Module 0x01: Forensics

Kevin ‘Hexstr Morgan’ Dickinson
Josh ‘HoboBeard’ Hammond
Christ ‘PossibleSloth’ Walz
• Week 1 Review
• Investigative Techniques
• Windows Forensics
• OSX and Linux Forensics
• Mobile (iOS) forensics
• Web Activity
• SQLite
• Anti-Forensics
• Anti-Anti-Forensics

Week 0x02: In-Depth
Week 0x02: In-Depth
• http://www.forensicfocus.com
• http://www.forensicswiki.org
• http://computer-forensics.sans.org/blog
• http://www.appleexaminer.com/
• http://hexstr-morgan.blogspot.com/
• http://tech.groups.yahoo.com/group/win4n6/messages

Leftovers: Resources
• **Backtrack 5**
  - [http://www.backtrack-linux.org/downloads](http://www.backtrack-linux.org/downloads)
  - Has a built-in ‘forensics’ mode for drive imaging
  - Includes a surprising amount of relevant tools

• **SANS Investigate Forensic Toolkit (SIFT)**
  - [http://computer-forensics.sans.org/community/downloads](http://computer-forensics.sans.org/community/downloads)
  - Project by Robert Lee of SANS, one of forensics’ most respected figures

• **AccessData FTK Imager**
  - Superb Windows/OSX/Linux imaging tool

---

**Getting Started**
Review: Week 1

- **Unallocated space**
  - VERY important - basis for much of forensics
  - Areas of a storage device not currently assigned to file(s) by the file system
  - Likely contains unspeakable things you thought were gone
  - Captured during physical imaging

- **Hashes**
  - Described as unique ‘digital fingerprint’, verify integrity
  - Important basis for much of forensics
  - Used to uniquely match individual files or exclude irrelevant (known) files
• **Imaging**
  • Duplicating some data source (e.g., hard disk, memory, etc.) for post-mortem review
  • Usually the first step in an investigation
  • Resulting evidence image encapsulated as a binary file (.001 or .E01) then fed into forensic tools (Encase, IEF, etc.) for review

• **Journaling**
  • Keeps track of changes committed to the file system in a atomic, circular fashion
  • Useful for recovering from unexpected shutdowns during file activity
• **FAT**
  • File Allocation Table
  • Resides at the beginning of a partition
  • Simple: files stored as a chain (singly-linked) list of clusters
  • Divided into equal-sized clusters, which made up of multiple sectors

Review: Week 1
• **File Carving**
  • Typically uses predictable file headers/signatures to independently identify files without the use of other (file system) information
    • Eg: JPEG file header: 0xFFD8 header, 0xFFD9 signature
  • Used for recovering deleted data or data that the file system has otherwise lost track of
  • Generally limited to contiguous clusters, cannot find data fragmented across a disk (with some exceptions)
  • **Demonstration** - hack.youtoo 2012 - FOR200 file carving
• **Automated tools**
  • No shame in using tools – the more expensive, the better...
  • …but you MUST still understand what’s going on under the hood
  • Use shell/cmd for everything and you will be unemployed
    • Stop being an insufferable nerd
  • **Bad example**: trying to write Facebook in notepad.exe
  • Serious uphill battle for forensic investigators – drives are getting bigger (4TB+) and cheaper every day
  • Programs are getting more complex and data protection is getting more sophisticated (eg: iOS) as privacy awareness grows
• **Timelines**
  • Complete chronological overview of a system’s activity
  • Gives an examiner a ‘big picture’ view of what’s happening on a system by trying together multiple artifact sources
  • Often used in incident response (IR) situations
  • Demonstation – Nullcon 2013 system timeline
• Previewing a system live
  • Different ways to accomplish this, unique issues w/ each
  • Sitting down in front of the system can provide a more insightful look into it
  • Restore image file to a spare drive
  • Build a VM from an image file

Investigative Techniques
• **Keyword Searching**
  • Highly effective, requires foresight
  • Lots of false positives and won’t give you much on it’s own but can provide interesting leads
  • Indexed – data processed upfront, instant search results
  • Search specific documents or across the entire drive
  • Can utilize highly-tuned expressions – often GREP-based
  • Like ‘strings’ except actually practical/useful
  • Works 90% of the time - unless data is encrypted, encoded or otherwise obfuscated
    • eg: Office 2007 documents (**Demo** – zip structure)
• **Hash Matching**
  • Utilize public (NIST NSRL) or private hash sets to quickly filter out large numbers of irrelevant files

• **Skintone Analysis**
  • Simple concept, analyses picture for relative percentage of skin-colored content
  • A few popular algorithms for quickly finding explicit content

• **Header/Signature Analysis**
  • Compares known file header/signatures to given extension
  • Renaming your .JPG to .MP3 won’t help, probably the opposite

**Investigative Techniques**
Nullcon 2013 – FOR400
Review
Windows Forensics: Registry

- Largest central source of artifacts on a Windows system
- Basis for much analysis – USB activity, shellbags, etc.
- Comprised of several different hives
  - ntuser.dat, contains user specific data - `%UserProfile%\ntuser.dat`
  - system, software, security - `%WinDir%\System32\config`
- Regripper
  - Developed by Harlan Carvey, strong automated tool for parsing
  - [http://code.google.com/p/winforensicaanalysis/downloads/list](http://code.google.com/p/winforensicaanalysis/downloads/list)
- Often employs ROT13 encoding for obfuscation purposes
• USB storage device history
  • `SYSTEM\ControlSet001\Enum\USBSTOR`
• Recently opened documents of various extensions
  • `SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs`
• Recent searches in Explorer
  • `SOFTWARE\Microsoft\Search Assistant\ACMru`
• Installed programs listed in Add/Remove software
  • `SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall`
• Network adapter information (IP, gateway, etc.)
  • `HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interface`
• Recently connected and ‘preferred’ Wireless networks
  • `HKLM\SOFTWARE\Microsoft\WZCSV\Parameters\Interfaces\GUID`
• Last 25 URLs typed into Internet Explorer
  • `HKCU\Software\Microsoft\Internet Explorer\TypedURLs`
• **System Restore** - Windows XP
  • Restore points - historical record of registry files
• **Volume Shadow (Copy) Service** – Windows Vista/7/8
  • Microsoft’s version of Time Machine
  • Extremely powerful
  • Technically complex, popular tools starting to parse this data
  • Restore zero-wiped (!) files
• Demonstation - Nullcon 2013 VSS Restore

Windows: SysRestore & VSS
**Extraction**
- HBGary Fastdump Pro (32 and 64-bit support)
- FTK Imager (also 32 and 64-bit support)
- WindowsScope (PCIE/ExpressCard)

**Crash Dumps**
- Similar artifacts to Hiberfil.sys (next page)
- Must be complete crash dump for use, eg: kernel and process memory

Windows: Physical Memory
• **Hiberfil.sys**
  • Dump of physical memory written to disk prior to entering hibernation mode
  • Complete snapshot of a live system - encryption keys, process info, network connections, etc.
  • Contents remain in unallocated space after resuming
  • Structure has been REed - can be fed straight into many popular memory analysis tools
• **Pagefile.sys**
  • Known as swapfile - virtual memory ‘overflow’ stored on disk
  • Can contain artifacts similar to Hiberfil.sys
  • Data stored in chunks of 4K or less (pagesize) – cannot carve files larger than this.

**Windows: Physical Memory**
• “Used by the Windows operating system to track user window viewing preferences. It does this by storing various Windows Explorer settings that relates to dimensions, settings, etc” – TZWorks
• Basically when you resize or otherwise give an Explorer window a custom view, This allows one to reopen the same folder at a later time with the settings from the previous time
• Comprised of subkeys in user registry hives
  • %UserProfile%\NTUSER.DAT
  • %UserProfile%\AppData\Local\Microsoft\Windows\UsrClass.dat
• Demonstration – TZWorks Shellbags Utility

Windows: Shellbags
• **INFO2 Records** - Windows XP
  • Predictable structure, can carve for
  • Original File Name
  • Original File Size
  • The Date And Time The File Was Deleted
  • The Files Unique Identifying Number In The Recycle Bin
  • The Drive Number That The File Came From

• **SI Records** – Windows Vista/7/8
  • 544 bytes long

**Windows: Recycle Bin**
• Correlates various artifacts from setup.api log, ntuser.dat, system and software registry hives
• Determine USB storage device was plugged into a system
  • Time(s) it was plugged in, unique device identifiers – serial number, VID, PID, volume letter
  • Key event during timeline analysis
  • Could assist in telling if a machine was compromised locally, combined with other timeline artifacts
• Woanware USBDeviceForensics [http://www.woanware.co.uk/?page_id=45](http://www.woanware.co.uk/?page_id=45)

Windows: USB Analysis
• Speeds up boot times and application startup
• Contains information regarding an executable’s path and when it was last run – important implications
• Leaves associated .pf entries on drive when a program is run
  • Stored in %WinDir%\Prefetch
  • ‘SHREDDER.EXE-12FA3910B.pf’
• Helpful for identifying file shredding, program install/uninstall events, etc.

Windows: Prefetch
• **LNK Files**
  • Also known as ‘shortcut’ files
  • Predictable file layout, can be carved for
  • Strong indication of when a file was last accessed
  • Internal dates, updated with value from file being opened each time it is accessed – separate source of timestamps

• **Thumbs.db**
  • Cached database of image thumbnails in a folder
  • Delete image(s) from a folder, still cached in thumbs.db
• **Plists**
  • Essentially OSX’s version of the registry
  • Decentralized, contain settings and historical data

• **Printing**
  • Translated as PDFs, can be carved from unallocated

• **USB device history**
  • Stored in kernel logs
  • Contains VID and PID entries, can be crossref’d against USB ID public repository for easy identification
• /var/log
  • Security logs, application logs, etc
  • Stays around for 4-5 weeks
• No file creation dates – interesting!
• /etc/shadow – MD5 hashes for users
• /etc/group – Group memberships
• /var/log/wtmp – User, source, time and duration of login
• $HOME/.ssh – hosts, public keys, private keys
• $HOME/.bash_history – history of shell, can be cleared
• Readahead – Linux equivalent of Windows prefetch files

Linux: General
• Crazy amount of data on personal devices
• Highly multifunctional - text messages, email, internet browsing, phone calls, GPS, camera, social media, etc…
• With you almost every moment of your existence, know exactly where you’ve been
• Popular smartphones use modified versions of popular desktop OSes, eg: Linux and OSX

Mobile Forensics
• **Cellebrite UFED**
  - Most popular tool, incredibly easy to use – ‘pushbutton forensics’, powerful analytics and reporting
  - Related, non-forensic version can be found in many wireless retailers
  - Phone capability matrix: [http://tinyurl.com/aphr69d](http://tinyurl.com/aphr69d)

• **Honorable Mentions and Other**
  - AccessData, Lantern, Oxygen, Elcomsoft, Paraben (lol)
  - Faraday bags/boxes/cages/etc. to block RF signals and prevent remote data destruction or tampering (remote wipe)

**Mobile Forensics: Tools**
Mobile Forensics: iOS

- **General**
  - Stripped down version of OSX
  - HFS+ file system, standard OSX folder structure

- **Security**
  - Devices feature hardware-assisted AES-256 disk encryption since iPhone 3GS (early industry adopter)
  - Multiple methods
    - Encrypted data partitions - protects data at rest (eg: chipoff)
    - Very complex and tightly integrated into OS
    - HFS Content protection
    - Data Protection - available as a feature since iOS4, application-specific
Mobile Forensics: iOS

• **HFS Protection**
  - Attempt to dd image rdisk0 – contents will be unreadable
  - Introduced in iOS4, supports specific per-file encryption. Each file gets a unique file key used to encrypt its data fork
  - “File keys are stored (wrapped) in an extended attribute named com.apple.system.cprotect” - [http://code.google.com/p/iphone-dataprotection](http://code.google.com/p/iphone-dataprotection)
  - Discourages traditional file carving techniques as unallocated data is now garbage
  - Discussions taking place regarding using HFS+ journal to carve
• **Physical Extractions**
  • Critical hardware flaw in A4-based devices (iPad 1, iPhone 4) that allows unsigned code execution and unrestricted access to the device
  • Most tools based on Limera1n exploit
  • Can be used to bruteforce passcodes - protection via **PBKDF2**
  • Forensically sound – no permanent modifications, uploads custom toolkit to temporary RAMdisk for imaging
  • Uses iOS’ own methods to decrypt protected data on the fly
  • A5+ devices may still be vulnerable
    • Exploit doesn’t start at bootrom, permanently modifies OS files - less forensically sound, legal “grey area”
• **Logical Extractions**
  - Utilizes iTunes backup service to generate a dump of unprotected parts of the device’s filesystem
  - Requires device to be unlocked, backups might be encrypted
  - Demonstation – Cellebrite Analyzer

• **Sogeti Labs**
  - Mobile RE specialists, frequent CTF competitors
  - [http://esec-lab.sogeti.com/post/Low-level-iOS-forensics](http://esec-lab.sogeti.com/post/Low-level-iOS-forensics)
• **Artifacts**
  • Utilizes sqlite .dbs for MANY of it’s functions
    • SMS, call log, browser history, user dictionary (?), etc.
  • Screen transition effects – litters unallocated with images (no longer particularly relevant)
  • User dictionary

• **Backups**
  • iCloud vs local
  • Local backups stored in %AppData%/Apple Computer/MobileSync/Backup
  • Local backups can be encrypted (recommended)
• **Android**
  • Most of the same things apply – sqlite DBs, utilizing exploits for physical extractions

• **Chipoff**
  • Advanced technique - physically removing (desoldering) device’s onboard storage (TSOP, BGA)
  • Acquire contents from damaged or otherwise inaccessible device – typically thwarted on onboard FDE
  • Requires logical reconstruction to properly arrange physical blocks and pages via Flash Translation Layer (FTL)

Mobile Forensics: Misc.
Mobile Forensics.
Misc.
• Popular tools - HstEx/NetAnalysis, CacheBack, IEF
  • Support all popular browsers
• Internet Explorer – index.dat
  • %AppData%\Local\Microsoft\Windows\Temporary Internet Files
• Firefox and Chrome – SQLite databases
  • %AppData%\ Mozilla\places.sqlite
  • %AppData%\Local\Google\Chrome\History
• Carve for database entries – predictable entry structure
• JSON fragments
  • Facebook, Twitter, Gmail/Gchat, Yahoo, etc.

Web Activity
• What is SQLite?
  • “SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine” – sqlite.org
  • It is public domain
  • SQLite is the most widely deployed database engine in the world
  • As previously discussed, used by Google Chrome, Mozilla Firefox, iOS (iPhone/iPad), Skype, etc.

SQLite Databases
• **File Format**
  
  • A SQLite file is divided into equal size pages
  
  • Types of pages include B-Tree page, overflow page, freelist page and locking page.
  
  • First page contains the database header: 0x53 0x51 0x4c 0x69 0x74 0x65 0x20 0x66 0x72 0x6d 0x61 0x74 0x33 0x00 or: “SQLite format 3”

<table>
<thead>
<tr>
<th>Offset (h)</th>
<th>00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000000000</td>
<td>53 51 4C 69 74 65 20 66 6F 72 6D 61 74 03 00 00</td>
</tr>
<tr>
<td>000000010</td>
<td>10 00 02 02 00 40 20 20 00 00 03 70 00 00 19 04</td>
</tr>
<tr>
<td>000000020</td>
<td>00 00 00 E3 00 00 19 1C 00 00 00 35 00 00 00 04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Offset (d)</th>
<th>00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000000000</td>
<td>53 51 4C 69 74 65 20 66 6F 72 6D 61 74 03 00 00</td>
</tr>
<tr>
<td>000000016</td>
<td>10 00 02 02 00 40 20 20 00 00 03 70 00 00 19 04</td>
</tr>
<tr>
<td>000000032</td>
<td>00 00 00 E3 00 00 19 1C 00 00 00 35 00 00 00 04</td>
</tr>
</tbody>
</table>
- **Number of Pages**

  ![Page Offset Table]

- **First freelist trunk page**

  ![First Freelist Table]

- **Number of freelist pages**

  ![Freelist Page Table]
• **B-Tree pages**
  • First byte is a flag
    • 0x02 – index B-tree internal node
    • 0x0A – index B-tree leaf node
    • 0x05 – index B-tree internal node
    • 0x0D – index B-tree leaf node
  • Header
    • Offset 1 – Byte offset of 1st block of free space
    • Offset 3 – Number of entries (cells) on the page
    • Offset 5 – Byte offset of 1st byte of cell
• Carving a database
  • Mostly want B-Tree pages, not necessarily stored contiguously.
  • Check first byte of pages for legal flag value: \texttt{0x02, 0x0A, 0x05, 0x0D}
  • Use header data as database fingerprint (tricky)

<table>
<thead>
<tr>
<th>Type</th>
<th>pg1 B-tree Page Header Offset 1</th>
<th>pg1 B-tree Page Header Offset 3</th>
<th>pg1 B-tree Page Header Offset 5</th>
<th>Page Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Chrome History</td>
<td>0F FC</td>
<td>00 15</td>
<td>05 06</td>
<td>4096</td>
</tr>
<tr>
<td>Google Chrome downloads.sq...</td>
<td>00 00</td>
<td>00 01</td>
<td>02 55</td>
<td>1024</td>
</tr>
<tr>
<td>Firefox places.sqlite</td>
<td>00 00</td>
<td>00 01</td>
<td>0F FB</td>
<td>4096</td>
</tr>
<tr>
<td>iPhone SMS.db</td>
<td>0F FC</td>
<td>00 11</td>
<td>02 45</td>
<td>4096</td>
</tr>
</tbody>
</table>
• **Full-Disk Encryption (FDE)**
  - If you don’t use it, you are dumb – so use it
  - OS passwords are useless: physical access to a machine without encryption = game over, drive is imaged, have everything
  - Plenty of good solutions available – no excuse not to use it
  - Hardware-based FDE built in to modern storage devices
    - Implementation varies (TPM, password), use with caution
  - Bitlocker – personal favorite, available on Windows 7 Ultimate and Windows 8 Pro
    - AES-256 w/ diffuser, strong TPM integration, multiple pre-boot authentication schemes (PIN, USB), AD integration
• **Full-Disk Encryption (cont.)**
  • Truecrypt on Linux, Windows and OSX
  • Filevault 2 on OSX
  • Not foolproof – still vulnerable to various implementation attacks eg: extracting encryption keys via DMA, cold-boot
  • **Disable** all unused ports in BIOS, particularly Firewire, PCMCIA/ExpressCard and Thunderbolt (sorry guys)
  • “Blocking the SBP-2 driver and Thunderbolt controllers to reduce 1394 DMA and Thunderbolt DMA threats to BitLocker”
    • [http://support.microsoft.com/kb/2516445](http://support.microsoft.com/kb/2516445)
    • Good Guy Microsoft
Anti-Forensics
• **Obfuscation**
  • Doesn’t work, don’t try it
  • Given sufficient motivation, we will find it

• **Steganography**
  • Useful and can be difficult to detect
  • Popular off-the-shelf tools (Encase, FTK) have near complete lack of steganalysis capabilities
  • Different popular file formats for hiding data
    • PNG – utilizes least significant bit (LSB). “The least significant bit i.e. the eighth bit inside an image is changed to a bit of the secret message”
    • MP3 – utilizes parity data in encoding process [http://www.petitcolas.net/fabien/steganography/mp3stego](http://www.petitcolas.net/fabien/steganography/mp3stego)

Anti-Forensics
• **Attacking forensic suites**
  • Good way to seriously piss off an examiner
  • You are probably screwed if things get to this point anyways
  • Circular references (eg: ‘42.zip’ zipbomb)
    • [http://newsgroups.derkeiler.com/Archive/Alt/alt.privacy/2008-04/msg00221.html](http://newsgroups.derkeiler.com/Archive/Alt/alt.privacy/2008-04/msg00221.html)
  • Fuzzing program’s parsers and internal viewers
    • Derbycon, [http://www.youtube.com/watch?v=-HK1JHR7LIM](http://www.youtube.com/watch?v=-HK1JHR7LIM)
  • NSRL ‘scrubbing’ to prevent hash filtering

• **Data destruction and sanitization**
  • Fill a recording media with junk data
  • `hexstr@here /#: dd if=dev/zero of=/dev/sda bs=512`
  • Some debate regarding necessity of multiple write passes, eg: [http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html](http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html)
  • Advanced physical (SEM/MFM/etc.) recovery techniques are highly cost-prohibitive and not practical, eg: restricted to intelligence agencies

---

**Anti-Forensics**
**Timestamp Manipulation**
- **Timestomp** - popular tool, included in Backtrack
- Targeted or blanketed - important technique for throwing off investigators and covering tracks
- Two sets of entries in NTFS MFT
  - $STANDARD_INFORMATION and $FILE_NAME
  - Only $STANDARD_INFORMATION is readily tamperable

**Encrypted VMs**
- Have experimented with it, not readily supported
- Probably not a good idea as memory is written to disk

**Mobile devices**
- Use a strong passcode (alphanumeric, 7+ digits)
- Do not leave device unlocked for prolonged periods
- Ensure your backups are in a secure location

Anti-Forensics
• **Breaking Encryption**
  • ‘Cold boot attack’ - volatile contents (encryption keys) stay in memory for a period of time, can be prolonged under controlled conditions
  • Extract physical memory via DMA-based attack – 1394a/b, Thunderbolt, PCI-E, ExpressCard, etc.)
  • Put system in hibernation mode, extract hibernation file (ineffective when system (OS) drive is encrypted)
  • ‘Evil Maid’ attack – bootloader poisoning
  • Attack crypto implementation – often weakest link, eg: AES ECB mode
    • Good old fashioned keyloggin’
• **Physical attacks**
  • “All bets are off”
  • Joint test action group (JTAG) ports – universal debugging interface for integrated circuits

• **Timestamp manipulation detection**
  • Use tools to compare $FILE_NAME and $STANDARD_INFORMATION MFT entries for disrepancies (Log2Timeline does this)

**Anti -Anti- Forensics**
References

- http://www.sqlite.org/fileformat2.html
- http://www.deer-run.com/~hal/LinuxForensicsForNon-LinuxFolks.pdf