Commander disk analysis
- Use of Paraben P2 Commander for image analysis

What you should know:

- Basic operation of computer systems and programs
- Basic understanding of digital forensics
- Basic understanding of Windows operating system

According to Casey (2004), “understanding file systems helps appreciate how information is arranged, giving insight into where it can be hidden on a Windows system and how it can be recovered and analyzed.” There are different versions of Windows operating systems in use ranging from the earlier versions like XP to the current Windows 8. To acquire data or analyze a system, the way and manner the specific operating system version on it operates needs to be known as each version has its peculiarities, however, this article gives a generic overview and does not go into the variances of the specific operating systems. We present an example using Windows 7.

It used to be advisable to pull out the plug on a running system that needed to be forensically analyzed – rather than shut down – so as to avoid tainting or losing; any evidence available therein, especially data in memory which

is highly volatile, making it forensically unsound; but with advancements in memory forensics, there is beginning to be a paradigm shift. Memory dumps can be taken by first responders without significantly damaging the evidence using memory forensic tools (like Mandiant Memoryze, Belkasoft Live RAM Capturer and Volatility). Such memory forensic tools are also quite good for detecting malware.

Windows systems computers mainly use one of two file systems: FAT and NTFS. The FAT (File Allocation Table) file system is the simplest of the two. Data are stored in FAT file systems are stored in sectors that are 512 bytes in size and a combination of sectors form a cluster. A cluster is the minimum unit of disk space that can be used to store a file; the smallest cluster comprises one sector. More than one file cannot be allocated to a cluster, but a file may not use up all the sectors in a cluster, there may be some space left. For example, a file of 1000 bytes will be store across two sectors (1024 bytes), leaving free 24 bytes. These 24 bytes are known as the slack space, which is more or less wasted. When a file is deleted on a system and the recycle bin is emptied, the file is not really lost, rather the system records that the cluster, which had been previously allocated for file storage, is now free (unallocated) for storage of a new file. This makes it possible to recover such a file completely if a new file is not saved to the cluster. In the event a new file is saved on the system, it will overwrite the deleted file. If it is of a larger size or equal to the previous space it will completely overwrite the previous one, making recovery more complicated if not impossible. However, if the new file is smaller than the former, there is a chance for partial recovery. For example, if a file of 1000 bytes was deleted, and a file of 700 bytes overwrote it, 300 bytes of the former will be easily recoverable using forensic tools. This partial recovery might be very significant for investigators, such those investigating child pornography who can be able to get a partial view of illegitimate photos that can serve as evidence to indict a suspect. FAT file system can show the last date and time of modification, and the last accessed data and its creation date and time, but does not show last accessed time, only the last accessed date is displayed (Casey, 2004). The NTFS (New Technology File System) supports larger disks than the FAT system and has less slack space by using compression. Information is stored in Master File Table (MFT) where every file in a directory has at least an entry (Oppenheimer, n.d.). NTFS as time stamps that can be used to track creation, modification and access of a file. In addition, NTFS has a change journal, which records objects added, modified and deleted, in streams, one for each volume on the system (Microsoft, 2003).

System logs are also valuable sources of information for an investigator. They can be used to determine when an act was committed and possibly by who (based on details like login information or IP address). It can also be possible to determine if someone else used another's system to commit the act; for example, different credentials were used to logon to the system to commit the act, or corroboration with CCTV footage shows the system owner was away at the time the act was committed, implying his credentials may have be compromised. A case illustrated by Casey (2004) refers to a disgruntled employee who configured his IP address to that of the CEO and then sent offensive messages, giving the impression that it was the CEO who sent that. Upon examining network data, it was discovered that the CEO's IP address was briefly set to a different MAC address, which happened to be that of the disgruntled staff. Internet activity is another area that leaves digital footprints. The Internet Explorer's or other browsers' history, cache and cookies are very good sources of information pertaining to Internet activity, additionally Internet Explorer maintains a rich database of Internet activity in the *index.dat* file. In the event Internet history, temporary files, cache and cookies are deleted or the browser was used in anonymous mode, there are tools that can recover such from the system (such Magnet Forensics' Internet Evidence Finder). There is also some information that can be retrieved in terms of pictures from the thumbnails view; this can be used to establish evidence against a suspect in a child pornography case.

INVESTIGATING A SYSTEM USING PARABEN P2COMMANDER DEMO ABOUT PARABEN P2 COMMANDER

P2 Commander is a forensic tool from Paraben Corporation that is built to process large volumes of data in a fast and efficient manner (Paraben Corporation, 2013). It is a commercially available tool, however, a demo version can be downloaded free for thirty days from the company's website. According to the website the tool can be used for a wide range of forensics analysis of systems like disk, chat, email, registry, Internet files and pornographic detection. The tool is quite robust and can be used for a wide range of investigations and analysis as stated, but its browser capability is restricted to Microsoft Internet Explorer, Google Chrome and Mozilla Firefox, other browsers like Opera and Safari for Windows are not included. The illustrations that follow are based on Paraben P2 Commander Version 3 on a Windows 7 system.

CREATING A CASE

After installing the Paraben P2 Commander software run it, the GUI as displayed in Figure 1 comes up. Click on *Create new case* in the welcome tab to the top left of the tab, which brings up the new case wizard (Figure 2).

Click next to enter case properties – that is the name of the case and description, stated as “Illustration” and “Example for illustrative purposes” in our example. The next stage involves entering additional information (Figure 4) where details like name of investigator, agency/company, phone and fax numbers, address email and comments. In the example, the name of investigator is entered as “Dauda Sule”; company, “Audit Associates”; comments, “Example for eForensics Magazine”; other entries are left blank. Click finish, this brings up a prompt as to where to save the case; it is saved in a directory called Paraben in this example as shown in Figure 5 (by default it saves to the Paraben directory in the Program files folder where the program was installed. Once saved, the program prompts for the category of evidence to be selected (Figure 7): logical drive, physical drive, image file, e-mail database, chat database, registry, Internet Browser data or other.

For this example, we select logical drive, and then drive H under source type; once okay is clicked, the program prompts to enter new evidence name (Figure 8), the default name (H: in the example) is there, but may be changed if required; the default name is not changed in this example. After that is entered, NTFS settings prompt (the system used in the example is a Windows 7 system and runs on NTFS file system) as shown in Figure 9 comes up giving options of criteria to be used for the evidence (search for deleted files, add the trash folder to the NTFS root, recover folder structure for bad images, and add the unallocated space folder to the NTFS root – all criteria are selected in this example).

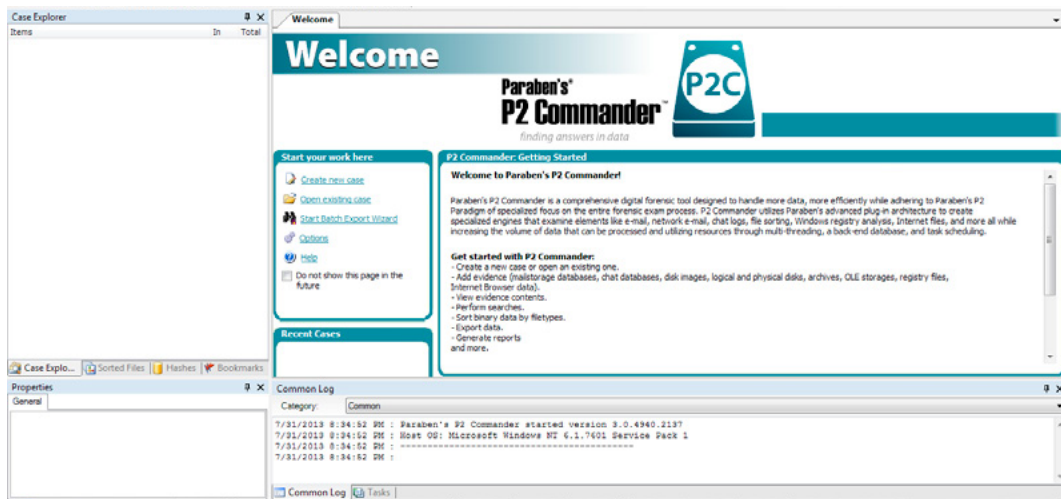


Figure 1. P2 Commander welcome interface

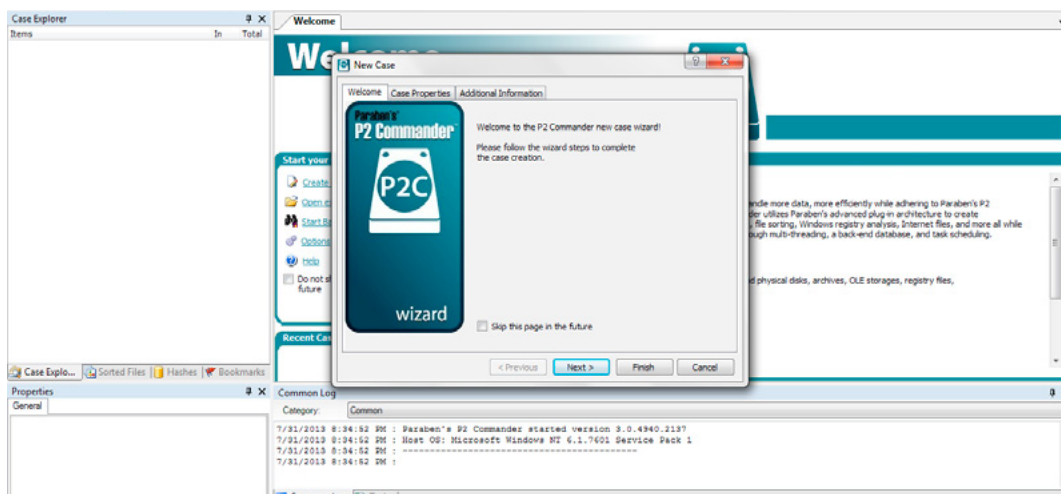


Figure 2. Welcome page of the new case wizard

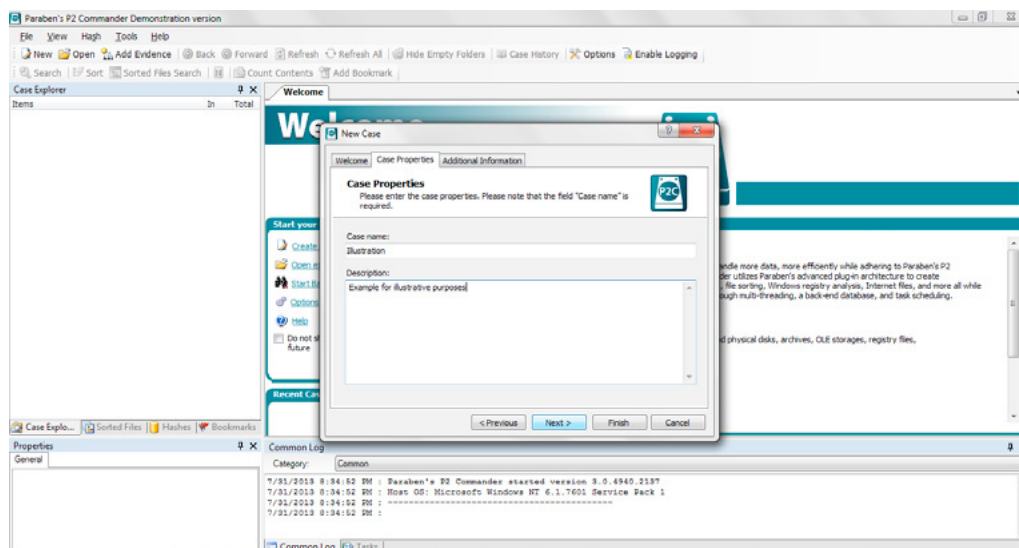


Figure 3. Case properties entry interface in the new case wizard

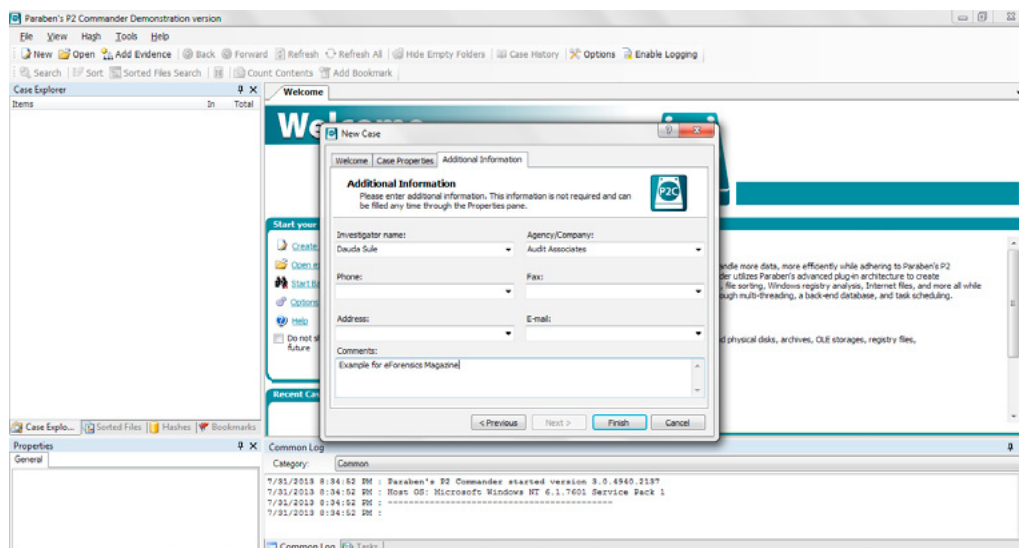


Figure 4. Additional information entry in the new case wizard

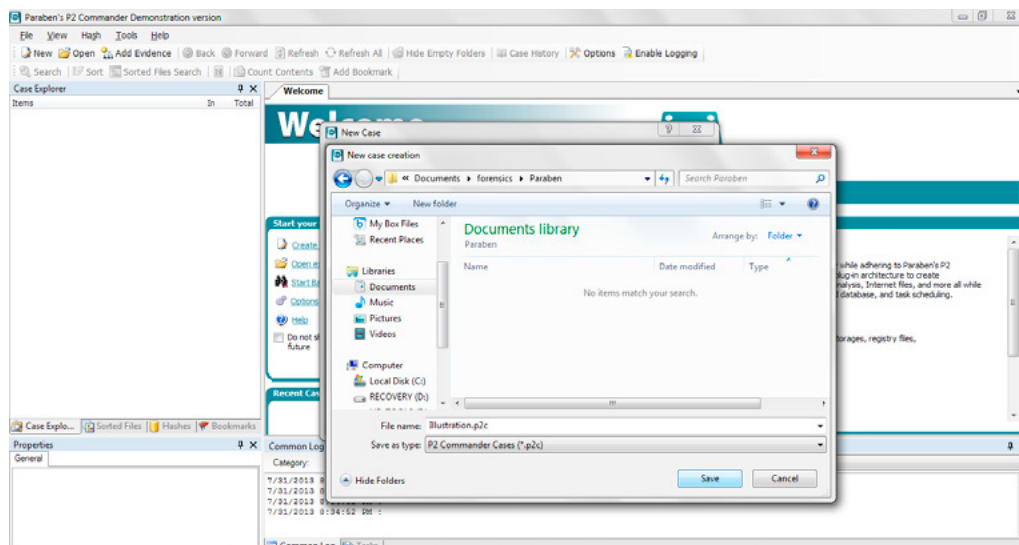


Figure 5. Selecting a directory to save the new case to



Figure 6. New case in process of being opened

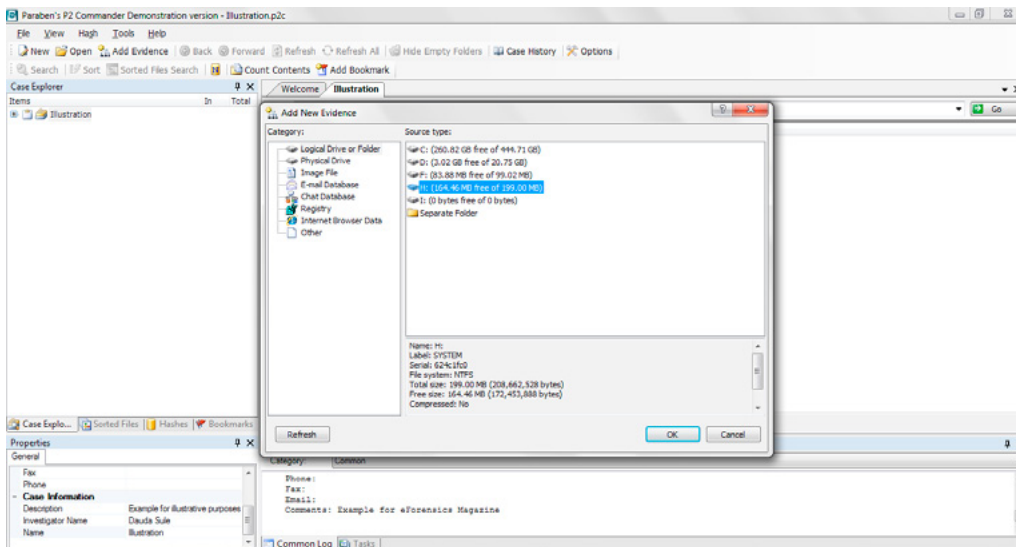


Figure 7. Adding evidence to the case

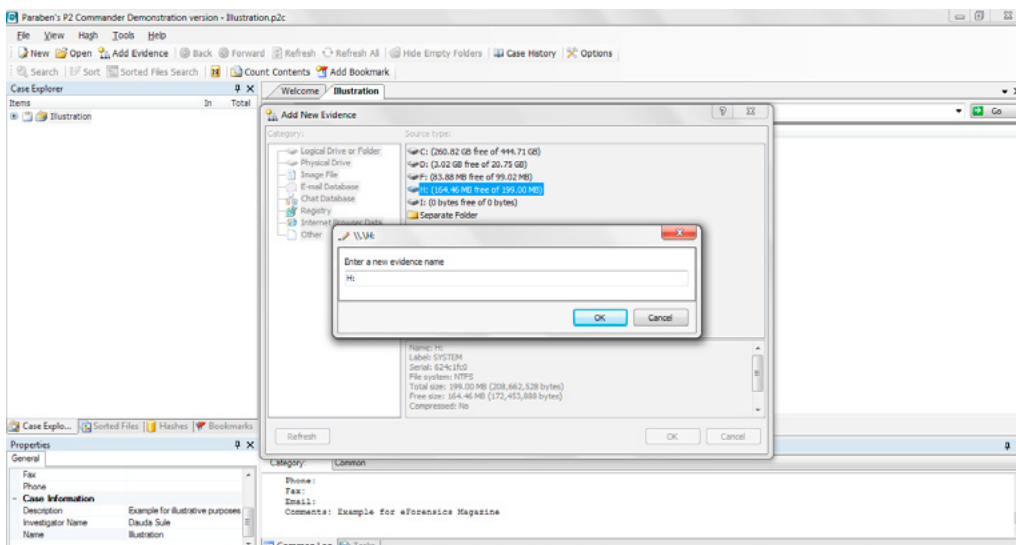


Figure 8. Selecting name for evidence

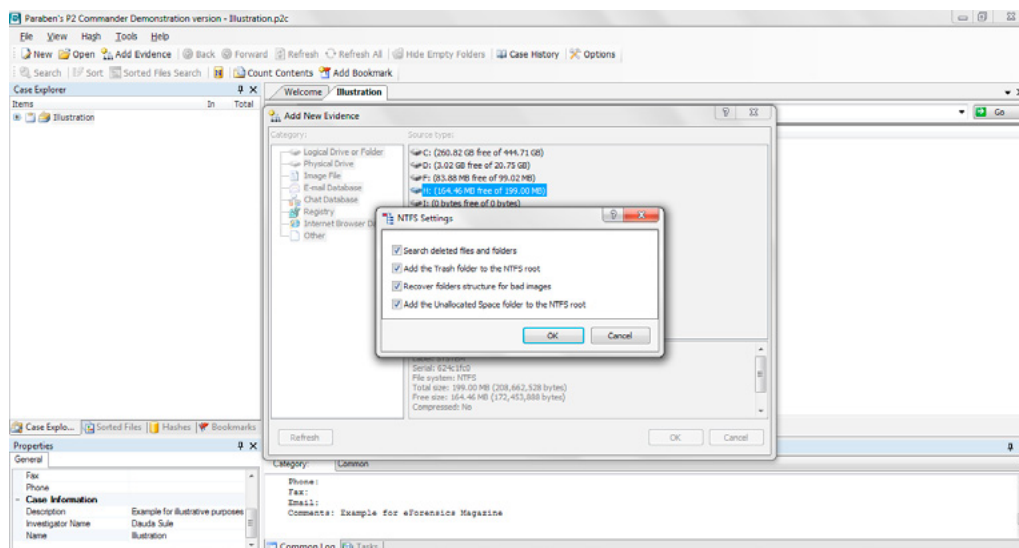


Figure 9. Settings for the evidence

EXAMINING A DRIVE

Having selected the drive to be examined, the investigation can now begin. First notice the content of the selected drive. Figure 10 shows the contents of the drive H: a word document, *document.doc*, and an image, *IMAG0912.jpg*.

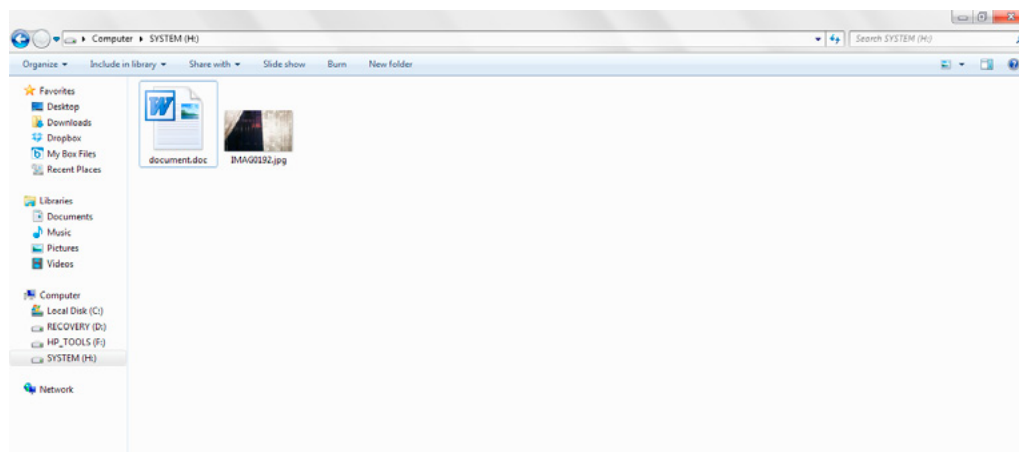


Figure 10. Contents of drive H

We can now examine what the drive contains using P2 Commander. We expand the case (*Illustration*) located in the top left corner of the interface; we expand the *NTFS* directory and click on the *Root* directory.

Among the contents of the *Root* directory are the contents of drive H, but to our amazement *document.doc* is seen to be a JPEG image data just like *IMAG0192.jpg*, and the thumbnails at the bottom of the interface further buttress that fact (Figure 11). Criminals often times try to hide their tracks by camouflaging documents to avoid detection. This could be done using different techniques from the simplest (like labeling an incriminating document or file with an innocent sounding name) to advanced techniques like steganography. What happened in this example is that the file extension for an image was changed from *.jpg* to *.doc* to avoid detection of the image by an investigator. In cases like child pornography, a suspect may try to hide incriminating pictures using such a technique in the hope of warding off investigators. Once there is an attempt by anyone who is not aware of what was done to open such document, the document would probably give an error message or open in codes that would be considered unintelligible by most people giving the impression that it must have become corrupted. However, with forensic tools like P2 Commander, an investigator can easily see through such a camouflage in a single glance as we have seen in the example.

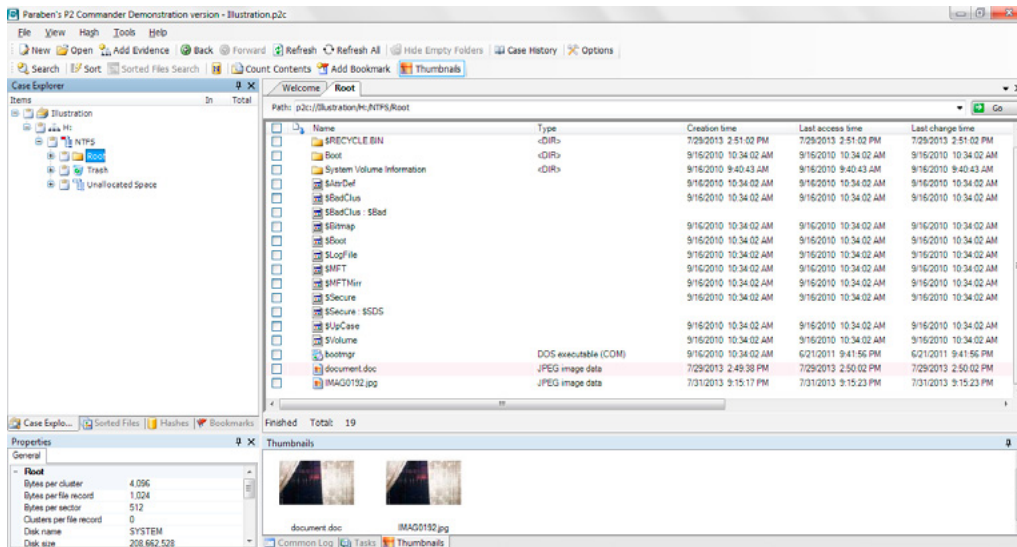


Figure 11. Viewing contents of Root directory in drive H

The times and dates of creation, last access, last modification and last change of document can also be viewed in the P2 Commander tab for selected directory, such time stamps can be used as proof for or against a suspect. For instance, if there is a child pornography image found on an endpoint that is shared by employees on shift basis, the time stamps could be used to determine on whose shift such an image was stored and accessed. Reviewing sign-in logs and registers along with CCTV footage can further corroborate this.

The trash directory can also be analyzed (note, the recycle bin was emptied before this analysis). Clicking on the trash directory shows contents emptied from the recycle bin. Despite the recycle bin having been emptied, we can see the contents that were deleted from the recycle bin that have not been overwritten. In Figure 12, we can see the deleted items with their file extensions and document types as well as the date and time of the deletion. As had been seen previously, the deleted item also has a .doc file extension, but the document type JPEG image data. Also, at the bottom left corner of the interface, we have the file properties which shows that the document was deleted (stated that *Deleted* is *True*) and the path was recycle bin.

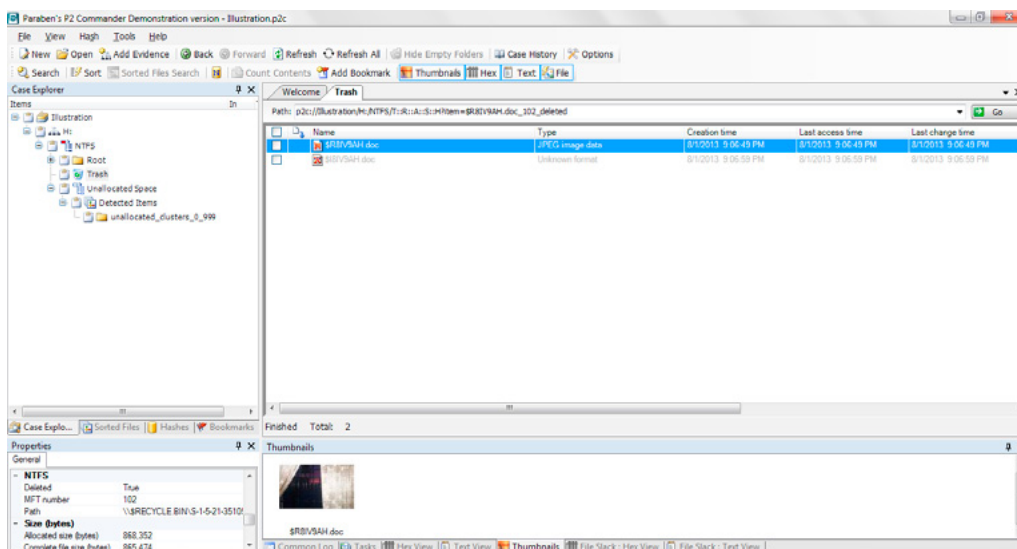


Figure 12. *Contents of trash directory*

Expanding the *Unallocated Space* directory shows the *Deleted Items* directory, which can be further expanded to reveal the unallocated clusters. In the unallocated clusters directory, we can see there is a JPEG image data document as shown in Figure 13. The contents of the unallocated clusters are reviewable

and recoverable documents, partially or fully, those were deleted from the recycle bin, but have not been fully overwritten. Such data might be very useful in a case. Casey (2004) gives an example of a blackmail case where the suspect claimed the blackmail letter document was a letter he originally wrote, but someone else modified and sent it while he was away on vacation. Various fragments of deleted material were recovered from his computer, one of the fragments in the slack space of another file (the owning file), which was created a couple of days before the suspect's vacation. Technically, this showed that the slack space had existed before the owning file, which helped to question the suspect's alibi.

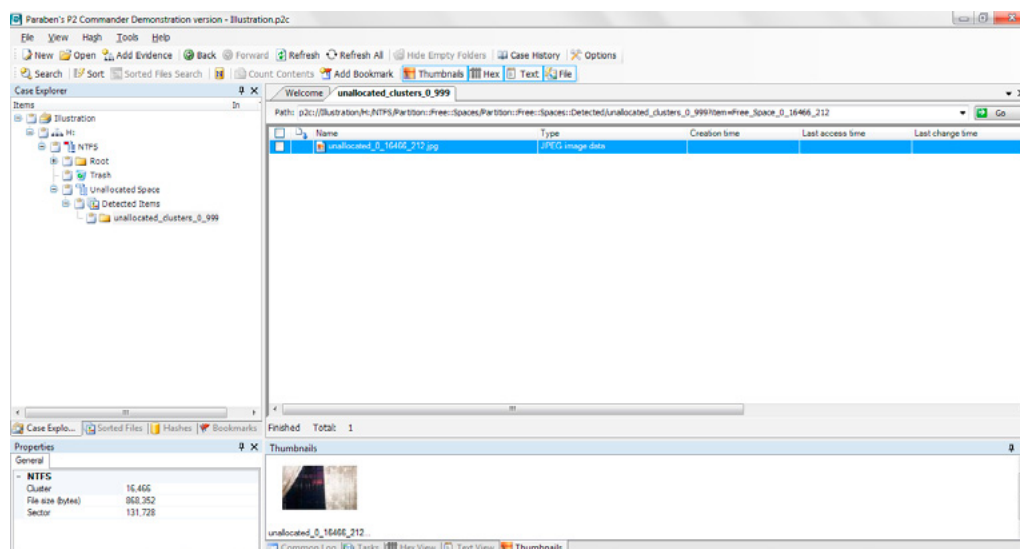


Figure 13. Contents of the unallocated clusters

CREATING A FORENSIC CONTAINER

The created case can be saved in a forensic container. Paraben has forensic containers that are encrypted and write protected such that the digital evidence can be stored securely and makes it easier for third parties to review the evidence. This helps to ensure a proper chain of custody and to show that the evidence was not tampered with or contaminated during storage. A forensic container is from *Tools* in the menu bar and *Create New Forensic Container* selected, as shown in Figure 14. Once clicked, the pop up shown in Figure 15 comes up which requires a file path to be chosen for saving the container. By default the containers are saved to the Paraben Corporation directory located program files directory, and saved to a folder called containers there in a folder called new_container (the previous directory used for saving the case is used in the example). There is also the need to select a password for the container, and the password confirmed. Once that is done, the forensic container is stored in the selected directory and can be analyzed and reviewed when and as necessary.

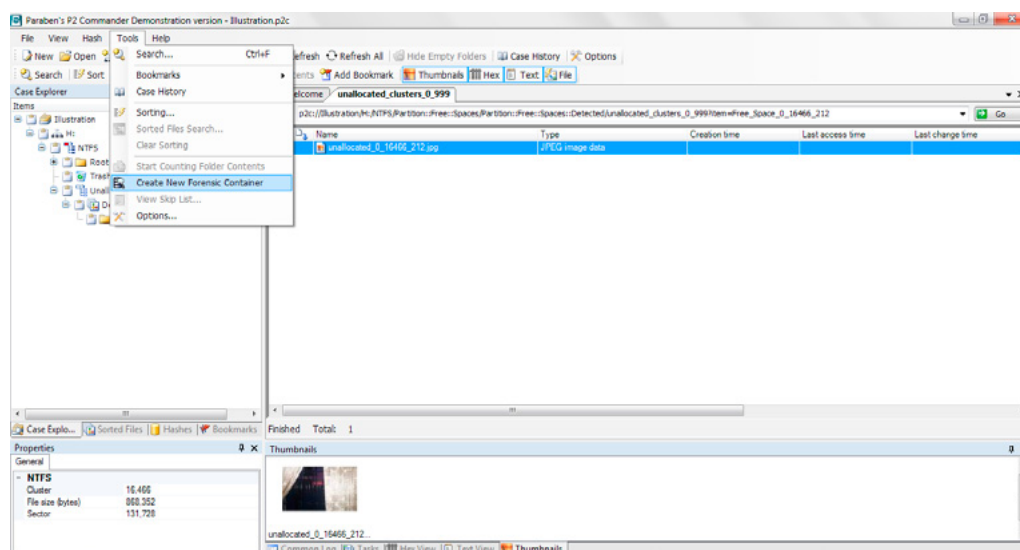


Figure 14. Create New Forensic Container option under Tools

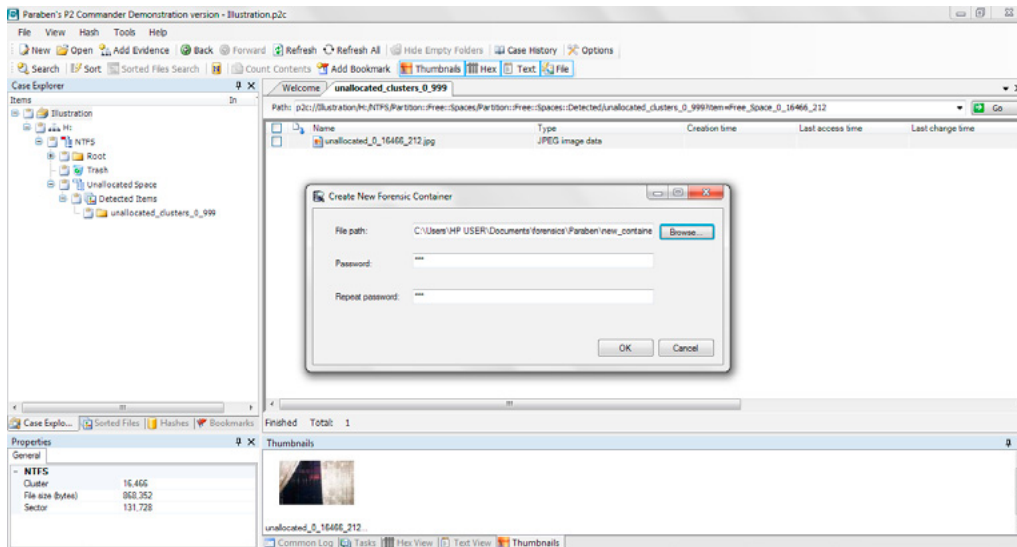


Figure 15. File path selection and password creation for forensic container

Figure 16 shows the directory containing the forensic container. The directory contains two files: the main file and the data file; the main file contains the file hierarchy, which is named after the forensic container name with file extension *.p2d*, while the data file contains the acquired data evidence (Paraben Corporation, 2013).

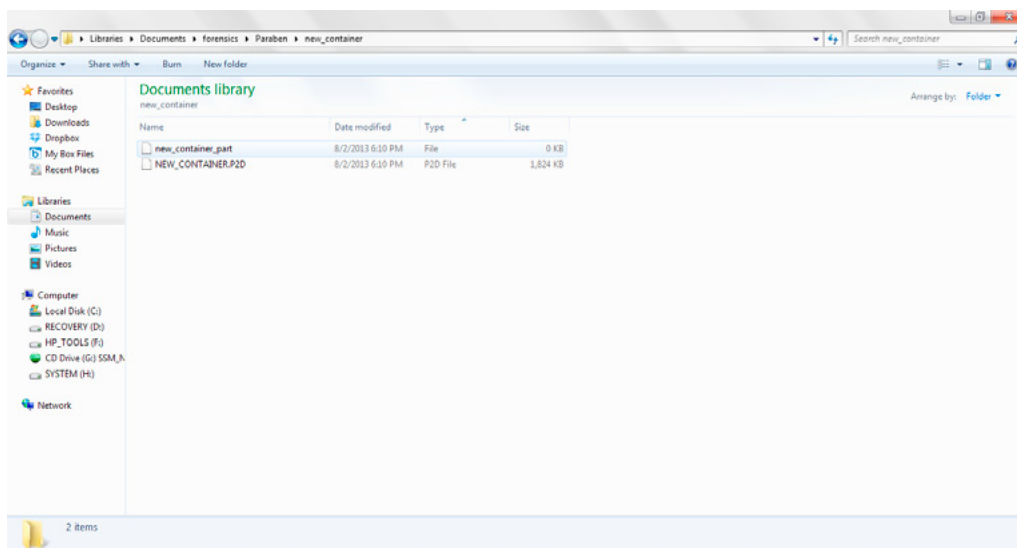


Figure 16. Forensic container directory containing case

To view the forensic container, click on *Add Evidence* bring up add evidence pop up, select other under category where *Forensic container file* is visible as show in Figure 18, click on *Forensic container file* and okay. Once okay is clicked, the program browses to the directory containing the case. There the *new_container* folder is opened and the *NEW_CONTAINER.P2D* file selected (the only thing visible in the container in this example, *.p2d* files are types to be selected). This brings up a pop up to enter new evidence name as shown in Figure 18, the default name *NEW_CONTAINER* is left in the example. Then the program prompts to enter the forensic container password (Figure 19). That done, the new container directory becomes visible in the case explorer pane (Figure 20). The container can be expanded to view captured evidence (reports are not available in the demo version, but are in the full version) and audit logs.

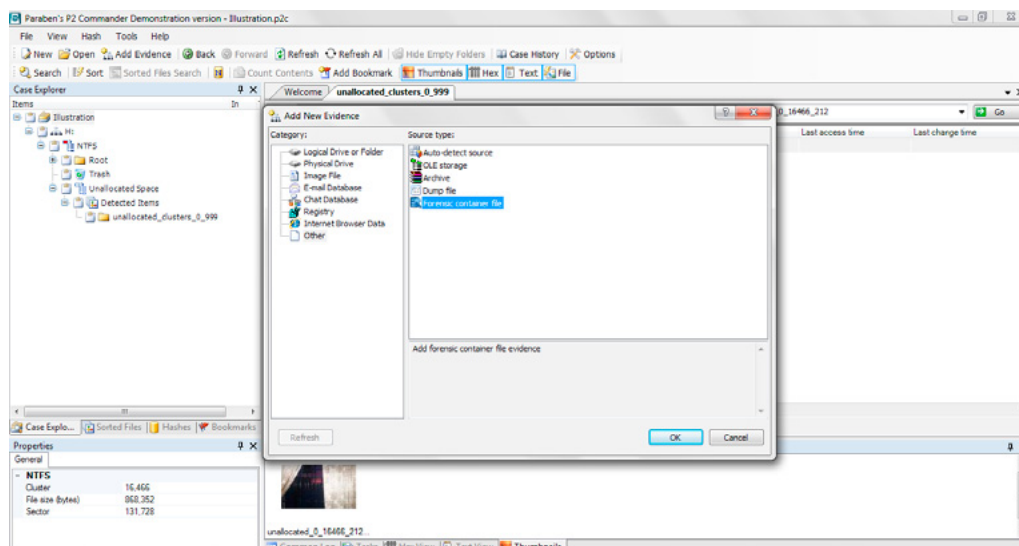


Figure 17. *Selecting Forensic container file*

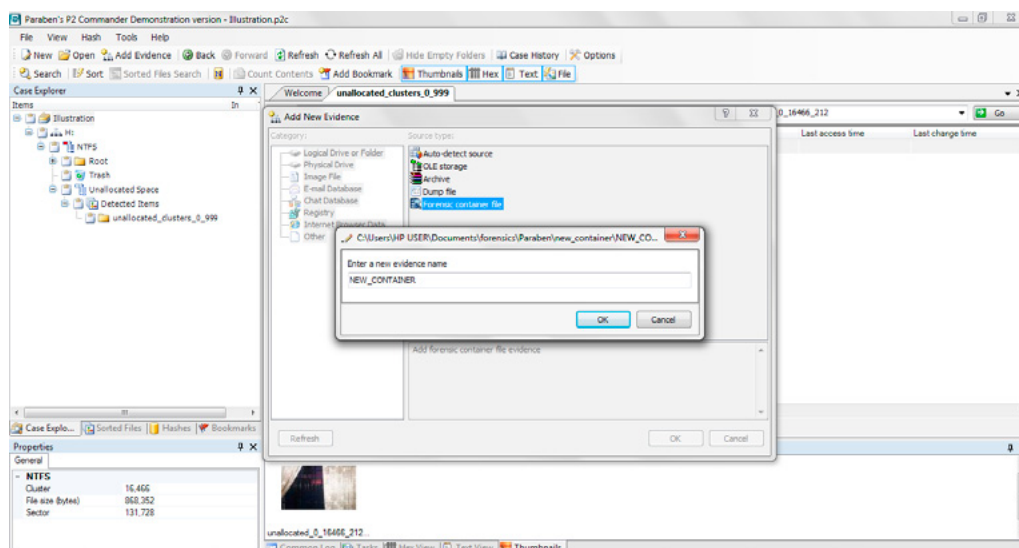


Figure 18. *Entering new evidence name*

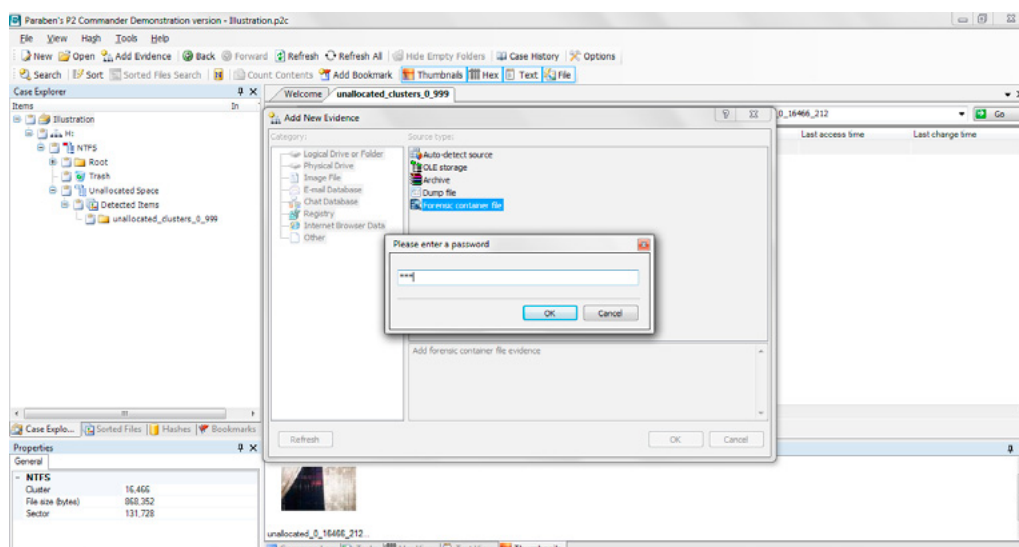


Figure 19. *Entering forensic container password*

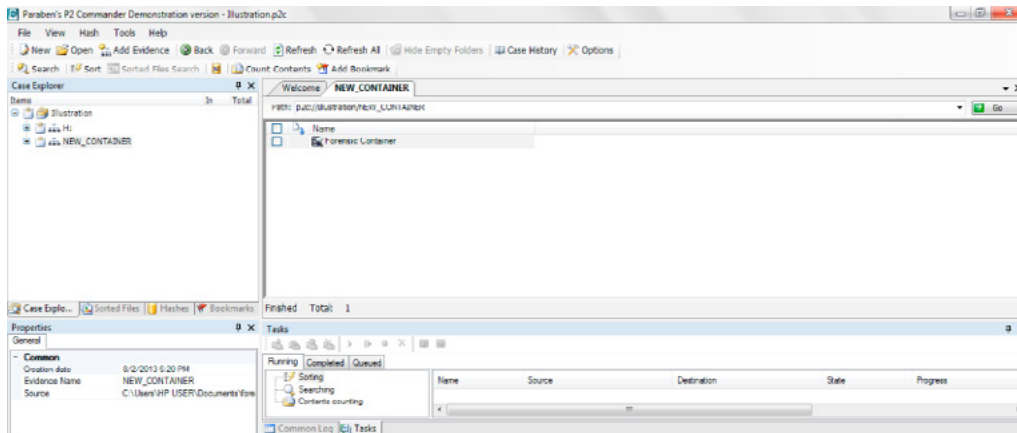


Figure 20. New container directory in the case explorer pane

SEARCHING FOR SUSPICIOUS IMAGE FILES

P2 Commander can be used to search for suspicious images like pornography. This can be very useful where investigating employee misconduct in terms of endpoint usage, sexual harassment or child pornography. A directory can be analyzed for such suspicious images selecting the directory and using the *Sorting* option under *Tools* from the menu bar. In Figure 21, the H directory is selected for sorting, that will search drive H and all the subdirectories in it for suspicious material.

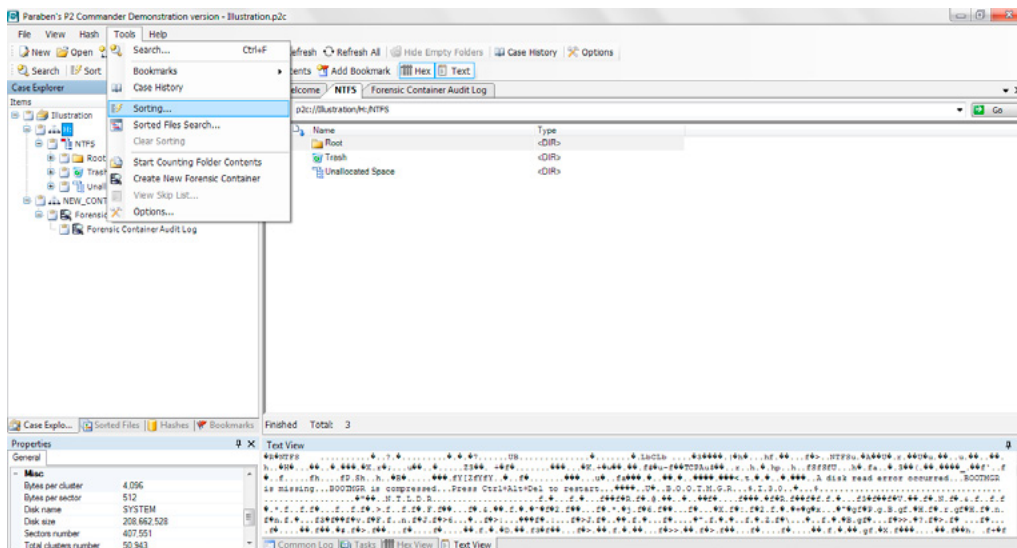


Figure 21. Sorting option under Tools

Once *Sorting* is clicked, the P2 Commander Sorting Engine pops up starting with general options for the sorting, files with undetected format and deleted data are added to the sort options in our example as shown in Figure 22. The next step is the *Image Analyzer Options*, which is selected specifically for detection of potentially pornographic material. The sensitivity of the image analyzer engine can be increased or decreased; increasing it makes it increase the number of files that will be labeled suspect, while decreasing reduces such. The default sensitivity level is 75 as used in our example (Figure 23). The file filter is used to restrict the files to be search by the image analyzer to a particular size, and the resolution filter restricts the search to resolution size. Both file filter and resolution filter are not used in the example. The final step is the *Advanced Options*, which offers additional search criteria like email attachment storage searches and some criteria that can be skipped (Figure 24), but nothing is selected in the advanced options in our example. Then finish is clicked to start the sorting. The process is shown in the *Task* pane at the bottom of the interface where the status can be seen to be running while the sorting is taking place. Once completed, the status can be seen as completed under the *Completed* tab in the *Task* pane (Figure 25).

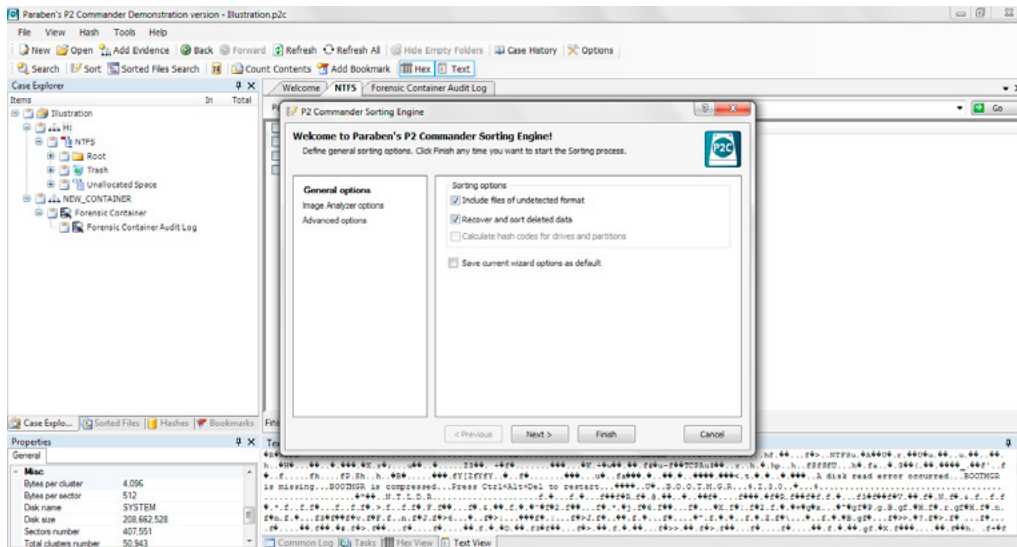


Figure 22. P2 Commander Sorting Engine general options

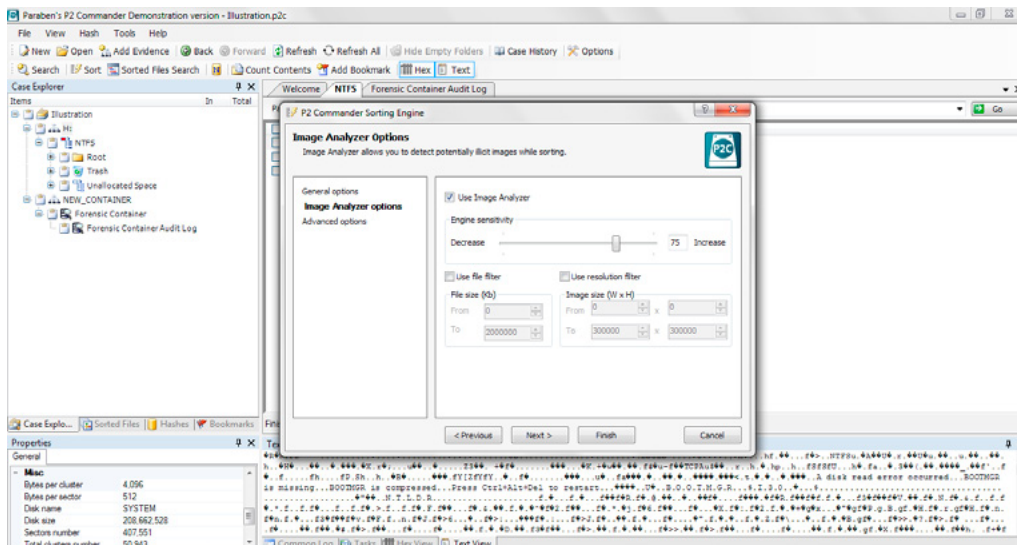


Figure 23. Image Analyzer Options of the sorting engine

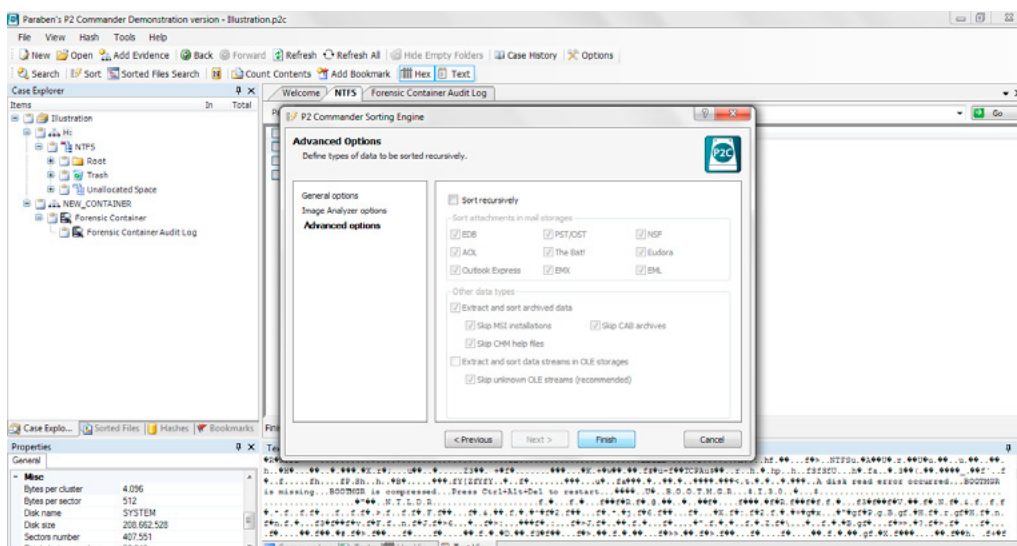


Figure 24. Advanced Options of the sorting engine

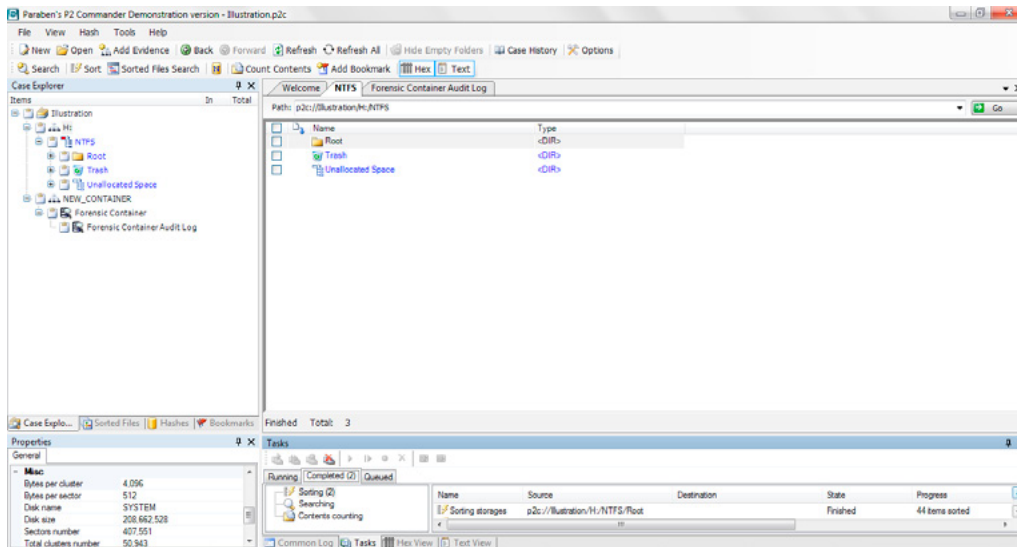


Figure 25. Completed tasks in the Completed tab of the sorting engine

The *Sorted Files* tab located under the *Case Explorer* view just to the right of the *Case Explorer* tab is clicked to view sorted files. The image analyzer results can then be expanded to see if there are any suspect files in the drive (Figure 26). It can be seen in the example that there are three items in the image analyzer results; two are low suspect and one highly suspect. Clicking on the low suspect directory reveals the two documents that we had previously seen on the drive: the image file and the apparent document file. Notice as before, the image analyzer is also not deceived by the change in file extension of the image named *document.doc* and reveals its actual file type and content (Figure 27); so a criminal trying to hide an inappropriate picture by changing the file extension would not be able to hide from the forensic investigator using a tool like Paraben's P2 Commander. A review of the highly suspect result shows an image of a nude hand – which was added to the drive (Figure 28). The image analyzer recognizes skin tones and shapes which are like sensitive human body parts, and hence flags any image that may look so, in our example the hand has a consistent tone that reflects uncovered skin and the shape also looks like other parts of the human anatomy. Porn detection software usually use criteria as skin colour, shape, consistent tone and colour spread over a space, and in the case of videos movement may be an added criterion to determine which files to flag as suspect.

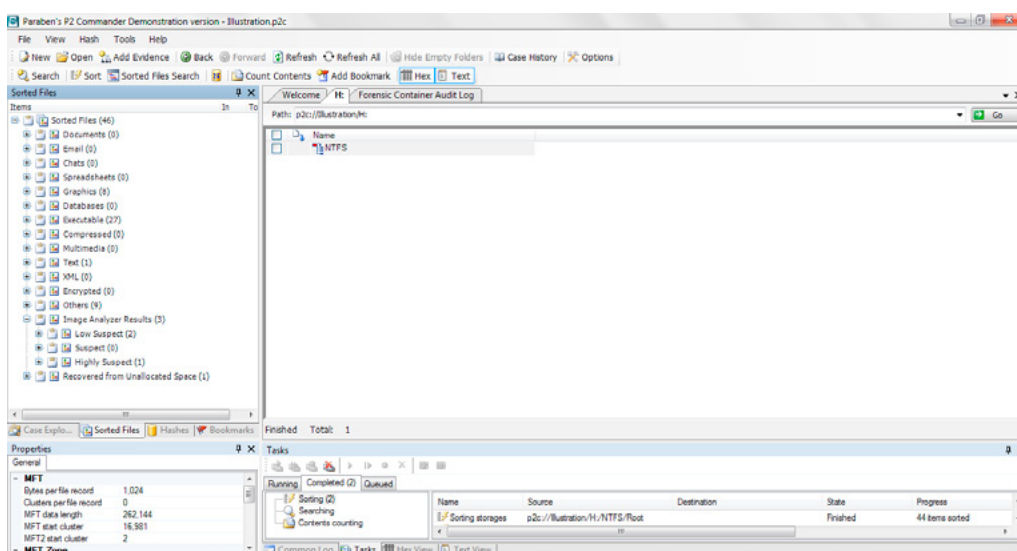


Figure 26. Sorted files view

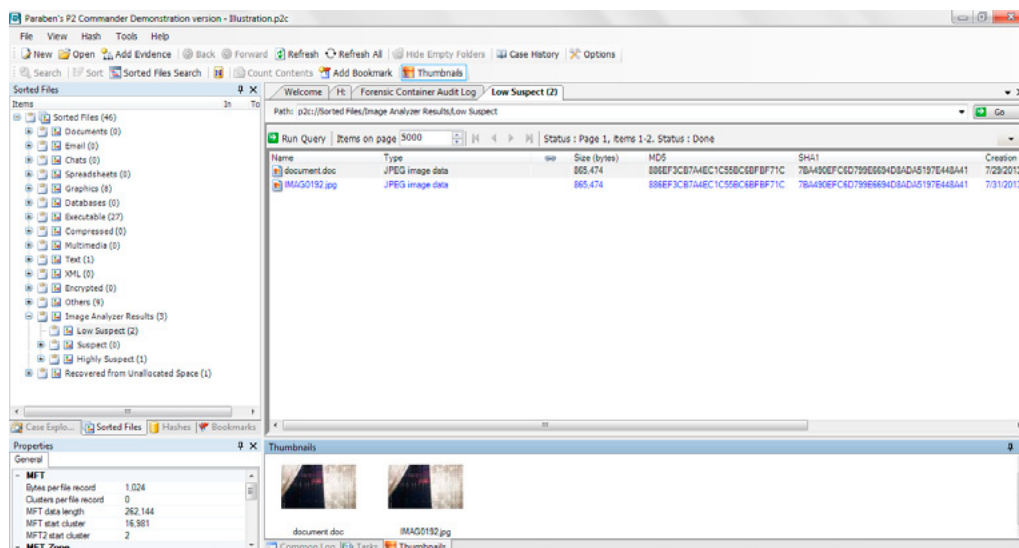


Figure 27. Review of low suspect images

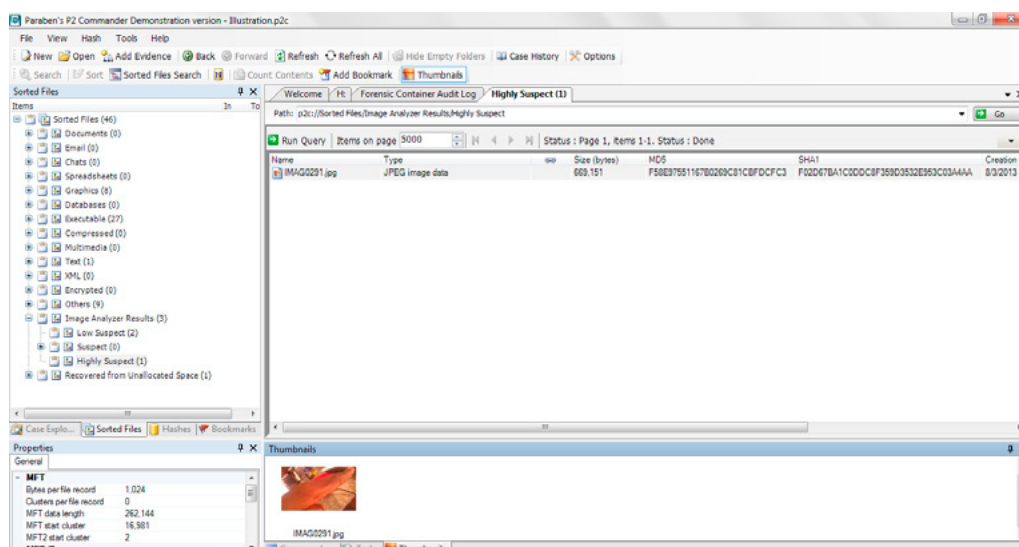


Figure 28. Review of highly suspect image file

An investigation may be carried out with the primary aim of detecting suspicious pornographic files and images which may be relevant to cases such as sexual harassment, employee misconduct or child pornography, which implies that, in the case of a criminal investigation, a warrant was obtained and authorization was given to search for such material. However, such files may be uncovered in the course of an investigation that was not primarily nor directly linked to the images. In such a situation, the investigator should best not pursue such files until he has stated that such data are available to the appropriate authorities and is granted authority to continue searching for and collecting such data as evidence. Going ahead to investigate and review such pornographic data without due authorization in an investigation that is not related might result in sanctions against the investigator, and presenting such data as evidence would most likely be thrown out.

CONCLUSION

Every operating system has its unique peculiarities in terms of operations, which can determine how to go about investigating it successfully. An investigator needs to be familiar with the operating system(s) on suspect machines that need to be investigated for evidence in any case to ensure evidence is properly and reasonably collected in a forensically sound manner. Windows is a very commonly used operating system, and therefore digital forensics investigators need to be familiar with the operating system and tools for investigating and analyzing it. There are many digital forensic investigation tools available,

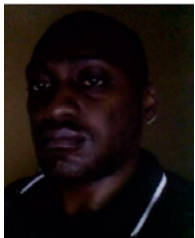
many of them can be used across multiple platform and operating systems, especially Microsoft Windows. Paraben's P2 Commander is quite robust and is very effective for many investigations that will need to be carried out on Windows systems.

Digital forensic tools continue to evolve as technology and the bad guys evolve in a bid to tackle digital crimes and offenses. Techniques used by offenders to mask their wicked activities can be unmasked with digital forensic tools (like trying to hide files by changing file extension). However, the tools might tend to be developed after-the-fact as the bad guys usually tend to be a couple of steps ahead in terms of technology, they are always constantly working to beat any development that has been made to track and apprehend them. That notwithstanding, digital forensic tools are still equal to the task of catching most offenders, and as stated continue to evolve to meet up with new challenges.

REFERENCES

- Casey, E. (2004) Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet. 2nd ed. Elsevier Academic press.
- Franklin, C. and Coustan, D. (2013) How Operating Systems Work [Online]. Available from: <http://computer.howstuffworks.com/operating-system1.htm/printable> (Accessed: 30 July 2013).
- Microsoft (2003) How NTFS Works [Online]. Availble from: [http://technet.microsoft.com/en-us/library/cc781134\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc781134(v=ws.10).aspx) (Accessed: 30 July 2013).
- Oppenheimer, P (n.d) File Systems Forensics: FAT and NTFS [Online]. Available from: <http://www.priscilla.com/Courses/ComputerForensics/pdfslides/FileSystemForensics.pdf> (Accessed: 30 July 2013).
- Paraben Corporation (2013) P2 Commander How To. Paraben Corporation.

ABOUT THE AUTHOR



Dauda Sule, CISA. He is currently the Marketing Manager of Audit Associates Limited which is a consultancy firm that specializes in designing and organizing training programs pertaining to auditing, fraud detection and prevention, information security and assurance, and anti-money laundering. He is a CISA and has an M.Sc. in Computer Security from the University of Liverpool. Dauda also has a first degree black belt in Taekwondo. He has previous experience of over five years in the Nigerian Banking industry, and also did some time in Gtech Computers (a computer and allied services company) as a systems security and assurance supervisor.

PTK Forensics professional

Collaborative

Multi-tasking

Easy-to-use

Case and
Evidence
Management

MAIN FEATURES

RAM
Analysis

Registry
Analysis

e-mail
Analysis

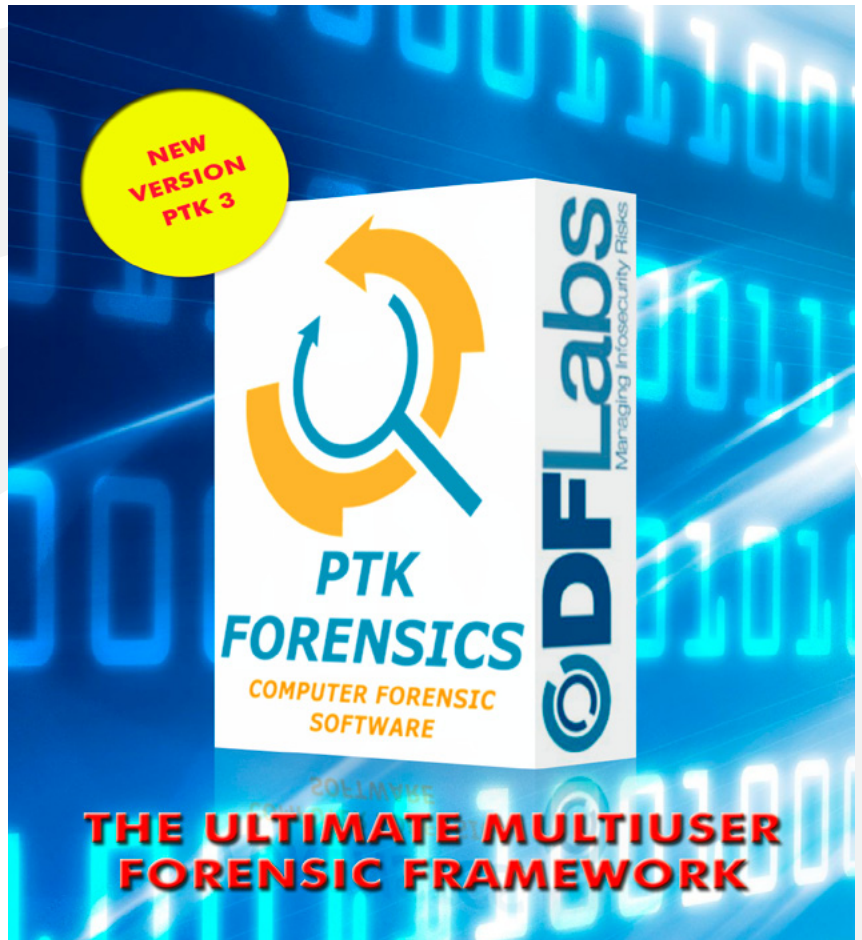
Timeline

Gallery

Keyword Search

Pre-Processing

Advanced
Reporting
System



SPECIAL PROMO 15% OFF
single user perpetual license

<http://www.ptkforensic.com>

promo code **E-FORNCS13**

HOW TO USE ENCRYPTED ITUNES BACKUPS

FOR SMS HISTORY WITHOUT THE DEVICE OR JAILBREAKING

by Gouthum Karadi, CISSP, CEH, MBA

A client comes to our firm to find out whether an intern took unauthorized photos of confidential talking points in order to warn the competition. Though the suspect had a private iPhone, he backed it up to a corporate system. We use the backup to correlate unauthorized activity with corporate policy.

What you will learn:

- How to use the Apple Macintosh as an iOS Forensics platform
- What iTunes 11.04 backs up from your iPhone
- Where the backup is stored on
- How to decrypt the backup with or without the password
- How to extract and correlate files within the backups to mobile activity

What you should know:

- How to use Apple Macintosh OS X 10.8.x
- Standard source control management syntax
- Basic terminal navigation and sudo
- SQL database navigation
- How to install and configure software from dmg, binary, or source

Imagine it is late Friday afternoon at Forensics, Inc. and you get a call from ABC Corp, one of your top clients. It seems that ABC had competitor XYZ cornered and agreeing to submit to a deal before a timely lunch. Yet when talks resumed after the break, XYZ began to negotiate more fiercely. The opponent began to negotiate using not only the exact tactics that ABC prepared for, but *even using the exact words in some cases*.

How could XYZ know what ABC was planning? Someone had to have leaked the internal talking points memorandum the morning of the negotiation. Whether there was any legal action available at this point was moot, what was more important was that the leak get plugged. The firm called us, their trusty forensics investigators to examine what happened. We immediately reassured them that we could help investigate how a leak may have occurred.

Since the time window was so narrow, and the individuals with access to the memo in question so small, this became a simple process of elimination. It seems that only one junior associate had access to the document and for a short period of time. This individual used a company provided Apple MacBook Pro, and a personal iPhone 4.

Since there was no Mobile Device Management (MDM) or Mobile Security Policy in place we had only a 13" MacBook Pro as evidence with a standard Acceptable Use Policy naming it as corporate property for corporate use. We focus our investigation on this one device and discover that the employee in

question backs up his phone to it religiously. All we have to do then, is to extract photo history, call log, and SMS messages from the backups.

GIVEN

Macbook Pro 13"

- OS X 10.8.x
- iTunes 11.04
- iPhone 4 – iOS version 6.1.4 Backups

GOAL

Extract SMS messages and Photos taken with Employee iPhone yet *stored* on corporate assets.

EXECUTIVE SUMMARY

Although the process itself was quick, yielding significant results, this article will use this case as a Proof-of-Concept (POC) of what is available from iTunes backups without any special tools. First we will describe what iTunes backups are, what they synchronize. Then we will show the formats and locations of the backup files. Next we will view them using freely available file recovery utilities. Finally we will show you common tools for extracting and correlating files.

ITUNES BACKUP CONTENTS

iTunes has two backup settings. One, is the full backup, which backs up all of the data in Table 1. The second is when it synchronizes, or “syncs” across Wi-Fi or by USB cable. Our goal is to extract what data is available and correlate it to the specified time period. Did the employee take a picture of the talking points memo? Did the employee then send it by SMS to opposing counsel?

ITUNES WILL BACK UP THE FOLLOWING INFORMATION

- Contacts* and Contact Favorites (regularly sync contacts to a computer or cloud service such as iCloud to back them up).
- App Store Application data including in-app purchases (except the Application itself, its tmp and Caches folder).
- Application settings, preferences, and data, including documents.
- Autofill for webpages.
- CalDAV and subscribed calendar accounts.
- Calendar accounts.
- Calendar events.
- Call history.
- Camera Roll (Photos, screenshots, images saved, and videos taken. Videos greater than 2 GB are backed up with iOS 4.0 and later). Note: For devices without a camera, Camera Roll is called Saved Photos.
- Game Center account.
- Home screen arrangement.
- In-app purchases.
- Keychain (this includes email account passwords, Wi-Fi passwords, and passwords you enter into websites and some other applications. If you encrypt the backup with iOS 4 and later, you can transfer the keychain information to the new device. With an unencrypted backup, you can restore the keychain only to the same iOS device. If you are restoring to a new device with an unencrypted backup, you will need to enter these passwords again.)
- List of External Sync Sources (MobileMe, Exchange ActiveSync).
- Location service preferences for apps and websites you have allowed to use your location.
- Mail accounts (mail messages are not backed up).
- Installed Profiles. When restoring a backup to a different device, installed configuration profiles are not restored (such as accounts, restrictions, or anything which can be specified through an installed profile.) Any accounts or settings that are not associated with an installed profile will still be restored.
- Map bookmarks, recent searches, and the current location displayed in Maps.
- Microsoft Exchange account configurations.
- Network settings (saved Wi-Fi hotspots, VPN settings, network preferences).
- Nike + iPod saved workouts and settings.

- Notes.
- Offline web application cache/database.
- Paired Bluetooth devices (which can only be used if restored to the same phone that did the backup).
- Safari bookmarks, cookies, history, offline data, and currently open pages.
- Saved suggestion corrections (these are saved automatically as you reject suggested corrections).
- Messages (iMessage and carrier SMS or MMS pictures and videos).
- Trusted hosts that have certificates that cannot be verified.
- Voice memos.
- Voicemail token. (This is not the voicemail password, but is used for validation when connecting. This is only restored to a phone with the same phone number on the SIM card).
- Wallpapers.
- Web clips.
- YouTube bookmarks and history.

* Your contacts are part of the backup to preserve recent calls and favorites lists. Back up your contacts to a supported personal information manager (PIM), iCloud, or another cloud-based service to avoid any potential contact data loss.

Note: Syncing phones does not copy all of the data of backups.

ICLOUD BACKUPS

To extract iCloud Backups, one needs the iCloud backup password. The same technique used below can be used to extract it from the user keychain. Once available the examiner must either restore the backup to another device and then backup and extract, or use ElcomSoft's Phone Password Breaker (EPPB) Pro. Note that the user will be notified if you restore to another device. The legalities of this need to be thoroughly examined. In our case it would be illegal without some form of warrant and legal investigation-or, as we recommend to our clients a Mobile Device Policy at the minimum. Mobile Device Management with Restrictions and Policies can further protect firms.

BACKUP FORMAT

As Apple has shown us in their documentation any picture or SMS sent from an iOS device is backed up. Now we only need to find the backups in question and extract them. In their raw state the files are difficult to read. They use hexadecimal format and are stored in the following locations.

ITUNES PLACES THE BACKUP FILES IN THE FOLLOWING PLACES

Mac: ~/Library/Application Support/MobileSync/Backup/

Windows XP: \Documents and Settings\ (username) \Application Data\Apple Computer\MobileSync\Backup\ Note: To quickly access the Application Data folder, click Start, and choose Run. Type %appdata% and click OK.

Windows Vista and Windows 7: \Users\ (username) \AppData\Roaming\Apple Computer\MobileSync\Backup\ Note: To quickly access the AppData folder, click Start. In the search bar, type %appdata% and press the Return key.

GO TO SYSTEM FOLDERS

In our situation we are searching an OS X System. Since you cannot browse to protected system file folders by default, we use the <Command+Shift+G> key combination in Finder. Or, you can use the Go | Go to Folder menu option.

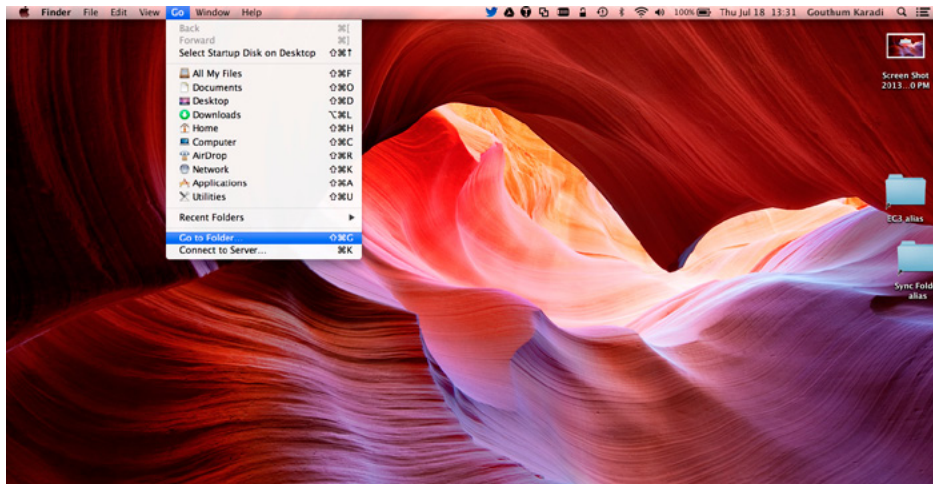


Figure 1. Go to Folder menu

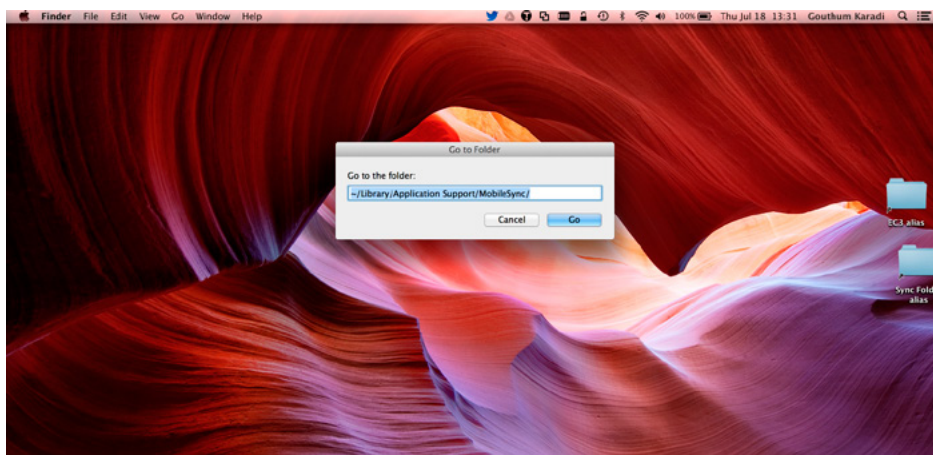


Figure 2. <Command + Shift + G> Dialog Box

MOBILESYNC BACKUP FOLDER

Once the popup is open, type /Users/<Username>/Library/MobileSync/Backup. Within the backup folder the investigator will find a series of subfolders with hexadecimal titles. Older Backups have the date time groupings appended to them.

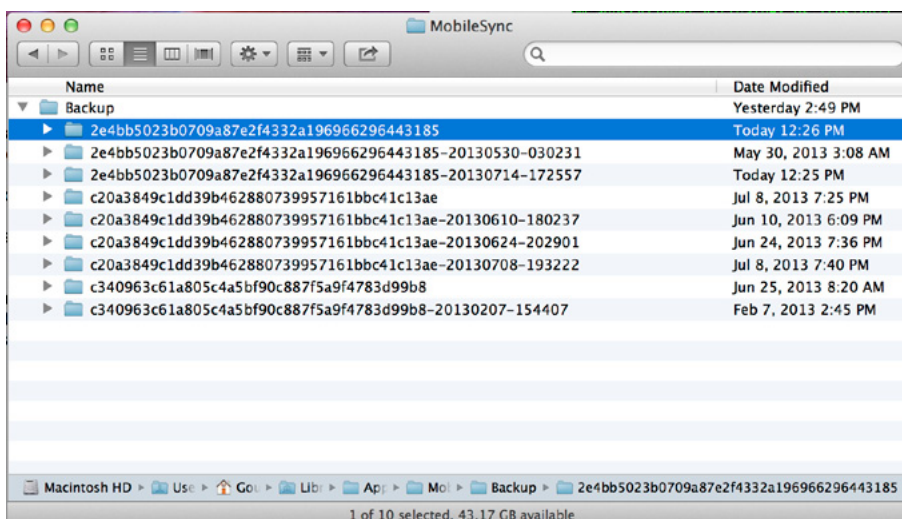


Figure 3. MobileSync folder

KEY PLIST FILES

Before expanding the Hex, one can examine three key .plist files: Info.plist, Manifest.plist, and Status.plist. Using xCode on the Mac, or any other plist viewer, one can see what each of the files' contents. Below you will find a table of the plist files and some notable info.

Name	Date Modified
ffa6f6dbcb39e49249a4676e36037146ae7d1382	Jun 10, 2013 5:21 PM
ffb5bdbe14c2160792666729c2a69ab8e9e935b2	Jan 25, 2013 6:03 AM
ffb54c3c8f1d91f6ed61ec49b1f68ac60315ee5	Jul 8, 2013 7:24 PM
ffb56ec1e7230a40e2825ac7e560bd2600ef80b	May 3, 2013 12:19 PM
ffe1bb7f06bc9b04e5068e2f7cc03a4cfbdec665	Jan 25, 2013 6:03 AM
fff6b2c7ff19a9ab9d7d0ae672b7df97bca33d24	Jul 8, 2013 7:24 PM
fff59da7dec00e1acba5c59439de03c176881f04	Jan 25, 2013 6:03 AM
ffcece7c11af595b7a74d35a953b1bb200e7dc4	Jul 8, 2013 7:24 PM
Manifest.mbdb	Jul 8, 2013 7:25 PM
Info.plist	Jul 8, 2013 7:25 PM
Manifest.plist	Jul 8, 2013 7:25 PM
Status.plist	Jul 8, 2013 7:25 PM
c20a3849c1dd39b462880739957161bbc41c13ae-20130610-180237	Jun 10, 2013 6:09 PM
c20a3849c1dd39b462880739957161bbc41c13ae-20130624-202901	Jun 24, 2013 7:36 PM
c20a3849c1dd39b462880739957161bbc41c13ae-20130708-193222	Jul 8, 2013 7:40 PM
c340963c61a805c4a5bf90c887f5a9f4783d99b8	Jun 25, 2013 8:20 AM
c340963c61a805c4a5bf90c887f5a9f4783d99b8-20130207-154407	Feb 7, 2013 2:45 PM

Figure 4. *Plist files in backup folders*

BACKUP .PLIST FILES

Info.plist

- Device Name/Display Name
- iTunes Version
- GUID/ICCID/IMEI
- Installed Apps
- Phone Number
- Product Type (for example, iPhone5,1)
- Product Version “7” (iOS 7 in this case)

Manifest.plist

- Device Name/Serial Number
- Encryptions Settings (YES/NO)
- WiFi Settings
- Passcode Settings (YES/NO)
- Backup Keybag
- Applications

STATUS.PLIST

- Details of backup
- Full Backup (YES/NO)
- Backup State (New)
- Date

ENCRYPTED BACKUPS

When a user backs up an iOS device to system, the backups are not encrypted by default. If they are not encrypted, they do not capture Keybag information. If they are encrypted with a password, they not only have key information, but these keys can be restored to a *different iOS device*.

If the examiner does not have the password to decrypt the backup they must access the device to seize this information. When the device is unavailable, as in this case, the backup password may be stored in the users Keychain in OSX. The root user must unlock a users keychain using their login password to

extract the iTunes password for decryption. When viewing encrypted backups in their raw state your editor of choice will show human unreadable symbols.

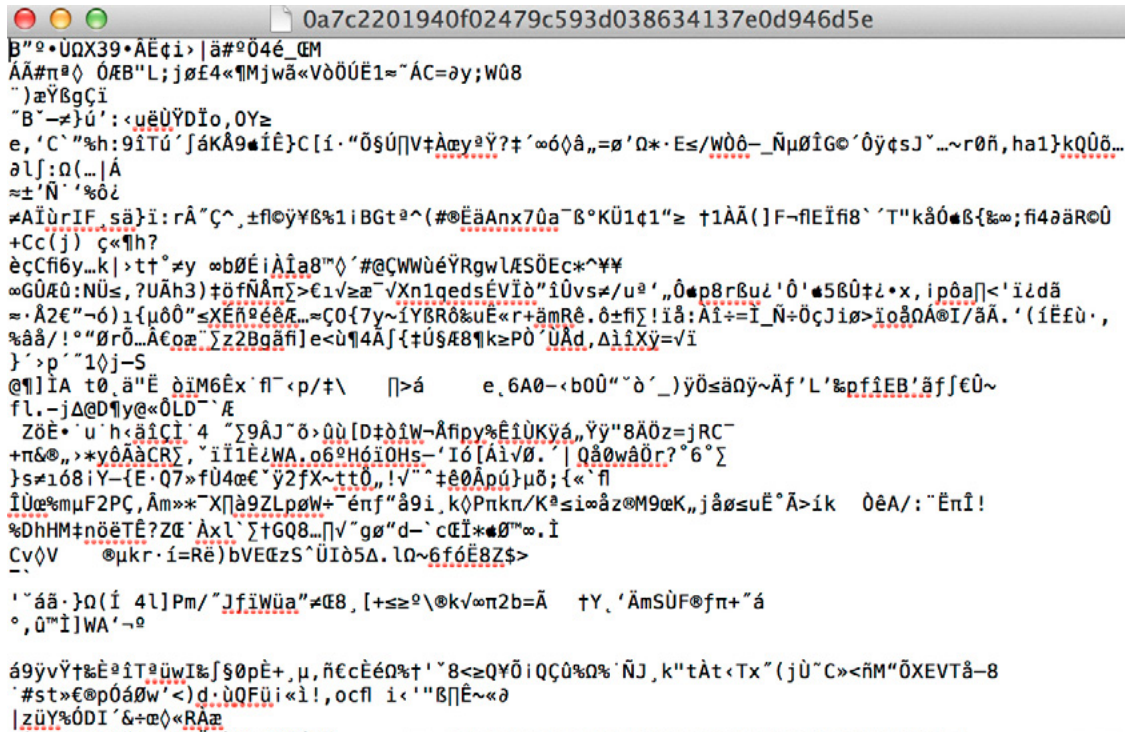


Figure 5. Encrypted backup file contents

NON-ENCRYPTED FILES

Non-encrypted backups will either open in the editor of choice for your system, .jpeg, .xml etc., or show an icon.

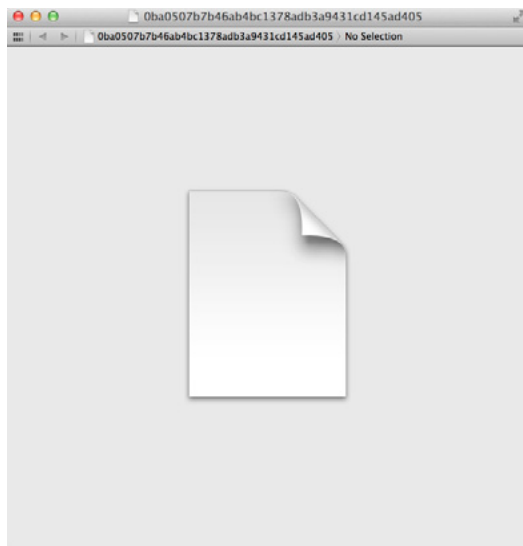


Figure 6. Unknown file

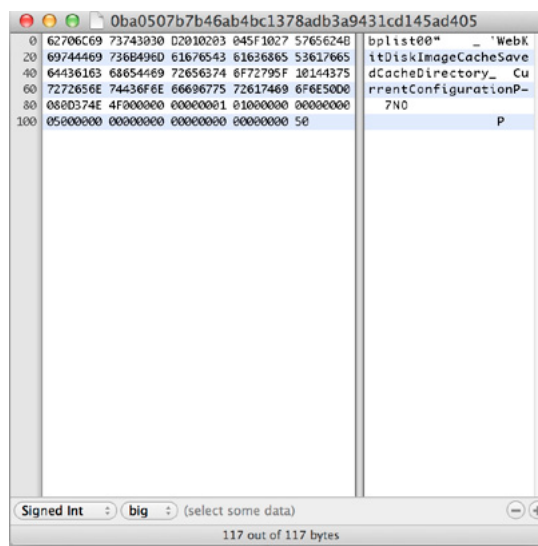


Figure 7. Same file in Hex Fiend

ITUNES BACKUP DECRYPTION

To unencrypt the backups one requires the iTunes backup password. There are several ways of retrieving this password. If the forensics team possesses the phone and has the right to handle it, it is a process of using the keychain on phone itself. Because this process requires booting a custom image on the phone it requires significant exposition beyond the scope of this article which focuses on the backup itself.

However, one can also bruteforce the backup using several paid iTunes backup password breakers:

- ElcomSoft Phone Password Breaker (EPPB)
- iTunes Backup Password Recovery
- Smartkey iTunes Backup Password Recovery

Of course we prefer open source and non-paid solutions when performing POCs. A forensic examiner must be able to explain how the tools being used work or risk having their information discounted. We extract the users login password hash and then bruteforce/dictionary attack it.

PASSWORD HASH LOCATION

Each user on OSX has a .plist file with their password hash located in `/var/db/dslocal/nodes/Default/users`. It is named `<username>.plist`.

Key	Type	Value
Root	Dictionary	(21 items)
jpegphoto	Array	(1 item)
authentication_authority	Array	(2 items)
passwordpolicyoptions	Array	(1 item)
picture	Array	(1 item)
_writers_picture	Array	(1 item)
hint	Array	(1 item)
shell	Array	(1 item)
_writers_realname	Array	(1 item)
realname	Array	(1 item)
name	Array	(1 item)
_writers_UserCertificate	Array	(1 item)
home	Array	(1 item)
KerberosKeys	Array	(1 item)
ShadowHashData	Array	(1 item)
Item 0	Data	<62706c69 73743030 d101025f 10145341 4c544544 2d534841 3531322d 50424b44 4632d303 04050607 085
uid	Array	(1 item)
_writers_passwd	Array	(1 item)
generateduid	Array	(1 item)
gid	Array	(1 item)
passwd	Array	(1 item)
Item 0	String	*****
_writers_hint	Array	(1 item)
_writers_jpegphoto	Array	(1 item)

Figure 8. plist with password hash

While types of encryption are beyond the scope of this article, please note that SMB file sharing uses MD4 a far weaker encryption than PBKDF2 (Salted SHA512) the default for Mountain Lion. Our solution uses the DaveGrohl utility for OS X Mountain Lion to crack the password.

It can also export the hash for use with John the Ripper and use distributed attacking. In this example our process takes just over 3 hours for an eight character alphanumeric password. We utilize a Macbook Pro Retina QuadCore 2.7GHz i7 with 16GB 1600 MHz DDR3. More complicated passwords take significantly longer or require greater computing power.

MacbookPro:DaveGrohl Administrator \$ `sudo ./dave -u iphoneuser --characters=*****`

- Loaded PBKDF2 (Salted SHA512) hash...
- Starting attack

```
urT] [TsnttrT] [torTlrT] [unnsruT] [lTlosuT] [ssououT]
0003:02:49      410,962 (kirstyl23) (burliest) [soTunTT] [ostnlTT] [lrrurT] [rsnttrT] [torTlrT]
[unnsruT] [lTlosuT] [ssououT]
0003:02:49      410,964 (kirstyl23) (burlily) [soTunTT] [ostnlTT] [TurrurT] [rsnttrT] [torTlrT]
[unnsruT] [lTlosuT] [ssououT]
0003:06:36      419,199 (popstar1) (cabalic) [tnnsnTT] [ourl1TT] [lTouurT] [ruuotrT] [nnorlrT]
[utunruT] [tlsttuT] [sustouT]
```


- Found password: '*****'
- (dictionary attack)

Finished in 13819.382 seconds / 514,951 guesses...

37 guesses per second.

Now that we have the login password, the process is simple. Login to the Macbook Pro, launch Keychain Access, and select the iTunes Backup password from the "All items" section. Double-click the iPhone Backup entry and select the "Show password" checkbox. Assuming you have the right login password, the dialog will show the password in the text field.

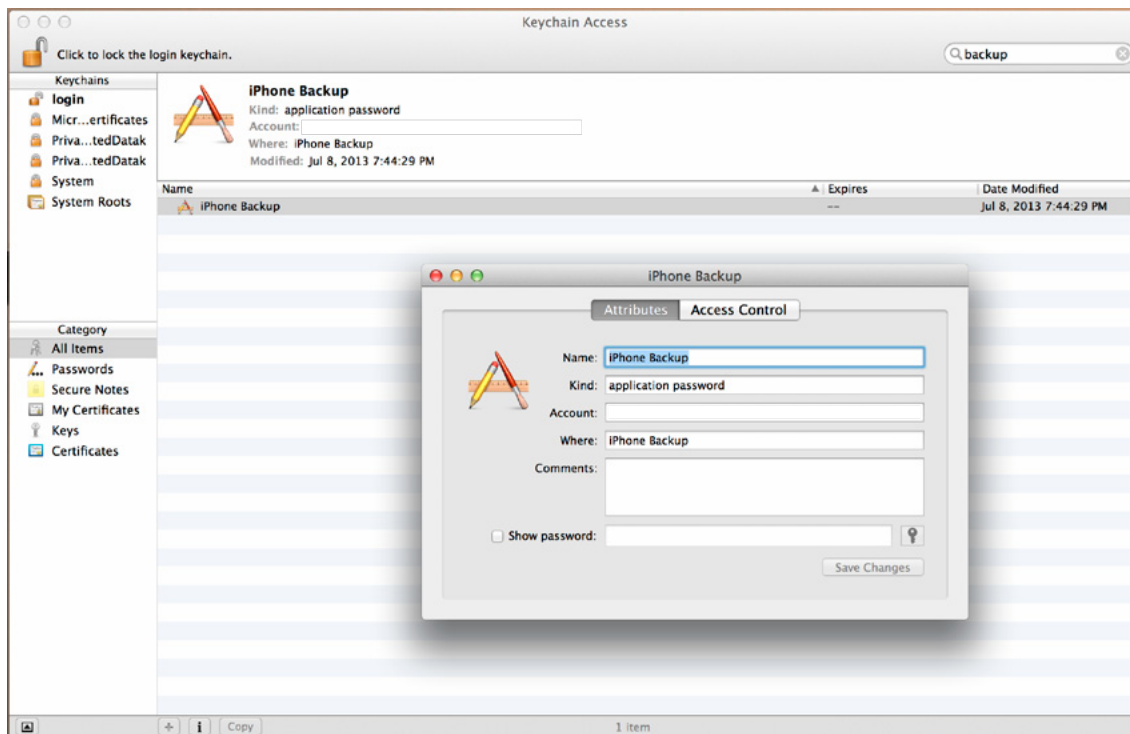


Figure 9. Keychain Access

IPHONE-DATAPROTECTION TOOLS

The Sogeti Labs have made their tools open source and available for performing disk image acquisition and backup extraction for iOS. These tools are invaluable if access to the device is available and very useful for iTunes backups. For encrypted backups we use the commands below to prepare our Apple OS X Mountain Lion System for backup extraction.

PREREQUISITE STEPS

- Install Xcode from the Apple App Store.
- Go to Preferences in Xcode and install command line tools.
- Install MacPorts for your operating system.
- Install OSXFuse.
- Install Mercurial.

STEPS FOR INSTALLING SOGETI TOOLS

Run the following from a terminal using an account with sudo privileges:

```
curl -O http://networkpx.googlecode.com/files/ldid
chmod +x ldid
sudo mv ldid /usr/bin/
#fix if unix tools were not installed with xcode
sudo ln -s /Developer/Platforms/iPhoneOS.platform/Developer/usr/bin/codesign_allocate /usr/bin
```

```
#create symlink to the new xcode folder
sudo ln -s /Applications/Xcode.App/Contents/Developer /
sudo easy_install M2crypto construct progressbar setuptools
sudo ARCHFLAGS='-arch i386 -arch x86_64' easy_install pycrypto
#clone the data protection tools from a directory which your account has write access to
hg clone https://code.google.com/p/iphone-dataprotection/
cd iphone-dataprotection
make -C img3fs/
```

IPHONE BACKUP DECRYPT

Use the following script with your newly available iTunes backup password to decrypt the backup.

```
sudo ln -s python iphone-dataprotection/python_scripts/backup_tool.py /Users/<username>/Library/
Application\ Support/MobileSync/Backup/<encrypted backup folder name> ~/iPhone-Backup-Extracts
```

Note: This script will ask for the password cracked previously in the decryption section.

ITUNES BACKUP HIERARCHY

Once assembled into a file structure, each backup has ten domain folders with an additional one for each installed application. In the figure we show two AppDomain objects entitled “AppDomain-TVcom,” and “AppDomain-com.amazon.Amazon,” both AppBundle titles from the App Publisher in iTunes. As noted elsewhere, encrypted backups have more information than non-encrypted ones, such as wireless passwords and the entire keychain.

Ten Domains Standard + 1 AppDomain for Each Installed Application

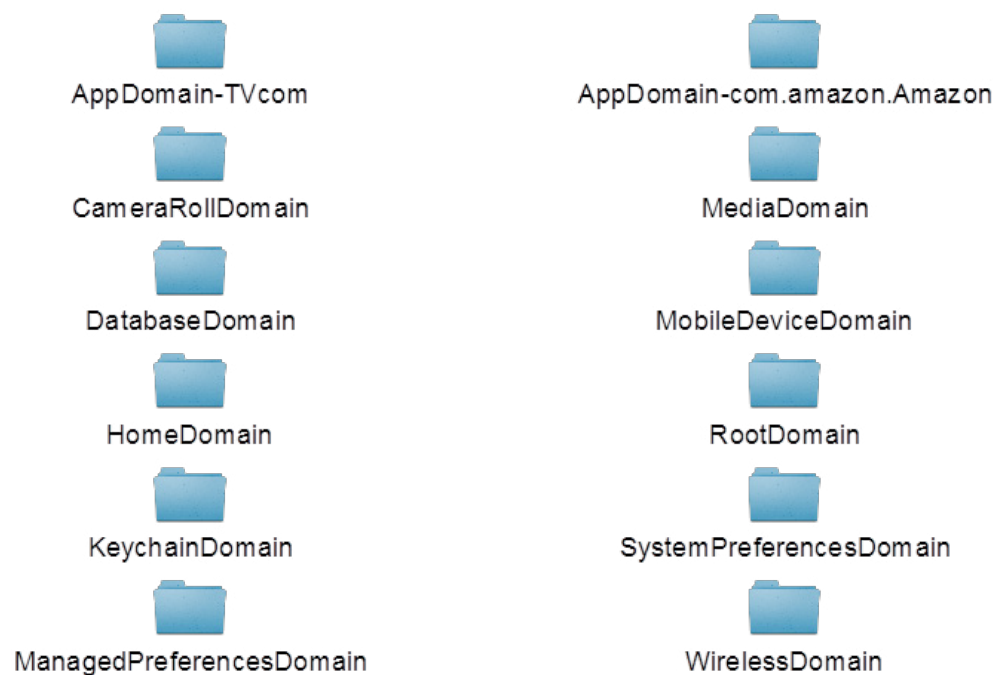


Figure 10. *iTunes Backup Hierarchy Assembled*

SMS SQLITE DATABASE

The SMS Messages from the backup are stored in the sms.db file which falls underneath the HomeDomain. For the purposes of this paper we used SQLiteStudio v.2.1.4 for OS X to browse the data. Attachments exist below the MediaDomain in the backup hierarchy.

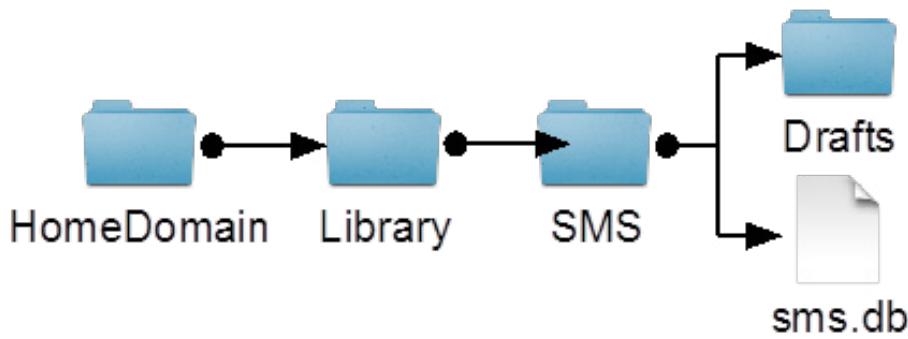


Figure 11. SMS.db Folder

SQLiteStudio (v2.1.4) [Editor 1]

SQL query Results History

Grid view Form view Plain text view

Total rows: 1390

#	ROWID	guid	text	treplcalservice_centerhandle_id	subject	country	attributedBody	version	type	service	account
1	13169216438806A-466F-3D1E	I am going to go!	1	1557			streamtype0110	10	1SMS	le:	ICAG
2	13169318725E35F-0883-4F1E	like the old dozi	1	1557			streamtype0110	10	1SMS	le:	ICAG
3	1316941F3E87363-756C-381E	In the old days 10	1	1557			streamtype0110	10	1SMS	le:	ICAG
4	1316951C7A70584-38DF-431E	No thons. 10	1	1557			streamtype0110	10	1SMS	le:	ICAG
5	13169612AF4D18-38ED-431E	Lo!	10	1557			streamtype0110	10	1SMS	le:	ICAG
6	1318431420940C-08E2-201E	Hi!	10	1557			streamtype0110	10	1SMS	le:	ICAG
7	131844158C0813F-E47C-451E	c u!!!	10	1557			streamtype0110	10	1SMS	le:	ICAG
8	13184518386414B-8973-AB1E	dont c u...not 10	1	1557			streamtype0110	10	1SMS	le:	ICAG
9	1318461C8023A64-4508-451E	Sitting at table 10	1	1557			streamtype0110	10	1SMS	le:	ICAG
10	1318471858631CA-36C3-471E	Lo!	10	1557			streamtype0110	10	1SMS	le:	ICAG
11	1318481120976C5-290C-BA1E	My kid sent me 110	1	1557			streamtype0110	10	1SMS	le:	ICAG
12	131849148E7110C-4A18-481E	Gimme sahee cofi	10	1557			streamtype0110	10	1SMS	le:	ICAG
13	131850150502521-129F-5E1E	There isnt any r10	1	1557			streamtype0110	10	1SMS	le:	ICAG
14	131851153468036-F670-441E	You did ju10	1	1557			streamtype0110	10	1SMS	le:	ICAG
15	13189510F62077F-8991-4F1E	Preparing for thi10	1	1561			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
16	1318961903FF48A-0ED7-461E	Hey Brian, I thi10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
17	131897161EDBF895-0902-4D1E	I know know if yoi10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
18	131911197180135-5A89-481E	Got probab1e the10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
19	1319121CAE5E2C0-8013-451E	Thanks dude!	10	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
20	1319131088510C4-1120-4D1E	ATP41FE!	10	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
21	131914164891628-AD09-451E	Set to never expl10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
22	1319151F421E477-57E5-441E	Will try. 10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
23	131916138A435F5-C9F7-461E	Ok we're ready f10	1	1561			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
24	13191715E98A86-026A-461E	Yay!	10	1561			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
25	132008127AC3D24-01A8-481E	Check out this v10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
26	1321051145CEA75-0767-481E	Dude I got the bi10	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
27	132120108577FC-0A30-4F1E	No	10	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG
28	132121197CBA421-4CD5-411E	http://www.ama210	1	1562			streamtype0110	10	1Message	p:+553030432161AA7	ICAG

1390 row(s) read in 0.008795 second(s).

Figure 12. sms.db Messages Table



Figure 13. SMS Attachments Folder

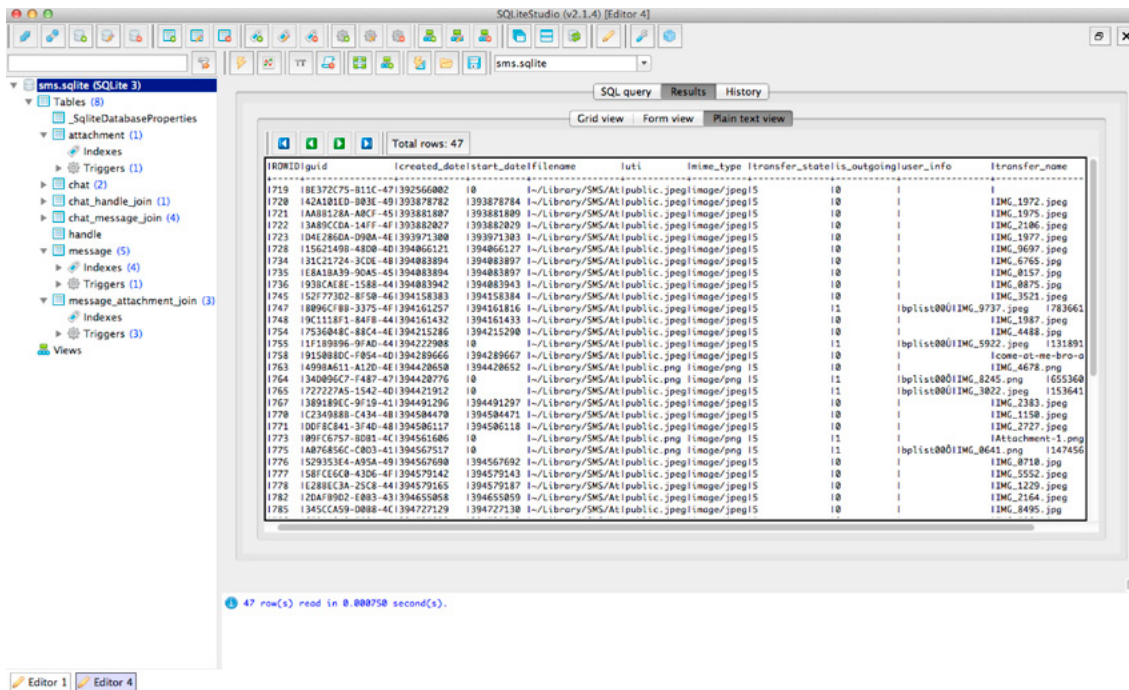


Figure 14. sms.db Attachments Table

COMMERCIAL TOOLS

Forensic investigators must be able to show how any tool works if ever in a court of law. Hence, best practice dictates that he or she be prepared to explain the inner workings such tools. Now that we know how iOS Backups are delineated, we can see a simple tool such as iBackupBot v.4.1.7 for Macintosh shown below. It quickly correlates backed up messages, contacts and attachments.

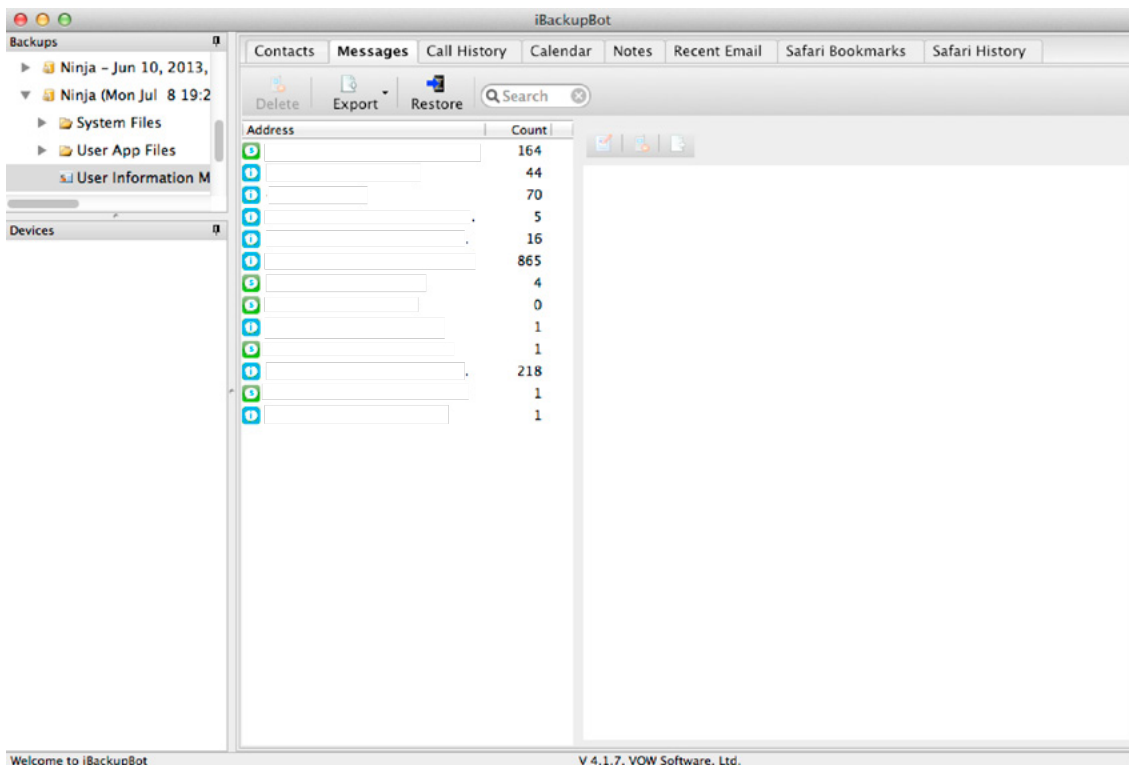


Figure 15. iBackupBot Messages View

CAVEATS

In the scenario up to this point, we describe a relatively straightforward path to cracking, decrypting, and extracting information from iTunes iOS backups. Yet in forensics, rarely do people make it easy to find evidence of unauthorized activity. For example, on OSX, users can delete files using Secure Erase. This finder option allows usage of the trashcan to securely delete files. By default it does a seven-pass erasure. In other words, the operating system then writes seven different passes of random data over the file location. Using the `/usr/bin/srm` binary however, a user can use up to *35 passes*! An administrator can delete the `srm` utility but cannot disable the “Empty Trash Securely” option. Also these options exist for securely erasing disk images.



Figure 16. *Empty Trash Securely*

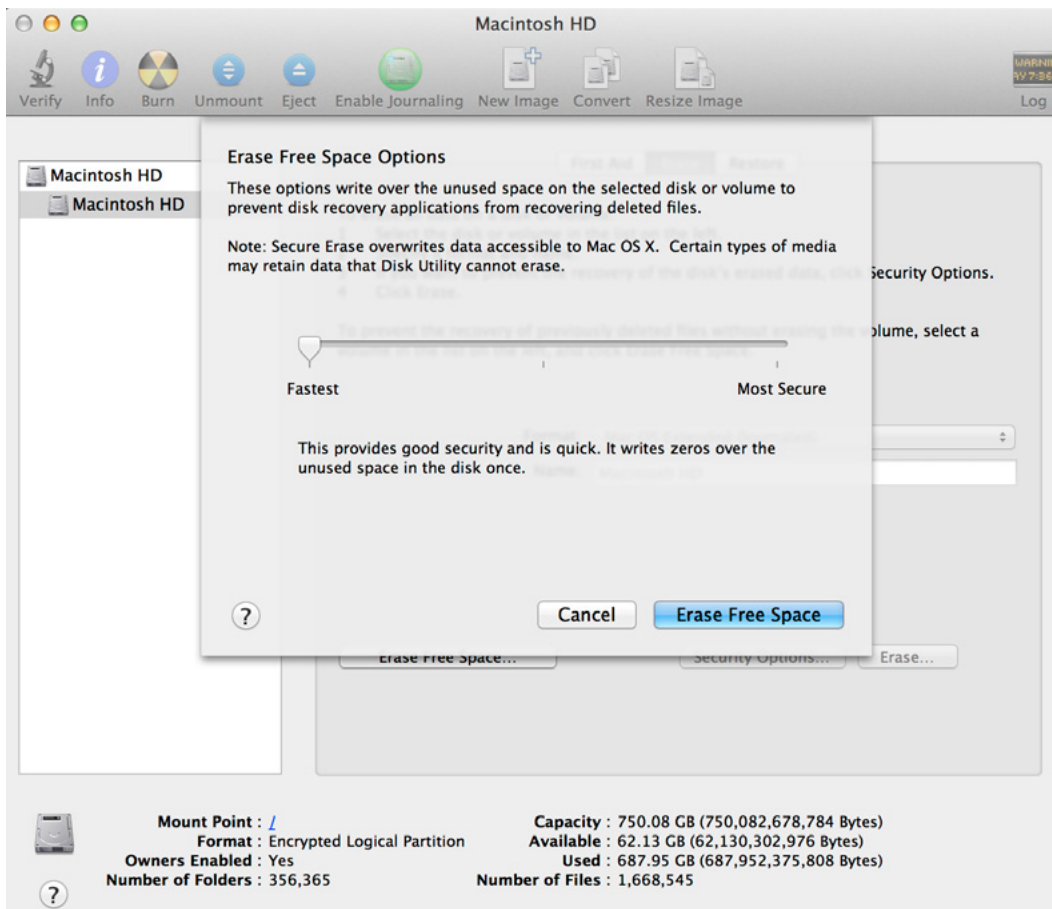


Figure 17. *Erase Disk Options*

RECOVERING ERASED FILES

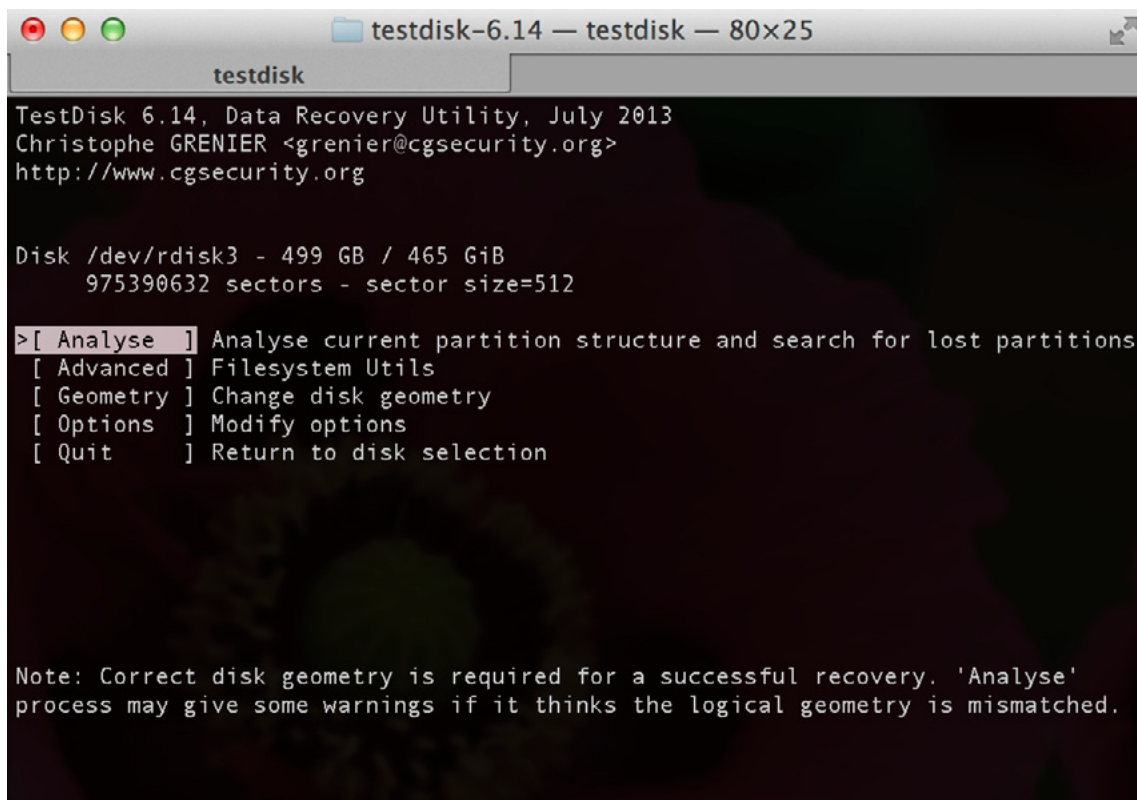
If a target or suspect has deleted files securely it is usually considered *impossible* to recover the data. In fact the 7-pass erase is DOD compliant in the United States. Now, if the file is not *securely* deleted however one can use either open source or closed/commercial tools for data carving.

OPEN SOURCE

TestDisk&PhotoRec 6.14 was released 30 July 2013. Like most expert tools, it has no GUI as of yet. It requires that the device in question is unmounted. In other words, one would have to either use a drive sled, or boot the mac in Target Disk Mode.

TARGET DISK MODE

To boot the OSX system in Target Disk Mode, shutdown the system completely. Connect a Firewire cable from your shutdown Macintosh to your live forensics macintosh. If you have a write blocker, position this between the target and examiners system. Power on the system and hold down the “T” key. You will see a Firewire icon on the target system and the drive will appear in the Disk Utility on the examining system. Once booted, ensure that the disk drive is not mounted. Then utilize the “sudo ./testdisk” to build a disk image.



```
testdisk-6.14 — testdisk — 80x25
testdisk
TestDisk 6.14, Data Recovery Utility, July 2013
Christophe GRENIER <grenier@cgsecurity.org>
http://www.cgsecurity.org

Disk /dev/rdisk3 - 499 GB / 465 GiB
975390632 sectors - sector size=512

>[ Analyse ] Analyse current partition structure and search for lost partitions
[ Advanced ] Filesystem Utils
[ Geometry ] Change disk geometry
[ Options ] Modify options
[ Quit ] Return to disk selection

Note: Correct disk geometry is required for a successful recovery. 'Analyse'
process may give some warnings if it thinks the logical geometry is mismatched.
```

Figure 18. TestDisk&PhotoRec 6.14

COMMERCIAL

Below are screenshots of paid utilities for drive imaging. The first, Disk Drill has numerous options for searching for deleted partitions and building log files of lost files. Secondly Data Recovery also has tools for Lost Partition Recovery and File Recovery. Whichever tools your team uses, *be sure to create an image of the drive to protect your evidence and chain of custody*. Only then is it advisable to perform data carving for your lost backups. Data carving itself is such an analytical process that even experts in the field cannot standardize a single methodology for interpretation. The best way to be prepared for defense challenges is to verify your discoveries with Open Source utilities such as “dd” usually considered the gold standard in imaging.

HOW TO USE ENCRYPTED ITUNES BACKUPS

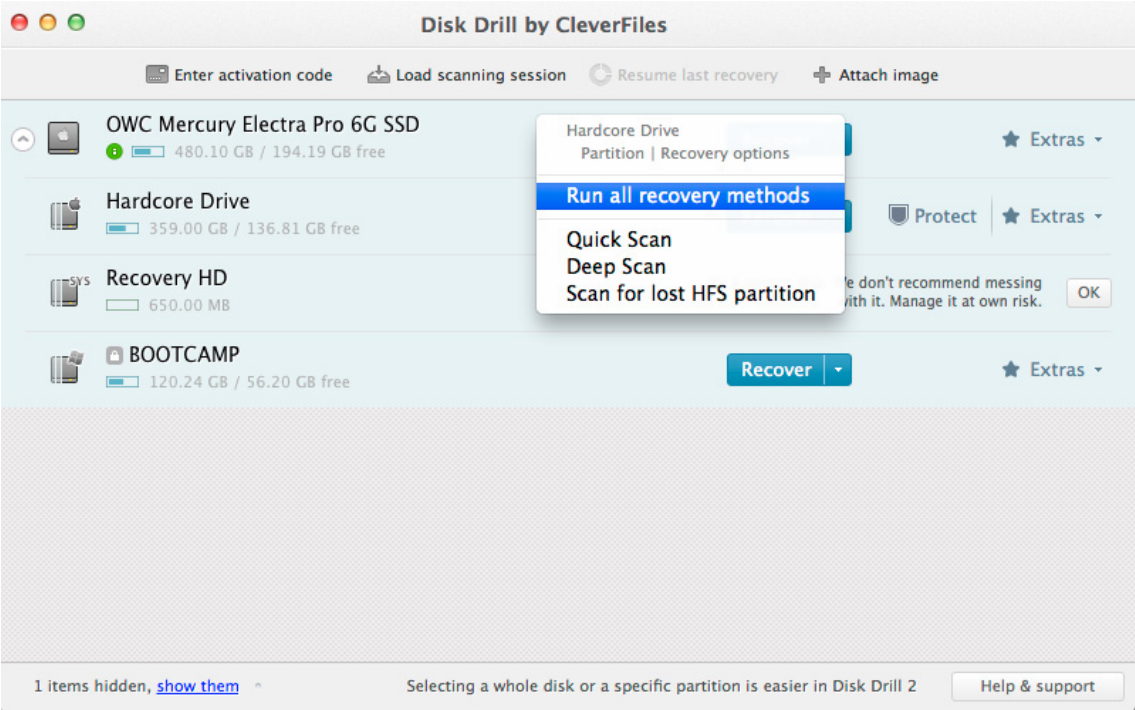


Figure 19. Disk Drill



Figure 20. Data Recovery

SUMMARY

This article shows a forensics investigator how to use Apple OS X for iOS forensics. First it shows what iTunes backs up, to where. Then it shows encrypted versus unencrypted backups. Next it models how to decrypt backups by using standard password cracking utilities and open source iPhone decryption tools. After this, it describes the hierarchy of the Backup as well as showing how to view it with freely available applications. Finally it offers commercial tools which do these things quite quickly. Remember though, even though commercial tools are often available to make our job easier, always start with a solid foundation before building your case. For whatever your case may be, defense challenges often attack the opacity of closed source commercial tools as well as an investigators ignorance of the actual process being employed. Remember the caveats above and how to find data in what may appear to be unrelated bits.

BIBLIOGRAPHY

- Zdziarski, Johnathan A. iPhone Forensics O'Reilly Media ISBN 0596153589
- Bedrune, Jean-Baptiste and Sigwald, Jean iPhone Data Protection in Depth, <http://conference.hitb.org/hitbsecconf2011ams/materials/>
- Satish, B iPhone Forensics <http://resources.infosecinstitute.com/iphone-forensics/>

ON THE WEB

- <http://support.apple.com/kb/ht4946> About iOS Backups
- <http://www.davegrohl.org> dave grohl password utility
- <http://www.macports.org/install.php> Installing MacPorts
- <http://mercurial.selenic.com> Mercurial SCM
- <http://osxfuse.github.io> OSX Fuse
- <http://code.google.com/p/iphone-dataprotection> Sogeti iphone-dataprotection
- <http://sqlitestudio.pl> SQLiteStudio
- <http://www.elcomsoft.com/eppb.html> ElcomSoft EPPB
- <http://www.icopybot.com/download.htm> iCopyBot Download
- <http://www.recoverlostpassword.com/products/itunes-backup-password-recovery.html> iTunes Backup Password Recovery

ABOUT THE AUTHOR



Gouthum Karadi works for Virtual Nexus, LLC. in California, USA. He has been working on computers since the first Tandy Color Computer 16K and 128K Macintosh. His life has taken him through UC Berkeley, the US Army, Microsoft, and a host of other interesting places including flying a private plane, surfing, and triathlons. Some of his certifications include Apple, Microsoft, Cisco, CISSP, and CEH.



Dr.WEB®

since 1992



Dr.Web 9.0

for Windows — the rapid response anti-virus

1. Reliable protection against the threats of tomorrow
2. Reliable protection against data loss
3. Secure communication, data transfer and Internet search



© Doctor Web
2003 — 2013

www.drweb.com

Free 30-day trial: <https://download.drweb.com>

New features in Dr.Web 9.0 for Windows: <http://products.drweb.com/9>

FREE bonus — Dr.Web Mobile Security:
<https://download.drweb.com/android>



HOW TO PERFORM A BASIC AND FAST FORENSIC ANALYSIS

ON MACINTOSH OPERATING SYSTEMS

– A QUICK START GUIDE

by **Deivison Pinheiro Franco**

Computer Forensics is an area that is very Windows-centric. Many tools pay lip service to Apple's Macintosh (Mac) platform, and others do not even recognize it at all. The few Mac tools available are either expensive or inadequate. Regardless, it is necessary for an investigator to know what to look for and where to look. This article is intended to give investigators a brief outline of what the file system and structure of a Mac looks like and to give a basic criteria on what to look for, as well as some generalized locations for where to look. It is far from a comprehensive forensic manual for Macintosh computers, but it does attempt to give an examiner relatively comfortable with Windows environments a place to start learning about Mac forensics.

What you will learn:

- A brief outline of what the file system and structure of a Mac looks like;
- Basic criteria on what to look for evidences in Mac Operating Systems;
- Basic Criteria and some generalized locations for where to look for evidences in Mac Operating Systems;
- The Mac Operating Systems Structure;
- Basic differences between Windows and Mac File Systems.

What you should know:

- A basic understanding of Mac Operating System Operation;
- A basic understanding of Mac Operating System Structure;
- A basic understanding of Mac Operating System Use.

Whether or not the PC crowd is ready, Apple is clawing its way back up the ladder in market share. The success of the iPod has made Apple a household name once again, and with the frustrations of Windows Vista now more than ever people are flocking towards the inviting and seemingly more stable environment Apple's Mac Operating System (Mac OS) offers. This presents a significant hurdle for forensic investigators because many are not familiar with Mac OS or UNIX systems in general. The law of numbers indicates that as Macintosh systems re-saturate the personal computer market space, their use in illicit

or illegal activities will increase as well, necessitating the need for investigators to become cross-platform in their abilities. This platform agnosticism will not come easy, and it will be difficult for investigators to be “experts” at both systems, simply because the rapid evolution of both platforms and the amount of time it takes to become fully comfortable with both sides.

With this in mind, this article was written to give investigators already familiar with at least basic Windows forensics, a “Quick Start” guide to the Mac platform and its internals. It begins with a logical look at the file structure, and then moves into a summary of the more common places investigators need to examine. As is resonated throughout this document, it is not meant as a comprehensive guide, since the scope of such a guide would be outside the space constraints of this article.

THE MAC OS FILE STRUCTURE

Many forensic investigators are familiar with the hierarchical file structure in Windows which typically begins with `C:\` or some other drive letter. The Mac follows a similar hierarchical structure, but with a different approach. This section addresses the “trees” of the Mac OS file structure, the purpose of each, and how it relates to a Windows structure.

THE UNIX ROOT

The fact that Mac OS is a UNIX system is important to investigators because many standard UNIX conventions are followed at a level most Mac users never venture to, and can be leveraged to an investigator’s advantage. Access at the UNIX level is often more rewarding forensically than access from most graphical tools. UNIX compliance also means that many of the open source tools UNIX systems enjoy are available for forensic investigators working with Mac hardware and software. At the same time, such a basis presents many more challenges, especially given the lack of experience many investigators have with Mac hardware and software.

As an example, many system files are marked Invisible in the Finder by a very neat metadata trick Apple built into their system so that normal Finder users never see them. A moderately technical user can set other files to be invisible very easily, using the SetFile utility Apple provides to developers. This process hides files or directories from the prying eyes of a casual user, but not from a typical UNIX user. Anybody with reason enough to hide something would be able to find instructions on how to do this easily through the internet. Figure 1 shows what a typical user would see in Finder compared with what is truly there at the UNIX level.

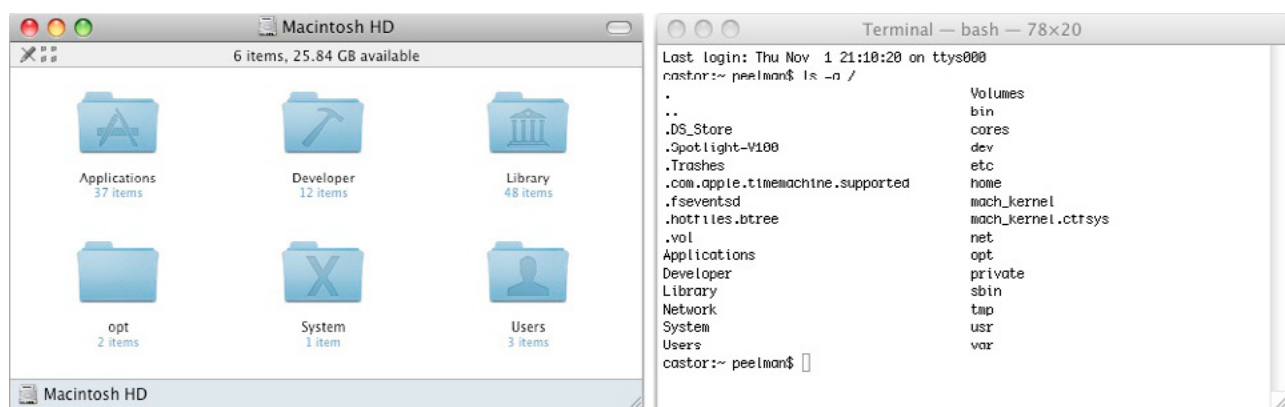


Figure 1. A comparison of what Finder users see and what is really there at a UNIX level

Notice in Figure 1, we only see 6 folders when using the Finder, with no files visible at the root level. However, when we look at the system at the UNIX level using Terminal, we see 28 files and folders. Something that is typically irrelevant but may bear weight in certain cases is that `/etc/`, `/tmp/`, and `/var/` all actually reside in `/private/`, but are symbolically linked to their standard UNIX location at the root of the file system.

This UNIX tree is very important and can reveal many key aspects of the system, however examining Mac OS purely from a UNIX Standpoint would be the topic of a separate, probably larger, paper, since any user who can truly manipulate an Mac OS system at this level will probably present more challenges

to investigators than a normal user. With that in mind, there are a few key points to know about the Mac's file system. Beneath the fancy hardware and flash interface, Macs use a traditional UNIX file system, with / being the root directory containing all other directories and files.

Any external file systems (USB/Firewire hard drives, Flash memory devices, digital cameras, anything that can be seen as a storage device) are mounted under /Volumes/ with a pseudonym name for each device (two devices can have the same name, for example, you can have two hard drives named "Macintosh HD". When the system encounters this situation, it would simply append a "1" to one of the drives when mounting). You can think of /Volumes as the My Computer of the Mac OS file system. Where on Windows you would see a C:\, D:\, and E:\ drives in My Computer, on a Mac if you navigate to /Volumes/ in the Finder you may see MyUSBKey, WDBook, etc. Logically from an access standpoint, the UNIX root is divided into several trees, each tree representing a domain of user access. Each of the four domains is addressed and detailed below.

THE SYSTEM TREE

At the UNIX root of the Mac OS file tree is the System Folder. This folder is owned by the system and is commonly not writable by the user without providing administrative credentials, meaning that modifications to this part of the system are typically deliberate and often worth investigating. The files and folders in this tree are restricted to the core files of the operating system. Anything that is necessary for the basic functionality of booting and presenting a basic GUI should be in this folder, including but not limited to: Applications such as Finder and Dock, the default user profile for creating new users, Preference Panes, Frameworks, mostly things that should never be touched by a user manually. Typically everything in this folder only change during software updates such as security packs or OS updates.

THE LOCAL TREE

The Local Tree represents files and folders that are not entirely necessary for the OS to function, but are available to all users of the machine and are generally important in day to day use of the computer. This tree includes the /Applications/ and /Library/ folders. If it is installed on the system, the /Developer/ folder would be included in this tree as well. Files and Folders in this tree are modifiable only by a user in an administrative group and those users must authenticate when doing anything that changes these files. Items contained here include not just applications, but support files, and preferences that are available for use by every user on the system. Unlike the System tree, this tree may in fact be modified quite often by a user, either when installing software, setting a global preference, or creating folders or files in a public space for Documents or other media that needs to be shared with the other users of the system. In the cases of installing applications or setting preferences, the user may not even realize the changes are being made at this level, even if they are asked for their user name and password when doing it.

THE USER TREE

The User Tree is generally rooted in a user's profile and is referred to synonymously with the "home" folder. It is sometimes referenced within the file system as a tilde ("~"), so ~/Documents/ refers to the documents folder of the user in question. For example, if joe is the short name of the current user, then ~ would actually point to /Users/joe/. It is merely a shortcut, and is used here within as short hand to describe locations based on a user's home directory.

Since Mac OS was built from the ground up as a multi-user UNIX system, each account a user has on a system can include its own applications, settings, documents, media, and temporary files; basically every user lives in their own little world and rarely needs to write to disk space outside of their profile, but each user still has read and execute access to the applications and capabilities of almost everything in the Local and System trees.

On this same note, a user's home directory is very portable and easily moved between systems. It is commonplace in the Mac universe for a user to move their entire profile when upgrading computers, and Apple actually provides a Migration Assistant that does just that (among other tasks such as migrating the local user database, etc.). This may become relevant if references are found in a user's home directory to items (applications, documents, etc.) that do not exist on the machine being analyzed. A suspect may have moved their home directory to an otherwise clean system and somehow disposed of the dirty machine. In some contexts it may be more important for a suspect to have a computer that appears clean rather than have no computer at all, or a hard drive that has recently been wiped and now contains a fresh OS install.

By default, each home directory on a Mac OS system does not include an Applications directory, but creation of one is recognized by the OS and it is given the same icon the more prominent `/Applications/` directory receives. Even on a tightly managed Mac OS system, users with only standard user accounts and no administrative privileges can download, install, and execute many types of software.

The User's Tree is owned by that current user, and that user has full control over everything in it. Other normal users typically cannot even see the contents of a user's home folder. Administrative users can, but typically only from the Terminal, and only after assuming a root shell, or issuing commands using `sudo`, and both of these methods require authentication.

THE NETWORK TREE

There is a fourth, lesser known and less often used tree is associated with a networked, highly organized, highly structured environment such as an office or a school. This Network Tree is more scattered than any of the other branches, but most of the applicable files will reside in `/Library/Managed Preferences/`. The folder `/Network` has nothing to do with this tree, it is simply a folder used to logically and dynamically manage network file shares and connectivity. The purpose of the Network tree is similar to a Windows Active Directory environment, where certain machine and user preferences are managed from a centralized location for various reasons. When seizing computers that are part of a larger network, it is a good idea to at least be aware of these settings and verify them with the systems or network administrator.

WHAT IS IT WITH ALL THESE TREES?

Each tree represents a logical domain of access to the system. In this way a system can be managed, yet still offer the user(s) quite a bit of freedom. For example, applications that follow Apple's development guidelines may search `/Library/Preferences/` for a relevant set of preference files before looking inside the User's own Preferences folder (`~/Library/Preferences/`) for a similar set. This allows a set of "local" or "machine" preferences that apply to every user of the machine, but still allows the user to keep preferences for his or her self. It should be noted that not all Applications follow those guidelines.

Preference Panes are a perfect illustration of this hierarchy and how it is beneficial. To manage system preferences, Apple wrote an application called "System Preferences", so you can think of System Preferences as the Windows Control Panel, but done right. System Preferences uses small bundles (specially structured folders) called Preference Panes to divide up its tasks. Many key preferences panes that are used to modify and manage important system settings (display resolutions, network settings, etc.) are necessary for the system, so they go in the System Tree (`/System/Library/Preference Panes/`). Later on, a user wants to add some functionality to their system, such as the Growl notification system. If they want the functionality to be available to every user on the system, they would add it to the Local Tree (specifically in `/Library/Preference Panes/`, notice the pattern yet?).

However, if they only want to add the functionality for themselves, and not share with everybody else, they can simply add it to the User Tree (`~/Library/Preference Panes/`, note the tilde). System Preferences, being intelligent and properly coded, looks in all three domains for preference panes to load when it starts up.

MAC OS AND WINDOWS FILE SYSTEMS COMPARISON

The System tree (`/System, /bin, /usr, /private`) is loosely analogous to the `C:\Windows\` or `C:\WINNT\` directory in a Windows environment. The Local Tree (`/Applications, /Library, /Developer`) is similarly related to the `C:\Program Files\` directory. Typically in a Windows environment, `C:\Program Files\` contains both Applications themselves and any associated support files (templates, example files, plugins, etc.). In the Mac world, Applications are contained in the `/Applications/` directory, and any support files necessary would be contained in the `/Library` folder, generally under `/Library/Application Support/`. Similar to the preferences example in the previous paragraph, many applications can look to both `/Library/` and the current User's Library Folder (`~/Library/`) for support files. The User tree (`/Users/`) is similar to the `C:\Documents and Settings\` folder, with a separate home folder for each user on the system.

WHERE TO LOOK

Forensic analysis of a Macintosh system is conceptually no different than on a Windows system. The same principles of isolation, acquisition, imaging, analysis, and reporting apply, but the procedures and context differ for the imaging and analysis stages. In some cases these differences are severe, in others they are subtle. This section is meant to give a brief primer into where to look during an analysis to find

the common files that may be relevant to an investigation. It is not a comprehensive guide, and therefore should be used as a learning tool, but not an investigative check list.

BROWSER INFORMATION

Web browsing is a daily pastime now, and like every other task, people always have a favorite tool. On the Mac side, there are many, many web browsers. From Apple's own Safari, to the ubiquitous Firefox, to the less known but still popular Camino, OmniWeb and Opera. Sticking with a "general user" theory, Safari and Firefox are the only two outlined below. Both programs download files to the Desktop (`~/Desktop/`) by default but can be configured to place them anywhere. The preference files for both programs are located in `~/Library/`.

SAFARI

Apple's native Safari browser is rapidly growing in market share as more and more users turn to Macs. It uses WebKit, an open source framework Apple develops and uses throughout its operating system. Safari spreads its forensically-useful files out over a few directories. Bookmarks, stored form values, browser history, information about the last session, and an icons file, are all contained in `~/Library/Safari/`. Cached internet files are stored in a series of folders in `~/Library/Caches/Safari/` and/or `~/Library/Caches/com.apple.safari/` depending on the version of Mac OS and Safari being used.

From Safari 3 Apple added a Private Browsing feature that reduces the amount of cached files to almost zero. There is also a Reset Safari feature that can blow away cache files, history files, etc. Both of these features create an issue for forensic recovery, but there is no way around it known at this time, so not much more information can be detailed.

FIREFOX

Many Windows investigators may already be familiar with Firefox profile system, and for all intents and purposes it is the same on Macs. The core files of a user's profile, including bookmarks, history, and any installed themes or extensions can be found in `~/Library/Application Support/Firefox/`. Cache files for a user can be found in `~/Library/Caches/Firefox/`. Each of these will contain a Profiles directory, each Profiles directory will contain one or more folders (profiles) with a unique folder name. Each folder name in the Caches directory will match up to one in the Application Support directory, so if you have multiple profiles you can use this to match the cache files with the proper profile.

E-MAIL

E-mail in general is forensic nightmare, with its lip service to security, lack of a trustable delivery trail, its wide open format that practically screams "edit me!", and the large variety of clients available. While Macs don't make email investigations much easier, they do not make it harder either. Mac users will often use one of three e-mail clients: web-based, Apple Mail, Microsoft Entourage. Please note that like web browsers, many other email clients exist for Macs, but that for the vast majority, one of these three will apply.

WEB-BASED E-MAIL

The first and most irrelevant to this section is web-based e-mail. You may be able to pull passwords from the Keychain (discussed later), get the addresses of what services were used from the browser history or the Keychain, and maybe recover any attachments that were download, but there is little to no way to get the actual messages read or sent, from the local machine. The service provider will need to be contacted and worked with to acquire the relevant data.

APPLE MAIL

Apple's native email client is true to the mantra of Mac OS, full-featured, flexible, powerful, yet remaining painfully simple. This is even true from a forensic perspective. All of a user's messages are stored inside `~/Library/Mail/`. There will be a sub directory for every account a user has setup inside the client, as well as a Mailboxes folder for any local mailboxes a user has created that are not associated with any one particular account. An example using an IMAP account would be: `~/Library/Mail/IMAPusername@mydomain.com/`. On newer systems, each message is stored as plain text with full headers in its own file with an extension of "emlx" inside of a folder with a .mbox extension.

Older systems stored everything in single text file using a standard UNIX convention called "mbox", which is a consolidated mailbox format. This file was generally a plain text document with all the messages

(with full headers) in a mailbox appended together, and was actually forensically easier to analyze, especially when compared to a similar situation such as Microsoft Outlook's PST format. The newer method has its own set of "pros" but makes forensic analysis of a user's email cache more difficult.

Also stored in `~/Library/Mail/` are several important plist (Property List) files containing information about Smart Mailboxes, message rules, and of particular forensic importance: opened attachments. On the topic of attachments, IMAP and Exchange accounts store attachments within the account folder in `~/Library/Mail/`. These types of accounts loosely let the messages to the attachments by placing each attachment in a subfolder named after the message ID that it goes with. POP based email accounts generally store attachments in `~/Library/Mail Downloads/`. The Mail.app client handles keeping track of which attachments go with which message for a POP account.

MICROSOFT ENTOURAGE

Microsoft Entourage is generally referred to by the Mac community as Outlook for Macs. It has the same basic feature set as Outlook, though with a very different interface, and is the Mac's only way to fully interface with Exchange environments. That last fact is relevant because Macs that exist inside many corporate environments that rely on Exchange will most likely have Entourage on them. Entourage stores all of its data inside the user's Documents folder (`~/Documents/Microsoft User Data/`). The contents of this folder can vary depending on how much a user utilizes the Office suite, but there will be two folders that should be examined in depth: `~/Documents/Microsoft User Data` and `~/Documents/Microsoft User Data/Saved Attachments/`.

The relevance of Saved Attachments should be pretty obvious. The Office Identities folder will contain subfolders for each identity a user has defined, but in most cases there will only be two things in this folder, a folder called "Main Identity" and a file called Newsgroups Cache. Inside a user's Main Identity folder will be a Database file which contains all of the user's messages in a compressed or otherwise scrambled manner. Other files in here include Rules, which is used to store defined message-handling rules, Mailing Lists, which is used to manage Mailing List rules (these differ from normal Mail rules), and Signatures, which keeps a listing of the user's signatures for signing messages. All three of these files are compressed or otherwise scrambled similarly to the Database file. This entire directory is theoretically portable and could potentially be read (through a write blocker) by an examination machine if placed or linked to the correct location on the examiner's Mac.

KEYCHAINS

Apple designed an ingenious method of storing small amounts of data that need to be secured in partly encrypted files called keychains. These files store user's passwords, certificates, and any secure notes (plain text only), in a partially encrypted and secure file inside the `~/Library/Keychains/` folder. The keychain is typically named `login.keychain` and can be moved from system to system, allowing a user to have the same saved credentials available to them on multiple systems. Forensically this little file can often yield a lot of data about a user's habits, since many users, except those with the most security conscious agendas, make use of the convenience of the keychain's abilities. If the user chooses convenience over security, keychains can store credentials for several key areas:

- Airport (802.11 wireless networks);
- Instant Message Account Passwords;
- VPN Passwords;
- Encrypted disk image passwords;
- Application passwords for anything that may require authentication;
- Website passwords (may include webmail passwords);
- Secure Notes;
- Digital Certificates for secure websites;
- Digital Certificates for Email Signing & verification.

Obviously this is one file that is very important for investigators to find and analyze. The one caveat to this is that the file is partially encrypted and can only be accessed with the user's login password, or if the user is particularly security conscious, they will use separate passwords for their login and their keychain. This makes discovering the actual passwords difficult without suspect cooperation, however simply viewing the file in plain text will show that not all of the data is encrypted, only the passwords themselves. Information such as web addresses, email addresses, services, and other

items may be plainly visible. One final caveat is that users are not limited to just one keychain file. They can create as many as they want, each with different data stored in it, and each can have its own password.

SYSTEM LOGS

The system logs on a Macintosh system may provide insight into a user's actions and the timestamps may help present a stable timeline of events. Apple created several locations for log files: `/var/log/`, `/Library/Logs/` and `~/Library/Logs/`.

The relevance of these logs depends on the context of the case, so suffice it to say that some may be more important than others, and some may be completely irrelevant. The most commonly useful ones are outlined below.

SYSTEM.LOG

System.log is the catch all log for the system. Many things get logged here, but the content of this log varies system to system, and even minute to minute, depending on the context in which the machine is used. The system will typically compress and archive these logs on a regular basis, so in addition to `/var/log/system.log`, there should be several different versions of this log with a number and ".bz2" appended to them. This will be true for many of the logs the system keeps.

SECURE.LOG

Secure keeps track of any changes to secure system areas. That means that any time a user has to authenticate to modify a normally off-limits part of the file system, it is logged here. Any time a user modifies a setting in System Preferences that affects more than just their user preferences, it gets logged here. Any time a terminal-savvy user authenticates to a higher level account to do something, it gets logged here. This includes using the 'sudo' command, and in that case the entire command is logged here. If it is not obvious by now, this file may be very relevant if the goal is to prove that somebody modified or deleted data, or otherwise issued commands that required authentication from the system.

DAILY.OUT

Daily.out is the log of the daily event that all Macintosh machines execute by default, once a day, to do housekeeping tasks such as rotate logs and purge temporary files. The reason it is important is that it presents two key pieces of data that are forensically important, it records the results of an "uptime" command, reporting how long the system has been up, and it reports all mounted disks, network, local or otherwise. This provides a very convenient place for investigators to look to see what disks were mounted recently. The only bad part is that this script only runs once every 24 hours (the precise time of execution daily varies from machine to machine, and can be determined by view the time stamps inside the log), so the disks will only be recorded if they are plugged in at the right time of the day.

PPP.LOG

This log file provides a time stamped connection history for any VPN connections made from the system using the built in VPN software. Third party VPN software may use a different logging mechanism, see the program's documentation for ideas on where to look.

CONCLUSIONS

Macs really are a world apart from their Windows-running counterparts, but I hope I have outlined the similarities in a way that takes some of the fear out of having to look at one forensically. As I said in the introduction, many of the same forensic principles, SOPs, logic, and instinct can be carried over from Windows forensics to the Mac side, and even the approach taken can be very similar, it's just the format of the data being analyzed will be slightly different.

Unfortunately, many things were cut from this article for the sake of space, since this was originally intended to cover a broader scope. Some of the things left out includes chat/Instant Message logging, encrypted disk images including FileVault, secure virtual memory and its impact, locations of applications, support files, preferences, etc. The prominence of XML configuration files for almost every facet of the system and which ones may hold the key to a mystery. On the investigative side, there are disk and partition operations in Disk Utility, some tips for live analysis of a system, the new rules that Mac OS makes with features like Time Machine, a walkthrough of the Mac boot process, and a basic guide to using Terminal.app, which would have evolved into an introduction to using the UNIX side of the operating system from the command line. There is a lot of data available on any of these topics, but not

much of it has been digested into a forensically useful format. Also, as was noted earlier, the platform is evolving rapidly, with a new major release from Apple every 18-24 months. This makes getting and staying current on Macs that much more difficult.

REFERENCES

- API Reference: Mac OS Manual Pages. (n.d.). Retrieved Dec. 5, 2007, from <http://developer.apple.com/documentation/Darwin/Reference/ManPages/index.html>.
- Chang-Tsun Li. (2009). Handbook of Research on Computational Forensics, Digital Crime and Investigation. New York: Information Science Reference.
- Craiger, P., Burke, P., Olivier, M., & Sheno, S. (2006). Advances in Digital Forensics II (IFIP International Federation for Information Processing). New York: Springer.
- developer.apple.com/documentation/Security/Conceptual/keychainServConcepts/index.html.
- [documentation/MacOSX/Conceptual/BPFileSystem/index.html](http://developer.apple.com/documentation/MacOSX/Conceptual/BPFileSystem/index.html).
- Donnelly, D. (n.d.). Mac Forensics – BlackBag Technologies. Retrieved Dec. 5, 2007, from <http://www.macos.utah.edu/documentation/security/forensics.html>.
- Eoghan Casey. (2010). Handbook of Computer Crime Investigation Investigation. California: Elsevier.
- Eoghan Casey. (2010). Handbook of Digital Forensics and Investigation. California: Elsevier.
- File System Overview. (2006, June 28). Retrieved Dec. 5, 2007, from <http://developer.apple.com/>
- Keychain Services Programming Guide. (2007, January 8). Retrieved Dec. 5, 2007, f
- Knaster, S. (2005). Hacking Mac OS X Tiger: Serious Hacks, Mods and Customizations. New York, NY: Wiley.
- Lamb, D., Miquelon-Weismann, M., Moreau, D., & Orton, I. (2006). Cybercrime: The Investigation,
- McCormack, D., & Trent, M. (2005). Beginning Mac OS X Programming. Indianapolis: Wrox.
- Open Source – Internet & Web – WebKit. (n.d.). Retrieved Dec. 5, 2007, from <http://developer.apple.com/opensource/internet/webkit.html>.
- Technical Note TN1150: HFS Plus Volume Format. (2004, March 5). Retrieved Dec. 5, 2007, from <http://developer.apple.com/technotes/tn/tn1150.html>.

ABOUT THE AUTHOR



Deivison Pinheiro Franco is Graduated in Data Processing. Specialist in Computer Networks, in Computer Networks Support and in Forensic Sciences (Emphasis in Forensic Computing). Security Analyst of Bank of Amazônia. Professor at various colleges and universities of disciplines like: Computer Forensics, Information Security, Systems Audit, Computer Networks, Computer Architecture and Operating Systems. Computer Forensic Expert, IT Auditor and Pentester with the following certifications: CEH – Certified Ethical Hacker, CHFI – Certified Hacking Forensic Investigator, DSEH – Data Security Ethical Hacker, DSFE – Data Security Forensics Examiner, DSO – Data Security Officer and ISO/IEC 27002 Foundation.

HOW TO STEAL GMAIL CREDENTIALS

USING SE-TOOLKIT – A CASE STUDY IN SOCIAL ENGINEERING A STEP-BY-STEP GUIDE USING KALI LINUX SOCIAL ENGINEERING TOOLKIT (SET)

by Kevin M. Moker

Hacking? Why hack when you can trick someone more easily than trying to hack into his or her computer? I am talking about social engineering (SE). SE, in the context of information security, is the ability to manipulate someone to steal certain information. Using SE you can steal credit card numbers, or better yet steal someone's login credentials. With no hacking involved, you will be able to easily reroute payroll funds from an employee's account to another account before they even know the money is gone.

What you will learn:

- The ease of hijacking a victim's account
- How to use Social Engineering Toolkit

What you should know:

- Kali Linux
- VMWare
- Command line

However, with the right knowledge, a victim could thwart an adverse attack. Non-technical individuals should learn how to protect themselves when online. Non-techies should understand what SE is and how to protect themselves.

INTRODUCTION

Social Engineering, from an information security perspective, is the art of manipulating victims to acquire sensitive information. For example, tricking someone to go to a phony website that looks like Facebook or Gmail and making them put in their login ID and password. The phony website and the social engineer practitioner will steal the login ID and clear text password, and begin masquerading as the victim. This article will illustrate an attack whereby a corporate victim is tricked into clicking a link, logging into the site, and releasing their personal information.

WHAT ARE YOU TRYING TO PROTECT AND WHY?

The goal of this article is not to show you how easy it is to hijack someone's credentials in a corporate environment, but to show you how to protect yourself from being duped into giving out your personal information. I will walk you through the steps illustrating the ease at which I can get your credentials and point out where you should be aware. It should be prefaced that I am not a

coder and nowhere near as technical as I should be. That should make you more nervous because just about anyone can pull off this trick, as long as the perpetrator does not get greedy. Greed normally blows covers of perps.

WHAT ARE THE STEPS YOU NEED TO PULL OFF THIS ATTACK?

The following six steps are the steps I use to maliciously acquire a victim's login credentials:

- Have administrative rights to your local computer
- Have an understanding of VMWare and Linux
- VMWare is installed
- Kali Linux image created
- Execute the Social-Engineer Toolkit (se-toolkit)
- Hijack a users PC that is not locked
- Tell the user to go to their Gmail account and see what I've sent

Give a background and overview of the problem you're trying to solve. Then walkthrough each step.

STEP 1: HAVE ADMINISTRATIVE RIGHTS TO YOUR LOCAL COMPUTER

A lot of times, many corporations will allow their employees to have local administrative rights to their computers. That means the employee can do just about anything on that computer, unless the company has some kind of confirmation management software installed. However, with administrative access you can turn the corporate controls off.

So now we have a computer with administrative rights to the computer. The next thing is to install the necessary software. We will first start with VMWare.

STEP 2: HAVE A BASIC UNDERSTANDING OF VMWARE AND LINUX

For this attack to be successful, you will need a basic understanding of VMWare and Linux. Without these two pieces you will have a hard time understanding how this attack works.

What is VMWare? VMWare is a virtualization software that allows you to run multiple operating system on one computer. In other words, you can runs Windows and Ubuntu Linux on the same machine. This is great when you want to run and test local attacks without harming any other system on the network.

STEP 3: VMWARE IS INSTALLED

The next step is to download and install VMWare. You'll have to purchase a license to use VMWare. You could also do this with VirtualBox, but my software of choice is VMWare for ease ability and compatibility.

What is VirtualBox? VirtualBox is very similar to VMWare, but VirtualBox is opensource. You can download it for free without a license.

STEP 4: KALI LINUX IMAGE CREATED

After you have installed VMWare you will download Kali Linux from www.kali.org. I suggest using the 32 bit iso version of Kali. I have a mac where I am running VMWare Fusion (Mac's version of VMWare) and the 32 bit version runs faster. You will have to figure out how to install the .iso into VMWare. What is Kali-Linux? Kali-Linux is previously known as BackTrack, which is a Debian Linux distribution used for digital forensics and penetration testing. The best way to start with Kali is to use the Live Build ISO. You'll be able to test the tools without installing the operating system into your virtual environment.

STEP 5: EXECUTE THE SOCIAL-ENGINEER TOOLKIT (SE-TOOLKIT)

For this attack to be successful you will need a basic understanding of SE-Toolkit. Go to <https://www.trustedsec.com> for more information. This procedure will be explained in more detail later in the article.

- What is SE-Toolkit?

STEP 6: HIJACK A USER'S PC THAT IS NOT LOCKED

This step involves finding a victim that has left their PC unlocked. In other words, they did not invoke ctrl+alt+del on their Windows PC. Then changing the www.google.com/mail shortcut to point to 192.168.153.158. How do you change a bookmark shortcut?

STEP 7: TELL THE USER TO GO TO THEIR GMAIL ACCOUNT AND SEE WHAT I'VE SENT

This step involves telling the victim to go to www.google.com/mail. More than likely the victim will go to their shortcut and not really check the URL bar. That will be illustrated later on in this article.

THE ATTACK

The perp's victim is using Windows XP-SP3 computer on Acme Corps network, but it doesn't really matter what version the victim is using because this will be a link to a spoofed website. The perp is a developer in the e-commerce department with administrative rights to his computer. The perp is using a VMWare Kali Linux image that comes with se-toolkit. The perp fires up his Kali Linux VMWare image and clicks on Applications → Kali Linux → Exploitation Tools → Social Engineering Toolkit → se-toolkit as illustrated in figure one:

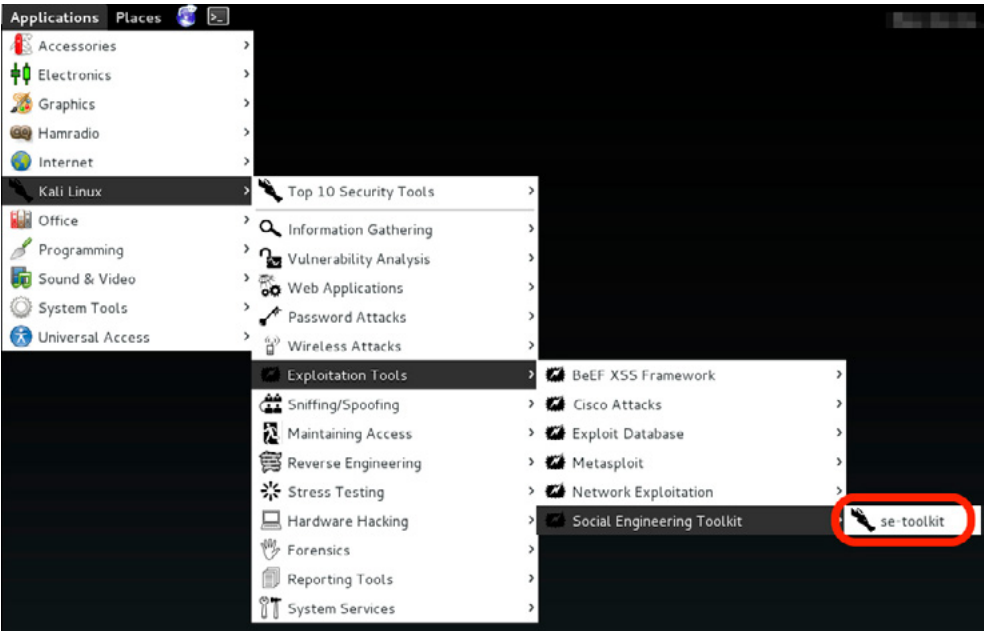


Figure 1. Starting se-toolkit

Figure 2 Illustrates the initial Social-Engineer Toolkit window.

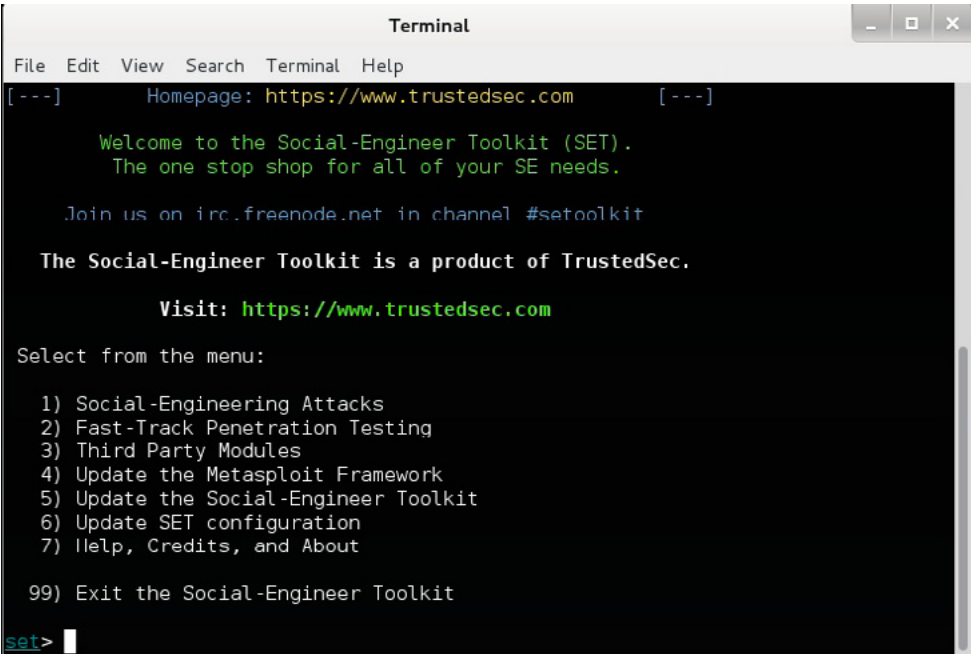


Figure 2. Initial se-toolkit Window

Here are the steps to begin the attack:

- Start by typing 1 for *Social-Engineering Attacks* at the command prompt and hit enter.
- Next, type 2 for *Website Attack Vectors* and hit enter.
- Next, type 3 for *Credential Harvester Attack Method* and hit enter.
- Next, type 1 for *Web Template* and hit enter.
- This is where the perp types in his IP address of his computer. For our example we will use 192.168.153.158. We will see this IP address again shortly. Once the IP address has been entered hit enter.
- The perp elects to use Gmail as his attack credentialed harvesting attack. Select 2 for Gmail and hit enter. See Figure 3.

```

should only have an index.html when using the import website
functionality.

1) Web Templates
158:webhacker> IP address for the POST back in Harvester/Tabnabbing:192.168.153.
1. Java Required
2. Gmail
3. Google
4. Facebook
5. Twitter
6. Yahoo

158:webhacker> Select a template:2
[*] Cloning the website: https://gmail.com
[*] This could take a little bit...

The best way to use this attack is if username and password form
fields are available. Regardless, this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:

```

Figure 3. *se-toolkit Running Spoofed Gmail Login Server*

- Now the perp goes over to the victim's machine. The perp notices that the victim's machine is not locked (no ctrl+alt+del). The perp goes to the victim's Chrome bookmark page and changes their Gmail shortcut to point to his spoofed Gmail login server – 192.168.153.158 (Does the IP look familiar?).
- The perp then waits for the victim to get back to their desk. The perp goes over to the victim and says, "Hey, I just shot you an email to your Gmail account. Go check it out"
- The victim logs into the spoofed site not noticing that the IP has not been obfuscated. (See Figure 4) The victim enters their user name and password. Once the user name and password has been harvested se-toolkit redirects the victim to their actual Gmail account. We find that the victim's credentials for Gmail login were cached, so when the redirect happened the victim didn't notice the redirect.

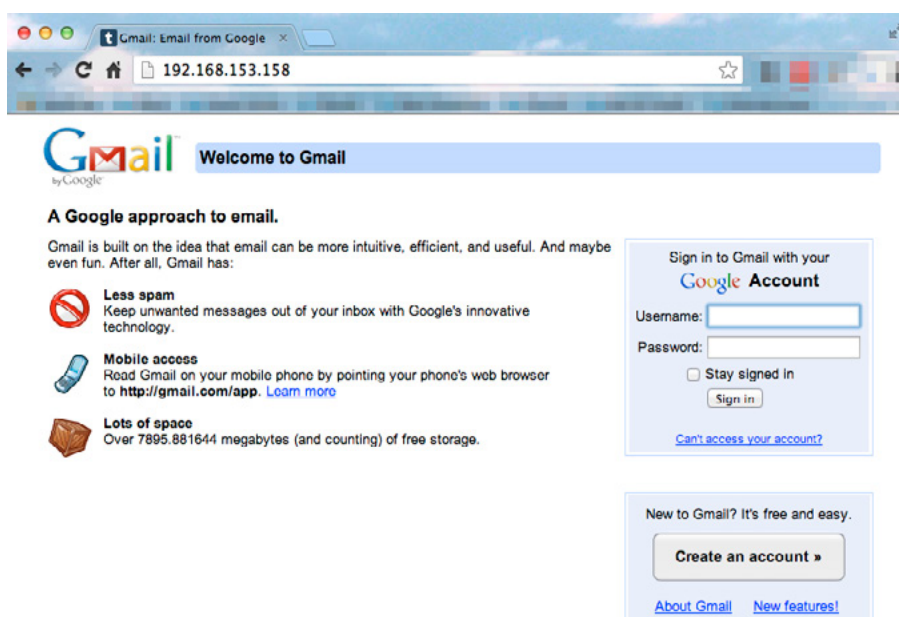


Figure 4. *Spoofed Gmail Login Server*

The only thing the victim missed, that could have tipped him or her off, is the fact that the IP address was not obfuscated. However, most users trust their bookmarks.

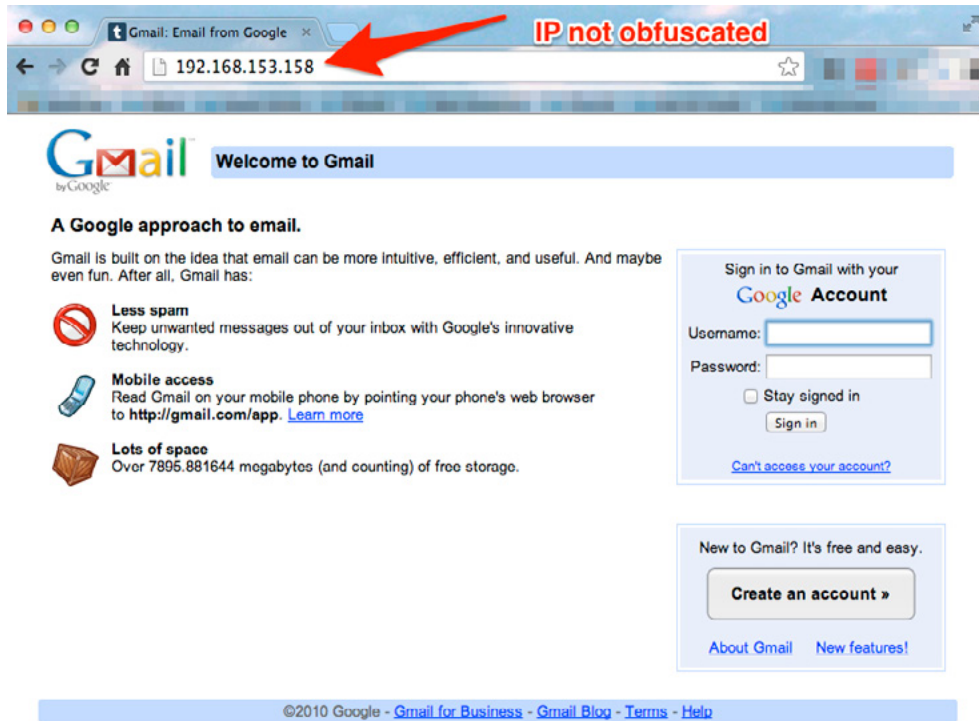


Figure 5. *IP Not Obfuscated*

Figure 6 illustrates what the perp collected. As you can see the perp got the full email address, which isn't a big deal. However, the perp also got the password, which would be considered strong but no match for se-toolkit and a determined perp.

IN SUMMARY

So, what's the moral of this story? The moral of this story is that companies need to be very vigilant about awareness training. Many organizations brush awareness training off and are willing to take the risk. Companies figure they can absorb these types of attacks, which they probably can.

However, what about the victims? This paper illustrated the ease with which to get a victim's credentials. Everyone should be aware of how easy this is and to heed the following advice:

- Know your bookmarks. Make sure they have not changed.
- Look in the URL field where the web address is illustrated. Ensure that the web address looks right. For example, *www.google.com* isn't spelled *www.gooogle.com*.
- Ensure there is a digital certificate that is signed by the hosting company. Just because you see a little lock in the browser does not mean it is the actual hosting company. However, you have to be diligent about reviewing the certificate.
- Have a healthy level of paranoia.

This advice is not foolproof and even the best can be fooled. Being aware of this attack and doing your best to thwart it can greatly improve your chances of staying safe. The digital world can be a fun place to learn and also an evil place where perpetrators lurk.

ON THE WEB

- [http://en.wikipedia.org/wiki/Social_engineering_\(security\)](http://en.wikipedia.org/wiki/Social_engineering_(security)) Social engineering (security)
- <http://kali.org/downloads> Kali Linux 32 Bit
- <https://my.vmware.com/web/vmware/downloads> VMware Downloads
- <https://www.trustedsec.com/downloads/social-engineer-toolkit> Social Engineer Toolkit

ABOUT THE AUTHOR



I have been in the information security field since 1990. I started my career with the United States Army as a Communication Security Specialist. I have acquired my CFE, CISSP, ISSMP and CISM. I have helped develop information security risk management programs for several Fortune 500 companies. I currently work in the retail sector for a Fortune 50 organization. For the past two years I have taught Digital Forensics at Western Connecticut State University. You can view some of my background information at <http://www.linkedin.com/in/kevinmoker/>.

WHAT TO EXPECT WHEN YOU'RE ENCRYPTING

CRYPTOGRAPHIC CHOICES FOR MAC AND WINDOWS

by Eric Vanderburg

There are a variety of options for encrypting data whether you are a Macintosh or Windows user. Some products work for both platforms but Apple and Microsoft have also developed their own built-in products geared towards protecting your data from unauthorized access. These encryption choices are presented here so that you can protect your data no matter which system you want to use.

What you will learn:

- How vulnerabilities were discovered and patches released historically
- How vulnerabilities are being sold on the open market
- Motivations for the sale of vulnerabilities

What you should know:

- The impact the vulnerabilities market has on secure computing
- The value of a new information commodity
- Ethics of intentionally building vulnerabilities into software

Cryptography is an interesting field of study and it forms the basis of much of the communication the average person takes for granted as they use computers, networks and the Internet. Encryption is the process of making a message such as a data file or communication stream unreadable to anyone lacking the appropriate decryption key. Encryption uses mathematical formulas to modify the data in such a way that it would be extremely difficult to put back together without the key. The information is combined along with a different routine of information making it impossible for any user to decrypt unless the key and the routine are available. Encryption has been used for thousands of years. The Caesar cypher is a method of scrambling text by substituting one character for another. Other early encryption methods used transposition where the order of characters were changed. As encryption became more mature, transposition and substitution were used in increasingly complex ways. Today, encryption methods are so complicated that most encryption and decryption operations are performed by computer. Computers also make it easier for end users and companies to encrypt data such as data on cell phones or personal computers. The forensic investigator must also be able to decrypt files in order to analyze them. Both Apple Macintosh (Mac) and Microsoft Windows machines come with built-in encryption and there are a variety of 3rd party applications used for encryption as well. This article explores these forms of encryption and how they differ as well as how the forensic investigator can decrypt these files to work on them.

WINDOWS ENCRYPTION

There are two types of built-in encryption features available for Microsoft Windows machines. They are BitLocker Drive Encryption and Encrypting File System (EFS). There are major differences between these features. While BitLocker assists the users in securing the files and folders available in the hard drive, Encrypting File System protects individual files. BitLocker is also used to secure removable drives and media. As such, the major difference between the functionality of the two standards is the way they secure the files in the drive. BitLocker secures drives and EFS secures files and folders within a drive.

Another major difference is that, BitLocker secures files irrespective of the users associated with it, which means that all the users associated with the computer can turn on/off this feature. But Encrypted File System uses individual accounts and permissions while encrypting files. Users can encrypt only those files that belong to them. BitLocker uses a special microchip, called Trusted Platform Module (TPM) which is hardwired to the motherboard of machines that require all advanced encryption features. But EFS does not require any such additional hardware. Moreover, only administrators have the right to turn on/off BitLocker advanced encryption features, but EFS does not require administrative permissions. All individual users can encrypt their files if needed. EFS security keys are stored in the operating system making it accessible to hackers who have skills and expertise in reading the source code of operating system. But BitLocker keys prevent the operating system itself from booting making it impossible for using the hard drive from a different computer. As individual users, you can use both BitLocker and EFS for added security.

BITLOCKER

BitLocker supports hard drives for Windows Vista and other recent Microsoft Windows operating systems. If you are using Windows XP, Windows 2000 and 2003, you will not be able to use BitLocker. Even though BitLocker services are available for Windows 7 and later editions, Windows 7 Home and Professional users cannot use the functionality. BitLocker encryption is available in 128 bit or 256 bit modes. The difference between these modes is in the amount of data that is uniquely used to generate cypher-text blocks. The larger the blocks, the harder it is to detect patterns in the encryption and to break encryption keys.

There are two possible ways by which data on a BitLocker encrypted hard drive can be accessed offline. One way is to boot the system from another operating system and the other way is to use the hard drive in another machine. They are called offline attacks. BitLocker comes to your rescue in both the above hacking techniques. Since the entire hard drive is encrypted, both the above methods become useless and your data is protected. You may wonder if you can access your own data from another machine or operating system with such high end encryption techniques. Well, in such case, you will be sent a recovery key which can be used to access data. This ensures that, your data does not end up in wrong hands, but you need not work harder for accessing your data. An enhancement to BitLocker is BitLocker to Go, which can be used to encrypt files and folders in removable hard drives such as USB drives, thumb drives etc. The major factor that you need to consider is BitLocker cannot be enforced while the operating system is running. Since BitLocker can be used only for avoiding offline attacks, you need to rely on standard operating system security techniques in protecting your computer while it is running. The major attacks on your system during its running time may be from malicious users trying to access the machine either locally or using remote connection. In either ways, your operating system should provide you with strict user access permissions and password policy by which such attacks can be eliminated.

ENCRYPTING FILE SYSTEM (EFS)

Encrypting File System (EFS) is relatively simpler and less effective than BitLocker, but it can be used to encrypt individual files on user level. The entire process is simpler that only a checkbox needs to be selected in order to turn on encryption. The checkbox is available in the file and folder properties. Users can also assign read/write permissions for various other users. The files are ready to use once they are opened. When the checkbox is unselected, the file becomes decrypted and any user can access them. EFS is available only certain versions of Windows 7 operating system. Windows 7 Home Basic, Home Premium and starter do not support EFS. There is an alternative for such Windows users where files can be decrypted using Cipher.exe command in the command prompt. An encrypted file can also be modified and copied to local system. The EFS certificated can also be imported and stored as back up files in local system.

MAC ENCRYPTION

While Windows uses BitLocker and EFS encryption technologies, Apple Mac OS uses FileVault. FileVault can be used in encrypting the entire drive for privacy. FileVault version 1 requires Mac OS 10.3 Pather, Mac OS 10.4 Tiger, Mac OS 10.5 Leopard or Mac OS 10.6 Snow Leopard. FileVault 1 encrypted a user's home directory but it did not encrypt the entire drive. Users create a password that is used to decrypt the files. If this password is lost, a recovery key may be used as well to decrypt the files.

FileVault 2 expands the functionality of FileVault by using the Advanced Encryption Standard (AES) 256 bit keys. It can also be used to encrypt the entire drive. FileVault 2 uses significantly more CPU than FileVault and decryption can be performed with a password or recovery key similar to FileVault. FileVault 2 requires Mac OS 10.7 Lion, Mac OS 10.8 Mountain Lion or Mac OS 10.9 Mavericks installed in the system.

Enabling/Disabling FileVault is an easy task as one simply needs to navigate to System Preferences page and click on Security and Privacy. Click on FileVault tab in Security and Privacy page, to enable/disable the services. There may be situation when multiple user accounts are available in a system. In such cases, administrators need to decide which users are allowed to unlock the encrypted drive. Only those users who are given permission to unlock the drive can access the system. Thus users who do not have permission to unlock cannot login to the system. Only after authorized users unlock the drive, will other users be able to use the system.

Once the users are assigned permissions for unlocking the drive, a recovery key is displayed which comes in handy when users forget the password for unlocking the drive. The recovery key can be used in such situations to unlock the drive and set a new password. It is advisable that recovery key should be stored externally in secure places, other than storing the key in the system itself because, when the system is locked, the recovery key will also be encrypted and cannot be accessed when you forget the password. The recovery key can also be stored with Apple in the cloud. You will be given option for storing the recovery key with Apple once it is displayed. If you prefer to store the key with Apple, you will need to answer three secret questions. The answers you provide for the questions will be used for encrypting the recovery key which is sent to Apple. The only way by which you can retrieve the key from Apple is by answering the questions.

There are questions regarding when one would require using FileVault in their system? It depends on the sensitivity of data stored in the system and the level of mobility. For example, a desktop computer working as server will not require high level of protection but any laptop would require FileVault since there are chances that laptops get missed or stolen from any place. Also, highly sensitive data should always be encrypted to ensure restricted access. It is also important for you to copy files from one encrypted drive to another encrypted drive since Mac OS does not warn while files are copied from encrypted drive to insecure drive. Another important factor that you need to consider while encrypting your drives is that, you should not encrypt your back up drive with FileVault. This is because, if any problem occurs to your system drives, you will need to access backup drives from non-Mac machines, causing serious troubles. Any encrypted Mac drive is safe only until it is unlocked. Once it is unlocked, any user can access the drives and files stored in the drive. Hence, it is essential for you to use strong passwords that remain as a mystery to hackers.

SYMANTEC DRIVE ENCRYPTION

Symantec encryption standard comes with complete protection of data present in the system. All the files including user files, hidden files, system files and swap files can be encrypted using Symantec Drive Encryption. This encryption standard can be used in any type of systems such as laptops, removable media etc. All the data encrypted in the system can be managed through Symantec Encryption Management.

There are many key features available in the standard such as machine recovery, user friendly and PGP strong. Some of the key benefits of Symantec Drive Encryption are silent deployment which is nothing but rolling out of data without end user involving in the process, multi-platform coverage so that all types of systems such as laptops, PC, drives etc, high performance in almost all operating systems including Windows, MAC OS X and Linux operating system. This standard is used in many organizations and is getting popular very rapidly. Since this standard is used worldwide, there is long term strategy in place which will benefit you. As mentioned above, all the operating systems including Windows 8, Windows 7, Windows XP, Server operating system, MAC OS, Ubuntu and Red Hat Linux operating

systems. Many different keyboard languages are also supported including English, Belgian, Dutch and many more.

CHECKPOINT FULL DISK ENCRYPTION

Check Point Full Disk Encryption is another set of standards for encryption of files and folders present in any of your computer systems, laptops, smart phones etc. All the files such as operating system files, deleted files, temporary files and important user data can be encrypted and used with ease and privacy. An additional feature available for users is pre boot authentication which ensures user identity, thereby assisting in high level data security along with encryption that ensures data is not lost.

There are many benefits of using Check Point Full Disk Protection. This standard is similar to all the standards discussed above and it provides all general encryption functionalities. The encryption functionality comes into picture when your laptops are stolen as it prevents unauthorized users from getting entry into the system. This encryption mechanism supports all certifications including common criteria and BITS. The software can be used in almost all platforms ranging from Microsoft Windows to Apple MAC. The software has been used in many organizations ranging from less than 1000 seats to more than 100,000 seats. The software has also been the leader in mobile data protection which is a rapidly developing field in information security and privacy. You will also be getting a centrally managed end user solution which works with other security software architectures as well.

CREDANT MOBILE GUARDIAN

Mobile Guardian from Credant Technologies has been a leader in encryption software for the recent years. Their complete protection of data at administrator level along with easy to use controls and simple user interface, the software has been selling hot cakes across organizations. Many advanced features are supported in the software that makes it a best buy. Many positives have been identified with the software but there has not been any negative related to the software which makes it one of the best encryption software available in the market.

Starting from installation of the software to use of software at end level, the easy to use interface and controls make the process simpler. The wizard available for installation and securing data is simple and takes only a couple of minutes for the process to get completed. There is also proper documentation related to configuration of software and encrypting data. Once the software is distributed and installed in end user systems, Credant provides either 24 hour service or standard day service to the customers.

Well, the difference between Windows and Apple MAC encryption standards were analyzed in the beginning of this passage followed by various other encryption standards. With so many options for the organizations and individual users to choose from, encryption is no longer a daunting task. All you need to do is, purchase the product from the vendor and sit back and enjoy protected data. There will be many more standards getting introduced in the near future which will make the process easier.

DECRYPTING WITH ENCASE

With the need of digital forensics on the rise, many companies are striving hard to bring out products that investigate various computer systems, laptops etc. EnCase is one such product introduced by Guidance Software, which is a complete suite of forensics products aimed at bringing out hidden information, also allowing for file encryption etc. EnCase suite of products have been used in various forms of investigation and they are popular worldwide. The product is being used by almost 50% of Fortune 100 and Fortune 500 companies along with government agencies all around the globe. EnCase comes in three forms namely eDiscovery, Cyber security and Analytics. There are many advantages of using EnCase in a technology company since many cyber threats arise from such organizations. The latest of the EnCase suite of products is EnCase Forensic v7.08 which comes as the fastest forensic tool available in the market. The EnCase suite of Forensic tools can also be used for tablets, smart phones, removable media such as CD, DVDs, Pen Drive, Hard disks etc. The reports that are generated by the forensic tools are submitted to authorities requesting such analysis. The reports that are generated by the forensic tools are accepted in legal systems of many countries. The reports are generated based on the requirements of the clients, thereby helping them understand the findings in a granular manner.

Encase supports all the encryption standards that are discussed above. For example, Encase supports Microsoft's BitLocker and Encrypted File System. Apart from these two standards, Encase supports various other standards as well. Various other disk and volume encryption standards supported by Encase are McAfee Safeboot, PGP whole disk encryption, Full disk encryption, Utimaco Safeguard Easy and many more. Apart from Encrypting File System, Encase supports CREDANT mobile guardian and RMS.

REFERENCES

- S. Bunting, "EnCase Computer Forensics: The Official EnCase Certified Examiner Study Guide, 3rd Edition" John Wiley & Sons, Indianapolis, Indiana, 2012
- "What's the difference between BitLocker Drive Encryption and Encrypting File System?", Microsoft, 2013, retrieved from <http://windows.microsoft.com/en-in/windows7/whats-the-difference-between-bitlocker-drive-encryption-and-encrypting-file-system>
- T. Kessler, "OS X FileVault Questions Answered," 2012, retrieved from http://reviews.cnet.com/8301-13727_7-57398382-263/os-x-filevault-questions-answered/
- "How Drive Encryption Works", Symantec White Paper, Mountain View, CA, 2012

ABOUT THE AUTHOR



Eric A. Vanderburg, MBA, CISSP

Director, Information Systems and Security, JurInnov, Ltd.

Eric Vanderburg understands the intricacies inherent in today's technology and specializes in harnessing its potential and securing its weaknesses. He directs the efforts of multiple business units including Cyber Security, eDiscovery, Computer Forensics, Software Development, IT and Litigation Support at JurInnov, an eDiscovery and eSecurity consulting firm. Vanderburg holds over thirty vendor certifications and is completing a doctorate in information assurance. He has dedicated much of his career to designing and implementing systems, policies and procedures to enhance security, increase productivity, improve communications and provide information assurance. He has been invited to speak at conferences and events on technology

and information security and he is active in promoting security and technology awareness through various publications. Look for his latest book, "Storage+ Quick Review Guide", due for publication with McGraw Hill in December of 2013.



GUIDANCE SOFTWARE

The Standard in Digital Investigations.

www.encase.com

Guidance
SOFTWARE



FORENSIC APPROACH TO ANALYSIS OF FILE TIMESTAMPS

IN MICROSOFT WINDOWS OPERATING SYSTEMS AND NTFS FILE SYSTEM

by Matveeva Vesta Sergeevna, Leading specialist in computer forensics, Group-IB company

All existing file browsers display 3 timestamps for every file in NTFS file system. Nowadays there are a lot of utilities that can manipulate temporal attributes to conceal the traces of file using. However, every file in NTFS has 8 timestamps that are stored in file record in MFT and are used in detecting the fact of attributes substitution. The author suggests a method of revealing original timestamps after replacement and automated variant of it in case of a set of files.

What you will learn:

- Timestamps structures in NTFS file system
- Tools for timestamps analysis
- Experimented data on timestamps changing in the file system
- Method and pseudocode for detecting timestamps substitution

What you should know:

- MFT structure
- Forensics tools like Autopsy Browser, AccessData FTK Imager
- Time format in file records of NTFS file system

According to data from Wikipedia, typical character of the detective genre is a criminal who «commits a crime, wipes out tracks and tries to sabotage the investigation». There's nothing so novel in Wikipedia, but! We have to note that it seems impossible that a man who broke the law and committed act injurious to the public wouldn't try to save himself interfering with the investigation or hiding the evidences (exclusions, of course, exist).

In virtual space everything is a bit more complicated on this matter. Before the emergence of real cases of investigation of computer crimes and rapidly upcoming aspect of the field of commerce as investigations and forensic inquiries, criminals had been leaving a lot of tracks after their activity in a "hacked" system. At present techniques to resist computer crime investigations are growing. In this article one of those techniques is discussed, as well as the veil of mystery of computer criminalists is lifted.

Substitution of time attributes of files. It is done either manually, by changing system clock, or using special programs which exist in sufficient number. But what is interesting is that the same functional capabilities are also coded in a series of malicious software with the purpose of deceiving a user and referring

the file rather as a system file than suspicious. Thus, in the properties of a file the replaced information will be shown. But it's not all that easy from the criminalistics point of view. For identification of a substitution, specific features of a file system are used. All the information below is equally correct both for Microsoft Windows and NTFS file system as for the most frequently seen combination of OS and file system.

In NTFS file system time attributes of files are contained in a file record for each file in main file table (hereafter – MFT). Curiously enough, the file contains only 8 (!) and not 3 as we get used to. Two structures are responsible for time attributes: `$STANDARD_INFORMATION` and `$FILE_NAME`, each of them contains: date and time the file was created, date and time of last changes in the file, last access to the file, and also date and time of last changes of data in file record in MFT. In figure 1 it is shown a file record by means of software “AccessData FTK Imager”:

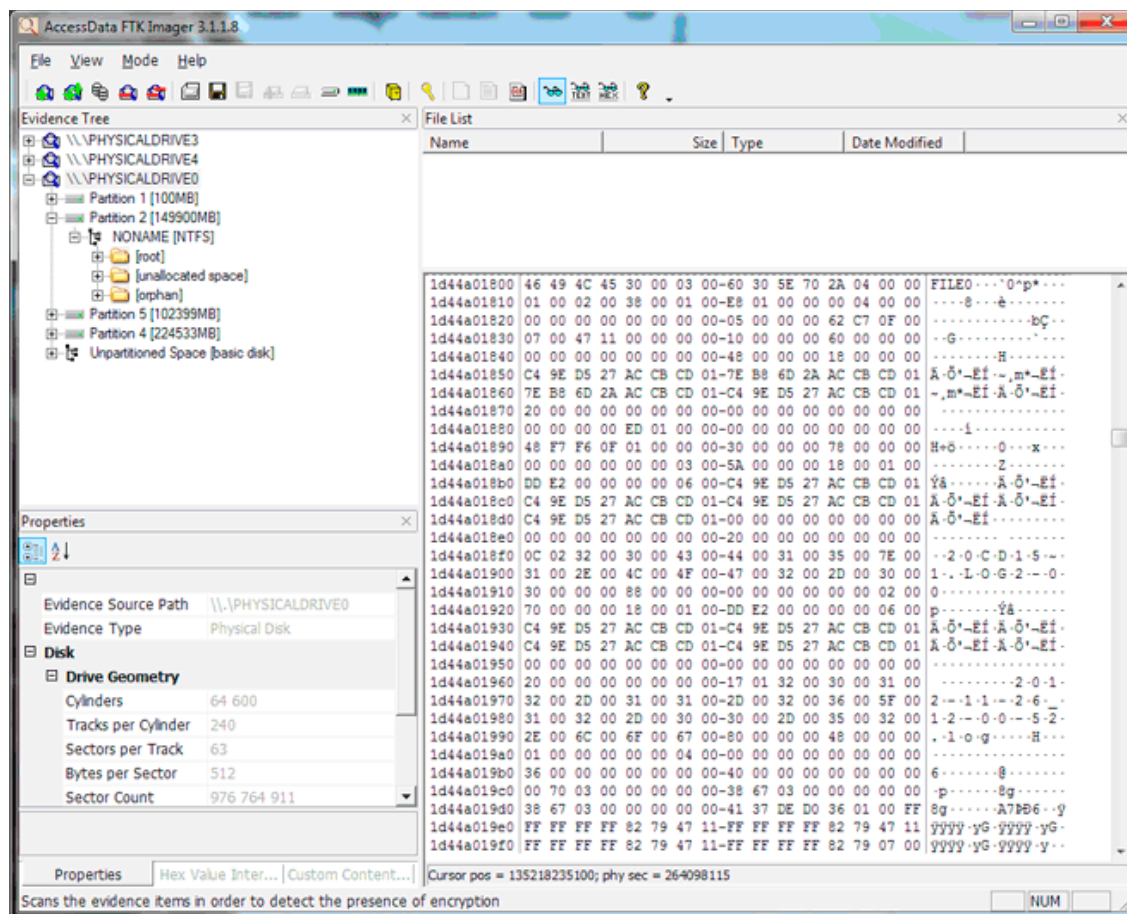


Figure 1. A file record by means of FTK Imager

Without going into details of displacement of above-mentioned structures, they can easily be recognized by their content. Time in these structures has OS Microsoft Windows format, whose records are as follows:

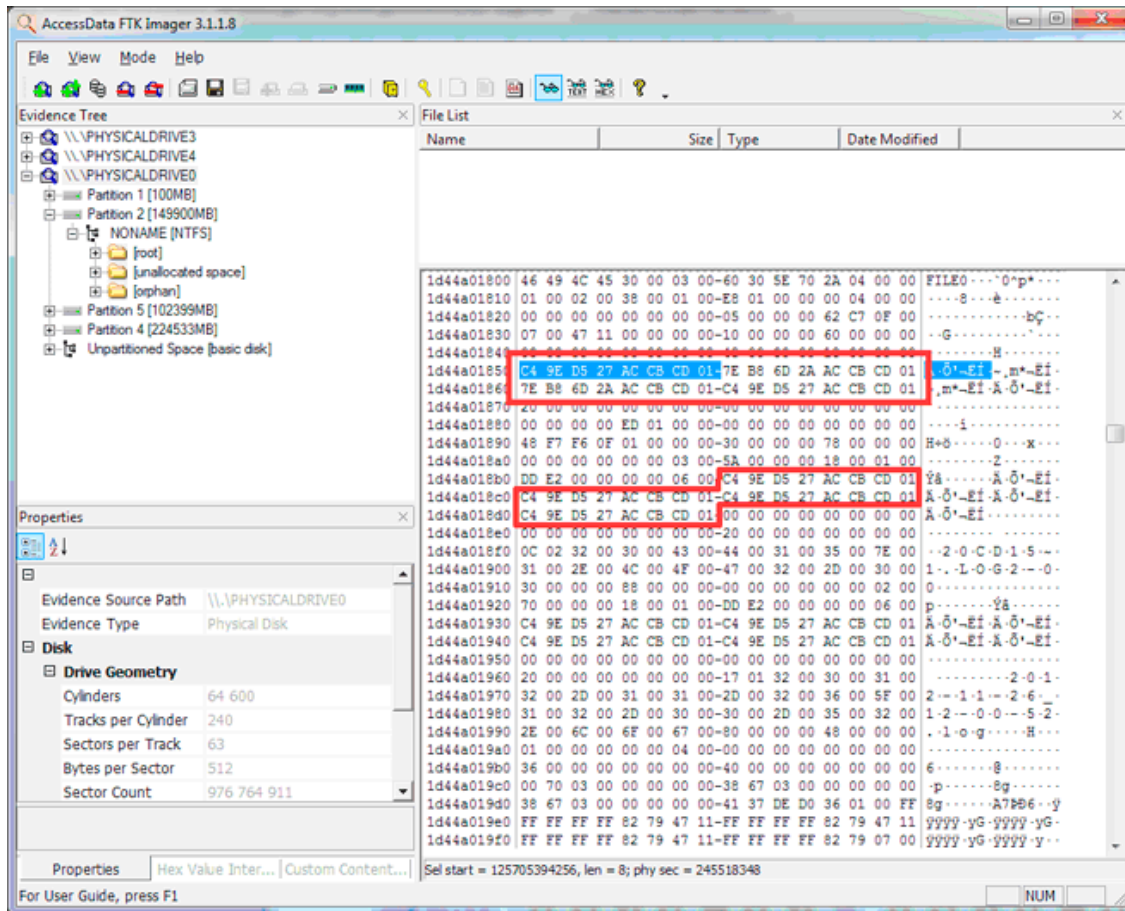


Figure 2. A file record by means of FTK Imager

For transformation of timestamp, which is indicated in file record, one can use specialized tool «DCode Date»:

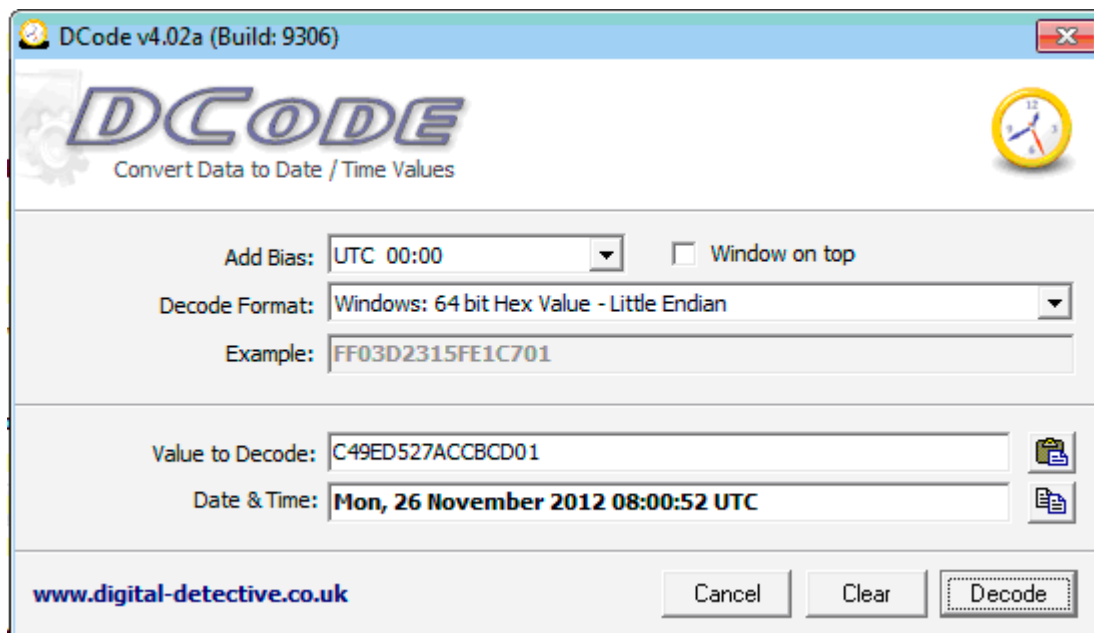


Figure 3. DCode window

or one can transform it manually. However, it is possible to get access to all eight attributes using automated tools for analysis: The Sleuth Kit (TSK) with a command «istat»:


```

root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help
root@SIFT-Workstation:/home/sansforensics/Desktop# istat -o 63 image.001 2202-128-4
MFT Entry Header Values:
Entry: 2202      Sequence: 145
$logFile Sequence Number: 67454735
Allocated File
Links: 2

$STANDARD_INFORMATION Attribute Values:
Flags: Archive
Owner ID: 0
Security ID: 1766 ( )
Last User Journal Update Sequence Number: 5091182288
Created:      Thu Mar  2 16:00:00 2006
File Modified: Sat Mar 21 18:09:06 2009
MFT Modified: Tue Dec  4 14:40:20 2012
Accessed:     Mon Dec 10 09:13:17 2012

$FILE_NAME Attribute Values:
Flags: Archive
Name: SlknEKbbrSQbvDFPEI.exe
Parent MFT Entry: 38832      Sequence: 3
Allocated Size: 0      Actual Size: 0
Created:      Fri Nov 23 16:40:10 2012
File Modified: Fri Nov 23 16:40:10 2012
MFT Modified: Fri Nov 23 16:40:10 2012
Accessed:     Fri Nov 23 16:40:10 2012

Attributes:
Type: $STANDARD_INFORMATION (16-0)  Name: N/A  Resident  size: 72
Type: $FILE_NAME (48-3)  Name: N/A  Resident  size: 90
Type: $FILE_NAME (48-2)  Name: N/A  Resident  size: 110
Type: $DATA (128-4)  Name: N/A  Non-Resident  size: 167424  init_size: 167424
37077796 37077797 37077798 37077799 37077800 37077801 37077802 37077803
37077804 37077805 37077806 37077807 37077808 37077809 37077810 37077811
37077812 37077813 37077814 37077815 37077816 37077817 37077818 37077819
37077820 37077821 37077822 37077823 37077824 37077825 37077826 37077827
root@SIFT-Workstation:/home/sansforensics/Desktop#

```

Figure 4. Commands

or in a graphics interface using program «Autopsy Browser».

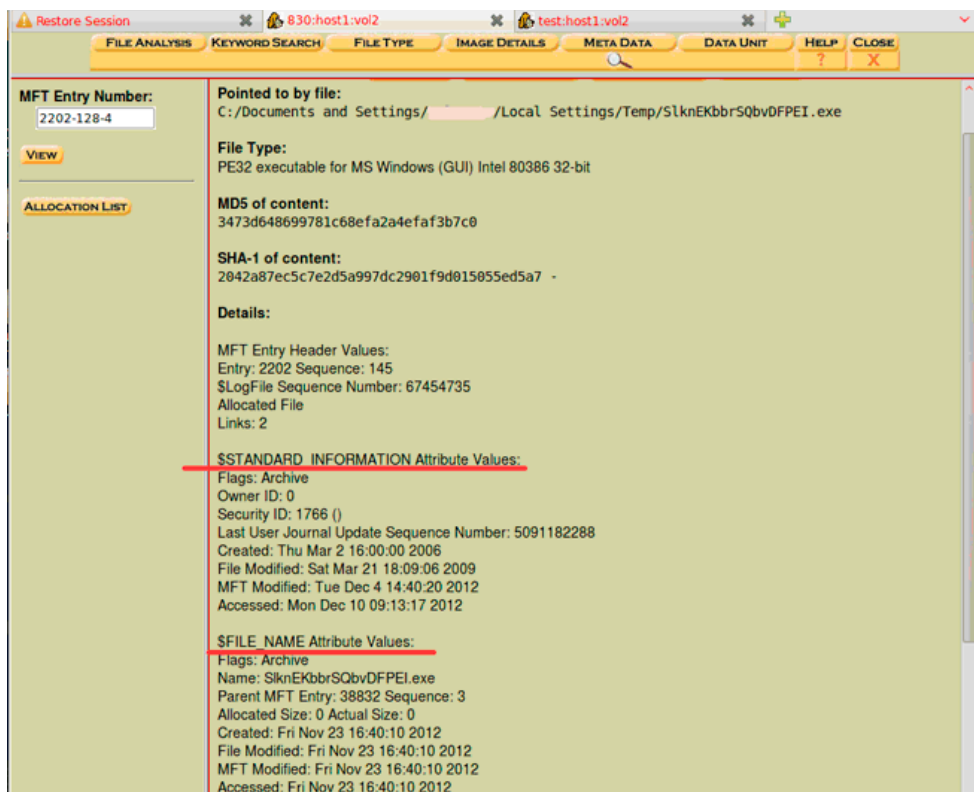


Figure 5. Commands

Now we come to the main point. Correct estimate of time attributes from `$STANDARD_INFORMATION` and `$FILE_NAME` structures gives a criminalist an opportunity to restore chronology of events correctly, which is very important during the investigation.

First of all it is essential to realize which actions on the files change their attributes. For that we conducted a series of tests on Microsoft Windows XP and 7 for processor's architectures x86 and x64, results of these tests are presented in table 1, 2 for text and executable files.

Table 1. Windows XP

FN	Rename	Local Move	Volume move	Copy	Create	Delete	Open	Properties	Attributes	Modify
Modification	x	x	x	x	x	x				
Accessed	x	x	x	x	x	x				
MFT modified	x	x	x	x	x	x				
Create	x	x	x	x	x	x				
	(SI)	(SI)	(created)	(created)		(SI)				

SI	Rename	Local Move	Volume move	Copy	Create	Delete	Open	Properties	Attributes	Modify
Modification					x					x
Accessed	x	x	x	x	x	x	x (PE) x(day)	x (PE)	x	x
MFT modified	x	x	x(PE)	x(PE)	x	x	x (PE)	x (PE)	x	x
Create			x	x	x					

Таблица – Windows 7

FN	Rename	Local Move	Volume move	Copy	Create	Delete	Open	Properties	Attributes	Modify
Modification	x	x	x	x	x	x				
Accessed	x	x	x	x	x	x				
MFT modified	x	x	x	x	x	x				
Create	x	x	x	x	x	x				
	(SI)	(SI)	(created)	(created)		(SI)				

SI	Rename	Local Move	Volume move	Copy	Create	Delete	Open	Properties	Attributes	Modify
Modification					x					x
Accessed	x	x	x	x	x	x(?)	x(day)	x(PE)	x	x
MFT modified	x	x		x	x	x			x	x
Create			x	x	x					

where the names of columns represent the following:

- Rename – changing the name of a file;
- Local Move – moving a file within a single file system;
- Volume move – moving a file from one file system to another;
- Copy – copying a file;
- Create – creating a file;
- Delete – deleting a file;
- Open – opening a file;
- Properties – viewing the properties of a file;
- Attributes – changing the attributes of a file;
- Modify – modifying a file;
- row entries represent time attributes of files, which are contained in `STANDARD_INFORMATION` (SI) and `FILE_NAME` (FN)

- x – changing the attribute of a file;
- x (PE) – changing the attribute of only PE-format files;
- x(day) – changing the attribute is performed only once a day at the first call to a file;
- x(?) – observed cases of changing the attribute, which occur not every time;
- (SI) – all data in FILE_NAME structure is copied from previous STANDARD_INFORMATION structure;
- (created) – all attributes in FILE_NAME structure match the date of creation of a file from STANDARD_INFORMATION structure.

As a result of similar tests with different file formats and OS, slight differences from tables 1, 2 were observed.

As we can see, data in \$FILE_NAME structure is created in the moment of creation of a file and constitute a copy of a date of creation and it changes due to renaming, local move and move between file systems, copying and deleting the file. Consequently, data in this structure cannot exceed time attributes from \$STANDARD_INFORMATION structure. But it is relevant only for time of creation and changes of file.

Time of last access to a file and attribute MFT modified can change only in \$STANDARD_INFORMATION structure. This is why time of last access to a file should be determined using \$STANDARD_INFORMATION structure. In OS Microsoft Windows Vista and 7 time of last access does not change by default in for the economy of system's resources. Specified option is controlled by the value of register key: HKEY_LOCAL_MACHINE\SYSTEM\ControlSet\Control\FileSystem.

NtfsDisableLastAccessUpdate = 1 (date and time of last access to a file are not changed by accessing a file) = 0 (date and time of last access to a file are changed by accessing a file). Tables 1, 2 show the results of tests with an option of changing the date of last access during call to a file.

Attribute MFT modified changes when at least one attribute of file record is changed. But in view of time resolution of NTFS file system, which is up to 1 hour, information about last access to a file is firstly saved in operation memory and then is recorded in file record. Consequently, there can be slight differences in these attributes. But it should be noted that changes in attribute Accessed do not always lead to changes in the attribute MFT modified.

In the view of described limitations, we will look into the approach of criminalist during the analysis of HMDD which was used during the fraudulent transaction in internet-banking system from the client's side. Figure 5 shows time attributes of malicious program of «Trojan.Carberp» family. Such programs are of frequent occurrence in such incidents as they allow copying of data which is necessary for authentication in systems of internet banking without user's notice. These programs have functionality to change displayed to user time attributes to identical for that of system files.

Very important aspect of restoring the chronology of events is the correct estimate of time of loading of a malicious program. Let's try to use obtained in this article information for determination of time attributes of file «SlknEKbbrSQbvdFPEI.exe».

Since attributes in \$FILE_NAME structure cannot exceed attributes in \$STANDARD_INFORMATION structure, then date of creation of file is: 23/11/2012, 16:40:10.

The original date of changes of a file is, however, unknown since in structure \$STANDARD_INFORMATION it is substituted, and in \$FILE_NAME – it is a copy of date and time of creation of a file, which was saved in file record at the creation of a file.

Date and time of last access is determined from structure \$STANDARD_INFORMATION – 10/12/2012, 9:13:17. Data in file record was changed last time in 04.12.2012, 14:40:20 in accordance with attribute MFT modified, which did not change during the access to the file.

Therefore, as a result of manual analysis one can find out correct time attributes of a file, which is quite a time-consuming work, if the number of files is large. For that reason below is given the pseudocode which allows to automate the process of determination of time attributes of a file getting as an input values of structures \$STANDARD_INFORMATION and \$FILE_NAME (date and time of last change of a file with substitution of attributes is taken from \$FILE_NAME structure):

Listing 1. Pseudocod for process automatisation

```
SI=null; // $STANDARD_INFORMATION structure
FN=null; // $FILE_NAME structure
Result=null; // real attributes structure

SI=receive_standard_information(file);
FN=receive_file_name(file);
If (SI!=0 and FN!=null)
{
Result.Created = FN.Created

If (SI.Modified < FN.Modified)
{
If (FN.Modified == SI.Created)
{
Result.Modified = SI.Modified
writeline ("The file was copied")
}
else Result.Modified = FN.Modified
}
else Result.Modified = SI.Modified

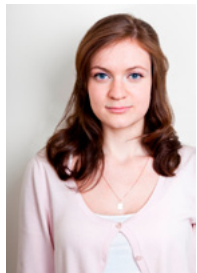
If (SI.Accessed < FN.Accessed) Result.Accessed = FN.Accessed
else
{
Result.Accessed = SI.Accessed
}
If (SI.MFT_modified < FN.MFT_modified) Result.MFT_modified = FN.MFT_modified
Else Result.MFT_modified = SI.MFT_modified

If (Result.Created > Result.Modified) writeline ("The file was copied to the OS")
}
```

There are a lot of refinements in determination of time attributes of files. But the main point of this article is the idea that using time attributes of a file it is possible to recover even actions applied to files which in turn helps computer criminalists in conducting the investigations and forensic inquiries.

REFERENCES

- Lee R., Windows 7 MFT Entry Timestamp Properties – SANS Forensics Community, 2010. – URL: <http://computer-forensics.sans.org/blog/2010/04/12/windows-7-mft-entry-timestamp-properties>.
- Carrier B., File System Forensic Analysis / B. Carrier – Pearson Education, 2005. – p. 400-502.
- Dave Hull, Digital Forensics: Detecting time stamp manipulation – SANS Forensics Community, 2010. – URL: <http://computer-forensics.sans.org/blog/2010/11/02/digital-forensics-time-stamp-manipulation/>
- Lance Mueller, Detecting timestamp changing utilities – 2009 – URL: <http://www.forensickb.com/2009/02/detecting-time-stamp-changing-utlities.html>

ABOUT THE AUTHOR

Matveeva Vesta Sergeevna graduated from the National Research Nuclear University «MEPhI» in 2011 where she obtained a major degree in Complex Information Security of Computer-based systems. At the moment Vesta specializes in computer forensics, doing her PhD degree in the Information Security and having an intense teaching practice with the students at the University.

In 2010 Vesta took part in IPICS academic summer school in Greece and presented work on theme: «Secure coding techniques».

From 2011 till present Vesta has been working in Group-IB, Russian and the CIS's (Commonwealth of Independent States) leading computer security company, specializing in the investigation of computer crime, information security breaches, and computer forensics.



Specializing in

- **iOS /OS X Forensics**
- **Mobility & Security Architecture**
- **Mobile Device Policy/BYOD**
- **Secure File Storage & Transfer/Cloud**
- **Open Source Integration**

WINDOWS FORENSICS AND SECURITY

by **Adrian Leon Mare**

The world we live in today is a technologically advanced world. While on one hand, commercialization of IT (Information technology) revolutionized our modern day lifestyle, it has raised a big question mark about the confidentiality and privacy of the information shared and managed using advanced means of communication.

As computer technology continues to evolve, the task of managing and handling private and sensitive information is becoming more and more challenging with each passing day. Increased rates of cyber crimes leading to unsolicited invasions of privacy have resulted in the emergence of a new field of computer science known as cyber forensics.

With the increasing demand of computer security in recent times, it has become more important than ever to understand the digital forensic technology.

WHAT IS DIGITAL FORENSICS/CYBER FORENSICS?

Also known as cyber forensics, computer forensics involve the application of acquiring and analyzing digital information (as a part of structured investigation) to be used as evidence in the court of law.

DIGITAL FORENSICS-PRIMARY GOALS

The primary goal of Digital Forensics is to carry out an organized and structured investigation in order to preserve, identify, extract, document and interpret digital information that is then utilized to prevent, detect and solve cyber incidents.

A typical forensic investigation consists of the following main steps:

- Preserving the data.
- Acquiring the data.
- Authenticating the data.
- Analyzing the data.

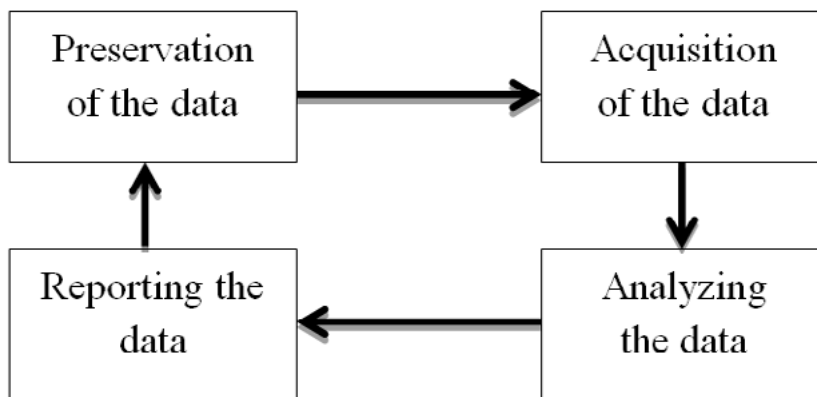


Figure 1. Steps involved in a Forensic Investigation Process

PRESERVING AND ACQUIRING THE DATA

The first and foremost step of a digital forensic investigation is to preserve and acquire the data from a computer. The step involves creating a bit by bit copy of the hard drive data.

AUTHENTICATING THE DATA

The next process involves verifying the data seized. To ensure that the acquired data is an exact copy of the contents of the hard drive, the md5/sha1 of the original and copied data are checked and matched.

ANALYZING THE DATA

This is perhaps the most important part of the investigation process which involves careful examination and analysis of the data using forensic tools.

The process mainly involves:

- Recovering deleted files /Data Recovery
- Tracking or identifying hacking activities

DIGITAL FORENSICS AND WINDOWS

21st century is the century of revolution and change. The transformation of the analog world into a digital world has raised new challenges and opportunities for technology lovers.

New forensic challenges arise with the introduction of newly released and latest operating systems. While on one hand, these newly released versions of Windows are aimed at making things easier for users, many of the functions (such as auto play, file indexing) performed by your operating system for your convenience can actually be used against you.

If you look at the current cyber crime statistics, you will notice that the highest percentage of cyber crimes is committed in the United States of America. 23% of the total cyber crimes take place in U.S. This calls for increased security measures to protect your confidential information from being misused.

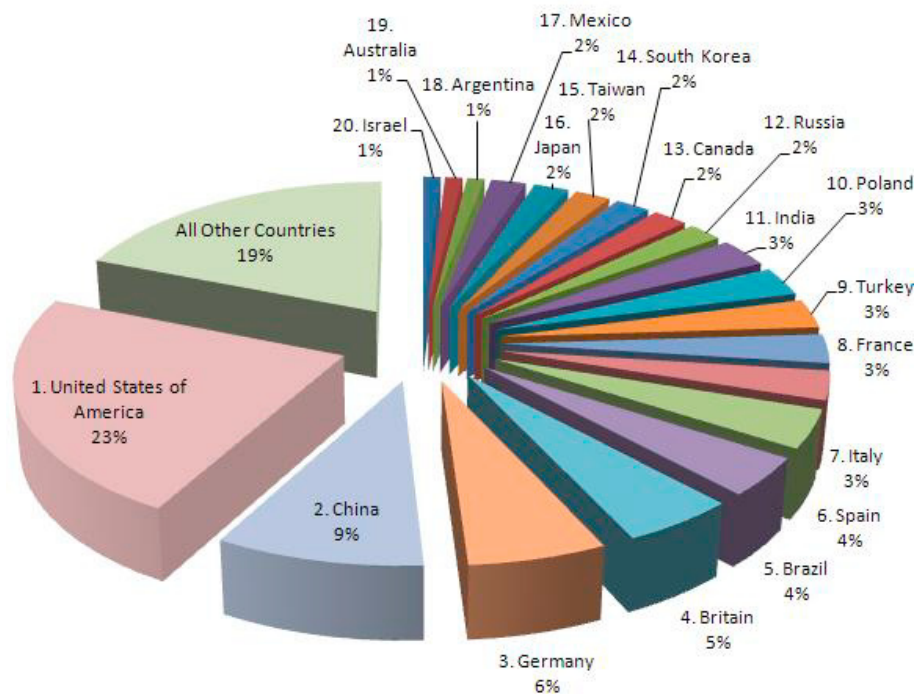


Figure 2. *Cybercrime*

The average user is mostly unaware of the fact that their newly upgraded operating system is leaving tracks of their activity. It is essential for users to know that valuable pieces of sensitive and confidential information is stored in Windows Artifacts. These artifacts can be used to recreate and restore the account history of a particular user.

DIGITAL FORENSICS AND WINDOWS-THE WINDOWS ARTIFACTS

Some of the artifacts of Windows 7 operating system include:

- Root user Folder
- Desktop
- Pinned files
- Recycle Bin Artifacts
- Registry Artifacts
- App Data Artifacts
- Favorites Artifacts
- Send to Artifacts
- Swap Files Artifacts
- Thumb Cache artifacts
- HKey Class Root Artifacts
- Cookies Artifacts
- Program files Artifacts
- Meta Data Artifacts
- My Documents Artifacts
- Recent Folder Artifacts
- Restore Points Artifacts
- Print Spooler Artifacts
- Logo Artifacts
- Start menu Artifacts
- Jump lists

Information collected from any of these artifacts can be used to recreate the account history of a user. To gain a better understanding of how these artifacts can be used to access or retrieve valuable information, it is essential to briefly discuss some of the most important Artifacts of Windows 7.

ROOT USER FOLDER ARTIFACTS

The Root User Folder gives access to the complete operating system. The Root User reserves the right to delete and modify files on the operating system besides having the rights to generate new users and award them some rights. Nonetheless, these rights cannot exceed the rights of a root user.

The Windows Folder is specified by `%SYSTEMROOT%`. The Folder can be accessed through `Start\Run\%SYSTEMROOT%\System32.`

DESKTOP ARTIFACTS

All the files present on the desktop of a user are stored in the desktop folder of the operating system. Typically, the desktop is populated either,

- By the user, or
- By programs that automatically create files and place them on the desktop.

The Desktop can be accessed using the following link: `C:\USERS\username\desktop.`

PINNED FILES/JUMP LISTS ARTIFACTS

Pinned Files or Jump lists is a relatively new feature introduced in Windows 7 released by Microsoft. Using the Jump lists all the pinned files can be accessed. Additionally, these lists also maintain a record of recently or last visited files relative to a particular software. Pinned files can be accessed from the jump list using the following link: `C:\Users\username\AppData\Roaming\Microsoft\Internet Explorer\QuickLaunch\UserPinned\TaskBar.`

RECYCLE BIN ARTIFACTS

The Recycle Bin stores the recently deleted files temporarily. These files can be restored easily. You can only view the Recycle Folder after un-checking the hide\protect system files option using the following link: `C:\$recycle.bin.`

REGISTRY ARTIFACTS

Registry is the location where the configuration information of Windows is kept and stored. It can be used to obtain information related to historical and current use of applications in addition to obtaining valuable pieces of information about option preferences and system settings. It can be accessed using the following link: `NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\WordWheelQuery.`

APP DATA ARTIFACTS

Application data or App data is a junction designed to provide backward compatibility. A junction can roughly be defined as a shortcut that serves to redirect programs and files to different locations. All the information related to settings configuration (of various apps) is stored in this folder. Furthermore, information related to the Windows address book and recently accessed files are also stored in this folder. The junction can be accessed through: `C: User\ (username)\AppData\Roaming\folder.`

FAVORITE ARTIFACTS

The folder contains valuable bits of information related to Windows Explorer and Internet Explorer favorites. The folder can be accessed using the following link: `C:\USERS\username\favorites.`

SEND TO ARTIFACTS

The Send to folder stores information pertaining to shortcuts to different locations, and other software apps on the operating system of your computer. These shortcuts serve as destination points. Using these destination points a file can be sent or activated. Furthermore, these points can also be modified as per your convenience. The Send to folder can be accessed using the following link: `C:\Users\username\AppData\Roaming\Microsoft\Windows\SendTo.`

SWAP FILES ARTIFACTS

Page Files or Swap files are the memory files of your computer that aid in expanding the memory of your computer. These files are not visible and are hidden by default settings. To view these files, following link can be used: `MyComputer>Properties>Taskmenu>AdvancedSystemSettings>Advancedtab>Performance>Settings>Performance options dialogue box>Advanced tab>Change.`

THUMBS CACHE ARTIFACTS

Thumbs.db files are files that are stored in every directory on the Windows systems that includes thumbnails. These are default files (created by default) and store valuable information that is not available elsewhere. The file is created locally amongst the images. The location where cache is stored is as follows: C:\Users\Username\AppData\Local\Microsoft\Windows\Explorer.

The display can be stopped by a user by checking on the 'Always show icon, not thumbnails' from the list of Folder options.

HKEY CLASS ROOT ARTIFACTS

The HKey Class Root or simply HKCR key contains sensitive information about different file name extensions in addition to containing information related to COM class registration. Furthermore, it is designed to be compatible with the 16-bit Window registry.

HKEY_LOCAL_MACHINE and HKEY_CURRENT_USER key both store valuable information related to file name extensions and class registration.

HKEY_LOCAL_MACHINE\Software\Classes: This key stores all the information pertaining to different users using the system.

The HKEY_CURRENT_USER\Software\Classes: On the other hand, this key stores information pertaining to the interactive user.

COOKIES ARTIFACTS

A number of website store information on your computer in the form of cookies. Cookies can roughly be defined as small text files containing information related to preferences and configuration of a particular user. These files can be accessed using the following link: C:\User\username\AppData\Roaming folder\Microsoft\Windows\Cookies.

PROGRAM FILES ARTIFACTS

Windows 7 consists of two Program files folders including;

- C:\program files
- C:\Program files (x86)

These folders are designed to be compatible for 32 bits and 64 bits version of Windows 7. The first one is compatible with the 64 bit version of Windows 7, whereas, the second one is compatible with 32 bit version of Windows 7.

META DATA ARTIFACTS

Meta Data simply refers to information related to data itself. Using the metadata artifacts, valuable strings of file information can be obtained that can be used as evidence in digital forensic investigation.

RESTORE POINTS ARTIFACTS

Windows & gives its users the option of restoring points thereby creating the image of your system. This essentially helps in providing users with an option to revert back to the point when the system was working perfectly in case of fatal system errors. This system image also contains the drives that are required by your operating system to run in addition to including program settings, system settings and file settings.

MY DOCUMENTS ARTIFACTS

My Documents contains all the information related to files that have been created by users themselves. Usually when a program is installed on a system, the information is stored in this folder. It is also known as the primary storage space meant for storing all the key information. The folder can be accessed through: C:\Users\username\MyDocuments.

START MENU ARTIFACTS

The traditional Start menu has been replaced by Start in Windows 7. Using software like classic shell, it is absolutely possible to get the menu back. In Windows 7, the right column of the start (new version of start menu), links to respective libraries are shown instead of folders.

LOGO ARTIFACTS

The Logos included in the Windows 7 Operating System include valuable information pertaining to application events information, security related events information, setup event information, forwarded event information, and application events information.

PRINT SPOOLER ARTIFACTS

Print Spooler is a software program responsible for organizing all the print jobs that have been sent to the print server or the computer printer. In essence all the print related information is stored in this folder. The folder can be accessed by using the following link: C:\\Window\\System32\\Spool\\Printers.

RECENT FOLDER ARTIFACTS

The Recent Folder stores links of the recently accessed or opened files by a specific user. The folder can be accessed by using the following link: C:\\Users\\username\\AppData\\Roaming\\Microsoft\\Windows\\Recent.

WINDOWS FORENSICS- ANALYSIS OF WINDOWS ARTIFACTS

Analysis of Windows artifacts is the perhaps the most crucial and important step of the investigation process that requires attention to detail. The following flowchart depicts a typical windows artifact analysis for the collection of evidence.

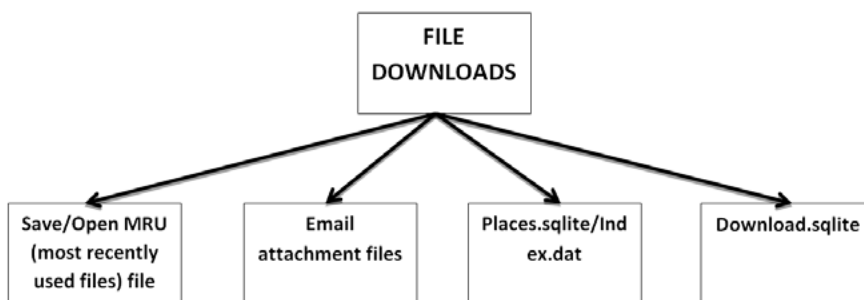


Figure 3. *File Downloads*

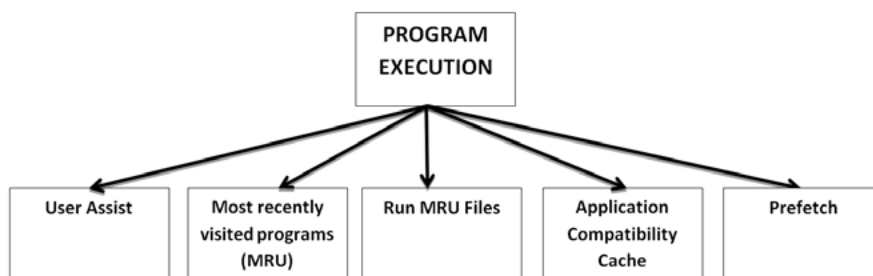


Figure 4. *Program Execution*

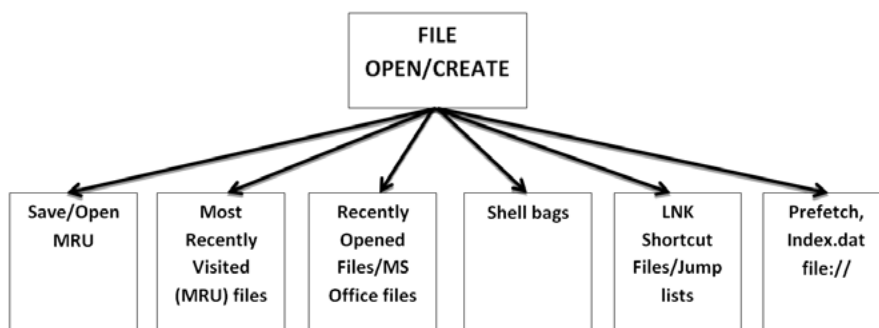


Figure 5. *Program Execution*

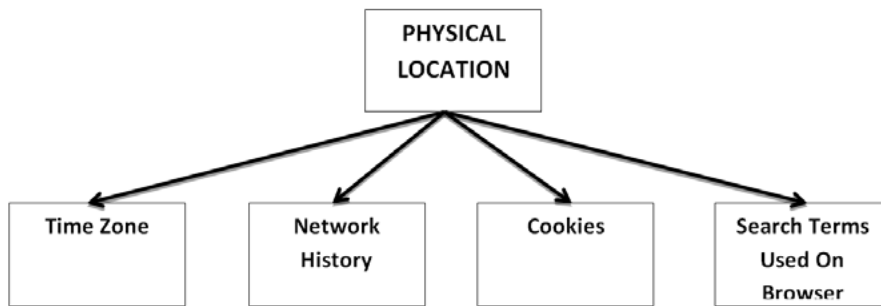


Figure 6. Physical Location Information

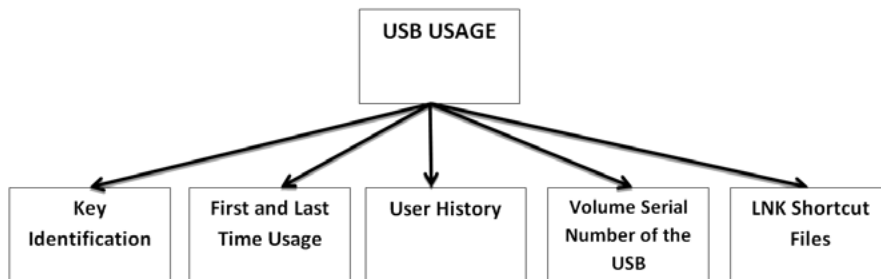


Figure 7. USB Usage Details

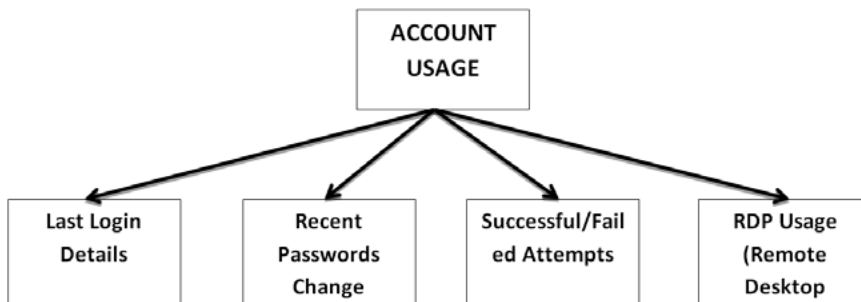


Figure 8. Account Usage Details

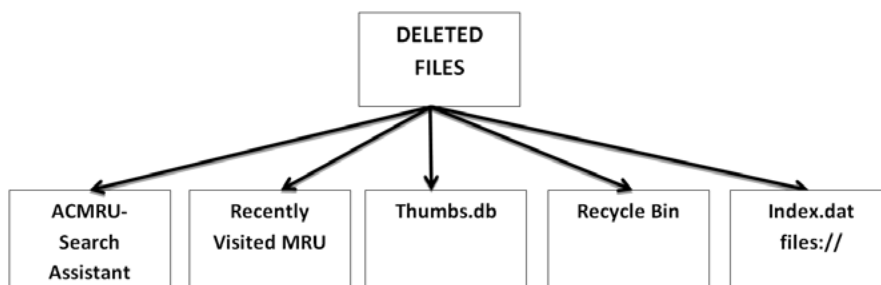
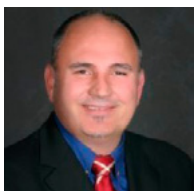


Figure 9. Deleted Files

ABOUT THE AUTHOR



Digital Forensic Expert Investigator Knowledge, training and experience in digital forensic investigations, electronic evidence discovery, data recovery and analysis, consulting and expert witness in criminal, civil and defense cases.

Over ten (10) years of progressive technical experience in designing, implementing, structuring, supporting, administrating, upgrading, documenting and maintaining networking environments.

Especially proficient in troubleshooting, audits, consulting, user support, customer relations, network topology, digital forensics, data recovery, backup strategies, designing, planning and implementation of network and wireless environments. Over ten (10) years in law enforcement.



**EXPERT DATA
FORENSICS**

**INVESTIGATORS OF
ELECTRONIC EVIDENCE**

Digital Forensic Lab Services

hard drives, computers, cell phones,
PDA's, memory chips and servers.

- investigation of electronic evidence
- forensic imaging
- custodians of digital data
- expert witness testimony
- forensic data recovery
- retrieval of deleted & lost data

Digital Forensic Investigation, Imaging & Recovery

Find us online:



Our company is dedicated to serving legal counsel, government, individuals and organizations. We forensically secure, analyze, recover and investigate digital data. We specialize in criminal, civil, and corporate matters. We provide Expert Witness services.

www.expertdataforensics.com

Tel: 702 435 8885; 888 355 3888





QuantumLeap

networking | security | consulting

Pescara

Via Colle Scorrano, 5
65100 Pescara
F. +39 0857992241
info@quantumleap.it

Roma

Piazza G. Marconi, 15
00144 Roma
T. +39 0632803612
F. +39 0632803283

www.quantumleap.it

UPDATE
NOW WITH
STIG
AUDITING

“IN SOME CASES
nipper studio
HAS VIRTUALLY
REMOVED
the **NEED FOR** a
MANUAL AUDIT”
CISCO SYSTEMS INC.

Titania's award winning Nipper Studio configuration auditing tool is helping security consultants and end-user organizations worldwide improve their network security. Its reports are more detailed than those typically produced by scanners, enabling you to maintain a higher level of vulnerability analysis in the intervals between penetration tests.

Now used in over 45 countries, Nipper Studio provides a thorough, fast & cost effective way to securely audit over 100 different types of network device. The NSA, FBI, DoD & U.S. Treasury already use it, so why not try it for free at www.titania.com



www.titania.com