Digital Forensics
Spring 2020
Weeks 1 - 10

Contents

In order of Weeks

Week 1 - Intro to Digital Forensics

Introduction
Digital Forensics - a branch of forensic science encompassing the recovery and investigation of material found in digital devices, often in relation to computer crime.

Forms of Digital Forensics
- *Forensic Analysis*
    - Evidence is recovered to support or oppose a hypothesis before a criminal court
- *eDiscovery*
    - A form of discovery related to civil litigation
- *Intrusion Detection*
    - A specialist investigation into the nature and extent of an unauthorized network infiltration/intrusion.

ISO 27037 - modern day standard for Digital Forensics
This International Standard ensures that responsible individuals manage potential digital evidence in practical ways that are acceptable worldwide, with the objective to facilitate investigation involving digital devices and digital evidence in a systematic and impartial manner while preserving its *integrity and authenticity.*

- International Digital Forensics Standard
- Ratified October 2012
- ICAP - Guidelines for the:
    - *identification*
        - Process involving the search for, recognition and documentation of potential digital evidence.
    - *collection*
        - Process of gathering the physical items that contain potential digital evidence.
    - *acquisition*
        - Process of creating a copy of data within a defined set. The product of an acquisition is a potential digital evidence copy.
    - *preservation of digital evidence.*
        - A process to maintain and safeguard the integrity and/or original condition of the potential digital evidence.
- Agencies should develop their policies and procedures in accord with this standard, so multinational cases can proceed unhindered.

Training required by ISO 27037
- *Digital Evidence First Responder (DEFR)*
    - Has the skill and training to arrive on an incident scene, assess the situation and take precautions to acquire and preserve evidence.
    - When necessary - call for specialised equipment, or a specialist.
- *Digital Evidence Specialist (DES)*
    - Has the skill to analyse the data and determine when another specialist should be called in to assist with the analysis.
        - Can be a government agency, officer of the law or an IT support specialist.

Devices and Media under the scope of ISO 27037
- Digital storage media used in standard computers like hard drives, floppy disks, optical and magneto optical disks, data devices with similar functions
- Mobile phones, *Personal Digital Assistants (PDAs)*, *Personal Electronic Devices (PEDs)*, memory cards
- Mobile navigation systems
- Digital still and video cameras (including CCTV)
- Standard computer with network connections
- Networks based on TCP/IP and other digital protocols
- Devices with similar functions as indicated above.

Some ISO 27037 key terms

- *Spoilation*
    - Act of making or allowing change(s) to the potential digital evidence that diminishes its evidential value. This will render it inadmissible in a judicial court.
- *Validation*
    - Confirmation, through the provision of objective proof, that the requirements for a specific intended use or application have been fulfilled.
- *Volatile Data*
    - Data that is especially prone to change and can be easily modified.
        - Changes can include:
            - Switching the power off
            - Passing through a magnetic field.
        - Volatile data also includes data that changes as the system state changes.
            - Data stored in RAM
            - Dynamic IP addresses
- *Verification Function*
    - Function which is used to verify that two sets of data are identical.
        - Verification functions are commonly implemented using hash functions such as MD5, SHA1, etc.

Types of Forensics

Industrial Actions

- Failure to comply with employment guidelines

Civil Actions

- Clandestine business operations
- Operating a company while at work
- Divorce proceedings

Criminal Actions

- Using a device to commit a crime
- Stealing the device

Intrusion by Malware

- Remote attacks propagated by the use of malware.

Steps in responding to a Security Breach (i.e. hacking)

- Investigation team takes action, where their aim is to provide Computing Security.
    - *Testing and verifying* the integrity of workstations and servers
    - *Looking for vulnerabilities* using Penetration Testing
    - *Identifying attacks* using Firewall Logs. These logs can also be sourced from:
        - Servers
        - Routers
        - End-user devices

The Investigations Triad

- Vulnerability/Threat Assessment and Risk Management
- Digital Investigations
- Network Intrusion Detection and Incident Response

Evidence Collection - rfc3227

- A *security incident* is a security-relevant system event in which the system's security policy is breached.
- *rfc3227* provides guidelines on the collection and archiving of evidence relevant to such a security incident.
- Unless the evidence collection is done correctly, the evidence can be thrown out in court.

Guiding Principles during Evidence Collection

- Adhere to your site's Security Policy and engage the appropriate Incident Handling/Law Enforcement Personnel.
- Capture as accurate a picture of the system as possible.
- Keep detailed notes:
    - Dates and times
    - If possible, generate an automatic transcript
        - E.g. on Unix systems, the 'script' program can be used, however the output file it generates should not be to media that is part of the evidence.
    - Notes and print-outs should be signed and dated.
- Note the difference between the system clock and UTC.
    - For each time stamp provided, indicate whether UTC or local time is used.
- Be prepared to testify (perhaps years later) outlining all actions you took and at what times.
- Minimise changes to the data as you are collecting it.
- Remove external avenues for change.
- Collection first, analysis later.
- Proceed from the volatile to the less volatile. Order of Volatility discussed in detail below.

Order of Volatility for a Typical System

From the volatile to the least volatile:

1. Registers, cache
2. Routing table, arp cache, process table, kernel statistics, memory.
3. Temporary file systems
4. Disk
5. Remote logging and monitoring data that is relevant to the system in question
6. Physical configuration, network topology
7. Archival media

Legal Considerations

Computer evidence needs to be:

- *Admissible*
    - Must conform to certain legal rules before it can be put before a court. Spoiled evidence is the opposite of this.
- *Authentic*
    - Must be possible to positively tie evidentiary material to the incident.
- *Complete*
    - Must tell the whole story and not just a particular perspective.
- *Reliable*
    - There must be nothing about how the evidence was collected and subsequently handled that casts doubt about its authenticity and veracity.
- *Believable*
    - It must be readily believable and understandable by a court.

Chain of Custody

The following need to be documented:

- Where, when and by whom:
    - was the evidence discovered and collected
    - was the evidence handled or examined.
- Who had custody of the evidence, during what period. How was it stored?
- When the evidence changed custody, when and how did the transfer occur (include shipping numbers, etc.)

Tool Philosophy

It is necessary to use tools with a small footprint to minimise evidence disruption - in order for the evidence to be deemed admissible in a court of law.

Tools that can provide a .txt output for later analysis and reporting are preferred.

For all these reasons, *command line tools* are preferred.

Week 2 - The Forensics Case

The Forensic Process
A suspicious item is found.

- *What* is the item?
- *How* did it get there?
- *When* was it placed there?
- *Who* put it there?
- *Why* was it placed there?

Is it forensic evidence?

Branches of Digital Forensics

- Computer Forensics
- Network Forensics
- Database Forensics
- Mobile Device Forensics

The Three Security Teams
Vulnerability, Threat Assessment and Risk Management

- Penetration Testing

Network Intrusion Detection and Incident Response

- Automatic monitoring of Firewall and IDS logs

Digital Investigations

- Forensic analysis of systems suspected of containing evidence
- Initiate the legal process as follows:
    - Allegation or complaint
    - Investigation
    - Case building
    - Trial

Digital Evidence examples

- Was the device used to commit a crime?
    - Sexploitation of minors
    - Communication of drug deals and their financial records
- Was it simple trespass?
    - I.e. looking inside another PC using SSH
- Was it theft and/or vandalism?
- Were a person's rights infringed?
    - Cyberstalking or social media harassment

Civil Cases

- Examples include:
    - Email harassment
    - Falsification of data
    - Discrimination
    - Embezzlement
    - Sabotage
    - Espionage
- The business needs to continue operating while the investigation proceeds.
- The primary aim is to stop any intrusion and minimise further losses/possible litigation.

Policies
The best way to reduce the risk of a civil case is to set up and enforce strong policies that are easy to read and follow.

- The main policy is for the *Acceptable Use* of the company's devices and networks.

- Published policies provide a line of authority for conducting an internal investigation.
- They state who has the right to initiate an investigation, take possession of evidence and access such evidence.

Difference between Live and Disk Forensics
It is suspected that a device is involved in an attack. How can this be confirmed?

Live Forensics
- The device is live and the attack is current or very recent.
- You want to capture live evidence before you power the device down.

Disk Forensics (post mortem)
- Device is powered down, attack is over.
- You want to examine permanent disk or USB storage for traces of the attack.

Lifespan of Data

Registers, peripheral memory, caches, etc.	nanoseconds
Main Memory	nanoseconds
Network state	milliseconds
Running processes	seconds
Disk	minutes
USBs, backup media, etc.	years
CD-ROMs, printouts, etc.	tens of years

The Forensic Method
Obtain Authority to Search
- This may be a Search Warrant.

Secure and Isolate
- Locate removable media
- Secure mobile devices (Faraday Bag/Cage)

Record the Scene
- Document and Photograph

Conduct a Systematic Search for Evidence
- Order of Volatility

Assess the Suspect
- Assess the risk of the suspect having the ability to hide or destroy evidence.

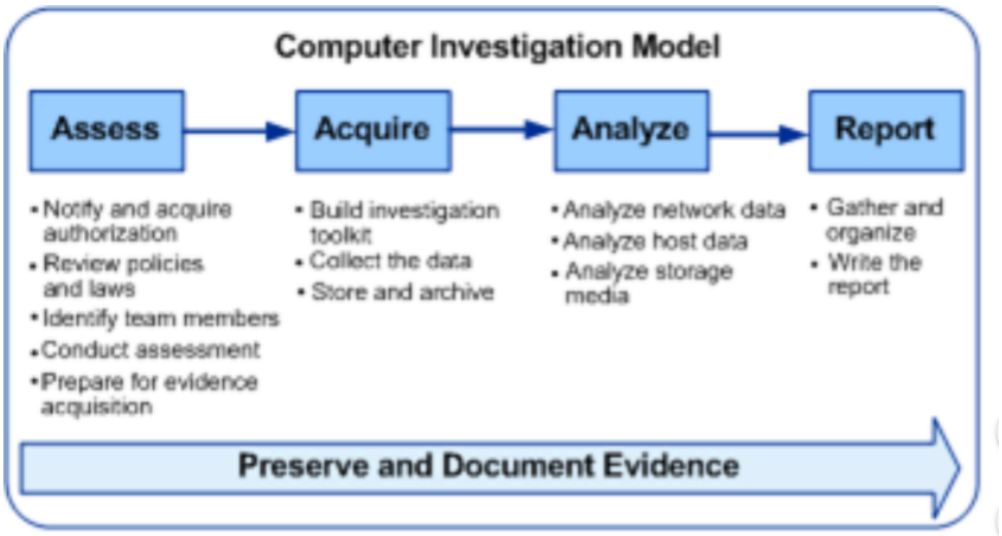
Collect and Package evidence
- Maintain a chain of custody

Evidence Analysis
- Analyse the evidence in a forensic lab.

Submission
- Submit the evidence as an *expert witness*.
    - An expert is allowed to give an opinion to the court.

Be prepared to have your methods challenged.

Main Objective: Assess, Acquire, Analyse and Report.



Forensics Principles

Aims of Forensics

- To gather admissible evidence legally and without interfering with business purposes.
- To gather evidence targeting the potential crimes and disputes that may adversely impact an organisation.
- To allow an investigation to proceed at a cost in proportion to the incident.
- To minimise interruption to the business from any investigation.
- To ensure that evidence makes a positive impact on the outcome of any legal action.

Corroboration

While one example of class evidence is not compelling, several *independent* class examples together can build a compelling case.

- I.e. a threatening letter may have also been printed by an Epson printer - and the suspect may also happen to possess an Epson printer.

Forensic Soundness

The methods used to obtain evidence must not alter the evidence.

- E.g. the act of reading a disk file will alter the time of last access stored within the file.

Similarly, the act of accessing memory will alter that memory. However, some minor alterations are inevitable and can be accepted by precedent.

- The processes used to obtain evidence must be well documented to identify possible changes.

Authentication

- Identifying the source of evidence
    - Human *and* digital device
    - One does not imply the other
- This can involve:
    - Oral evidence (a suspect identifies his laptop)
    - Circumstantial evidence
    - Digital evidence (a private encryption key is compelling)

Attribution

Liability is extended to a defendant who did not actually commit the criminal act.

- Asserting that the evidence found on a device can be attributed to one and only one person.

Examples:

- A web history file may contain web searches for “axe murderer”
- A wireshark packet capture may indicate visits to a child pornography website
- A web server apache2 log may indicate visits from the suspect.

We need to assert only that the suspect did the deed. We rely on:

- Authentication (logon passwords)
- DHCP logs for linking MAC addresses to IP addresses
- Gateway router logs for linking public IP addresses to private IP addresses (NAT)
- Phone GPS tracking (google maps)
- Syslog remote logging (and auth.log on Linux)
- *Net user* commands to find login timestamps
- Linux *last* command for logon details

Objectivity

Investigators should be free from bias when investigating. Use of judgemental language may harm an investigator’s soundness and reputation.

Repeatability

- The scientific method requires evidence to be able to be independently verified.
- The second investigator will need to be able to follow your documentation.
- In particular, the name and version of all tools used MUST be documented.

Evidence Exchange

Locard’s Exchange Principle

- Contact between two items will result in an exchange:
    - Between the suspect and the victim
    - Between the investigator and the crime scene.
- The exchange can be:
    - *Physical* (fingerprints)
    - *Digital* (an email)

Examples of Locard’s Principle

- In a computer intrusion, the attacker may leave evidence in disk space, log files and the Windows Registry.
- The act of sending an email may leave traces on the sender’s hard disk, complete with time stamps.

Evidence Integrity

Basics

- There is a need to confirm that the evidence has not been altered *after* collection.
- Most evidence is kept as disk files, so confirming the evidence’s integrity is done by *hashing* the files to get a *digital fingerprint* when the evidence is collected.
- Any copy of the evidence file used for forensics can be hashed again.
- The hash of the copy should match the hash of the original.

Forensic Acquisition

Working on a disk may require minor alterations to its contents.

- You must prove that these alterations are minor.

Thus, it is best to work on a copy of the disk.

- The copy can be to another, similar disk.
- Or the copy can be to an *image file*.
- The image file can be *raw*, or in a *forensic container*.
- You can also acquire the contents of the device’s RAM.

Evidence Characteristics

- Evidence traces can have *class* characteristics or *individual* characteristics.

- Class characteristics apply to many cases.
    - E.g. a threatening letter was written in MS Word version 2007. A copy of Word 2007 was found on the suspect's laptop.
- Individual characteristics apply to one case.
    - E.g. each copy of Photoshop embeds its serial number in every image produced.

Chain of Custody

- Chain of Custody forms are used to log when, where and why evidence was transferred.
    - Technique helps to minimise loss or contamination of evidence.

Levels of Certainty

- *C0 - Evidence contradicts the known facts*
    - Incorrect
- *C1 - Evidence is highly questionable*
    - Highly uncertain
- *C2 - Only one source of evidence which is not protected against tampering*
    - Somewhat uncertain
- *C3 - Some tamper protection, some inconsistencies*
    - Possible
- *C4 - Evidence is tamper proof, or there are multiple independent sources of evidence that agree*
    - Probable
- *C5 - Tamper proof evidence from several independent sources that agree, some minor uncertainties (loss of data, timing uncertainties)*
    - Almost certain
- *C6 - Tamper proof evidence with a high statistical probability*
    - Certain

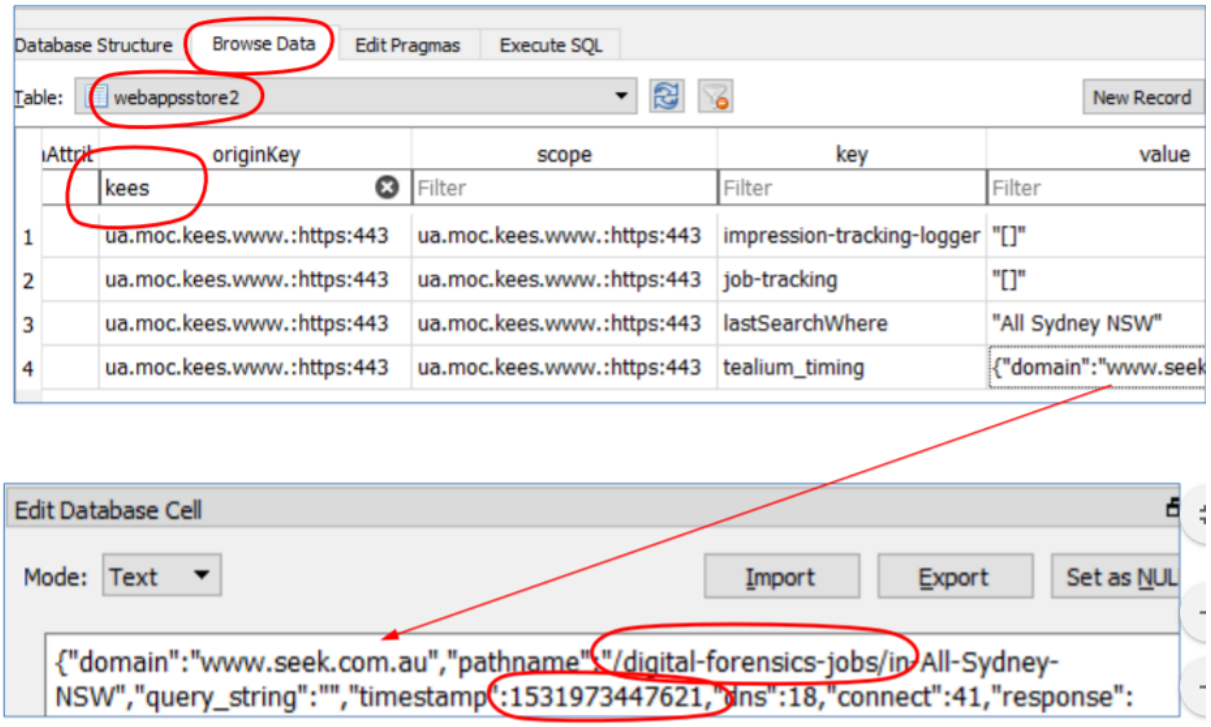
Week 3 - Web Browsers/Web Tracking and Browser History

Using Web Browsers for Forensics

A classic Browser Profile (i.e. Firefox)

- Bookmarks, Downloads and Browsing History
  - The *places.sqlite* file contains all your Firefox bookmarks and lists of all the files downloaded, as well as the websites visited.
- Passwords
  - Passwords are stored in the *key4.db* and *logins.json* files.
- Site-specific preferences
- Search engines
- Personal dictionary
- Autocomplete history
  - The *formhistory.sqlite* file remembers what you have searched for.
- Cookies
  - Cookies are all stored in the *cookies.sqlite* file.
- DOM storage
  - Designed to provide a larger, more secure, and easier-to-use alternative to storing information in cookies. Information is stored in the *webappsstore.sqlite* file for websites.
- Extensions
  - The extensions folder, if it exists, stores files for any extensions you have installed.
- Stored session
  - The *sessionstore.jsonlz4* file stores the currently open tabs and windows.

Web Forms in Firefox



Identifying a Web Client

We are given a packet capture file (.pcap). We are then told to look for forensic evidence.

Preferably before examining the .pcap file, the first step is to identify the Web Client and the OS.

- One way to do this is to use the browser to access a special device fingerprinting website (after acquiring and saving an image of the device).
- The *http* request string is informative.

- The detail may identify a suspect's PC
    - Even with inprivate browsing

This entire process is called *device fingerprinting.*

We must access the PC to confirm the identity.

Sites specialising in device fingerprinting:

- https://www.browserleaks.com/
- https://panopticlick.eff.org/browser-uniqueness

Web Server Tools

- Tracking software
    - Linked to Social Media and Search Engines
    - Analysis by Builtwith.com
- Analysis sites
    - W3Techs.com
- Detailed logs of visitors
    - Often Apache2
    - /var/log/apache2/access.log

Browser Characteristic	bits of identifying information
User Agent	10.14
HTTP_ACCEPT Headers	9.55
Browser Plugin Details	15.38
Time Zone	7.15
Screen Size and Color Depth	4.5
System Fonts	19.08
Are Cookies Enabled?	0.43
Limited supercookie test	0.96

Analysis of uts.edu.au by Builtwith
Analysis via w3techs.com

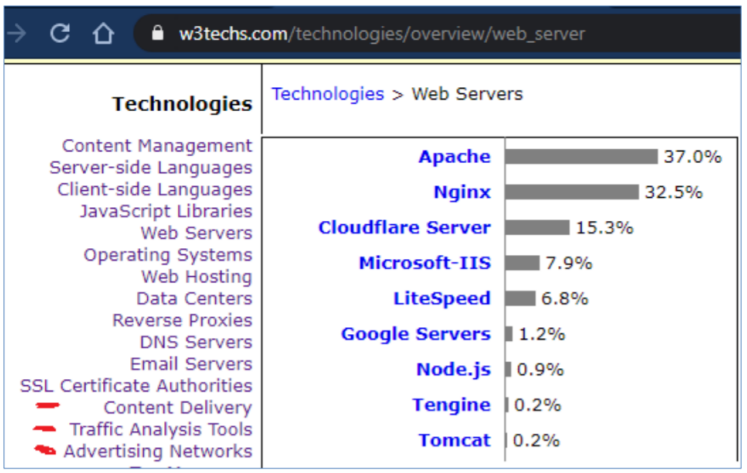
Analytics and Tracking
View Global Trends
iGoDigital

Audience Measurement
CrazyEgg

Site Optimization
Google Optimize 360
New Relic

Application Performance
Google Analytics
DoubleClick Floodlight

Conversion Optimization
Google AdWords Conversion
Facebook Signal
Facebook Pixel
LinkedIn Insights
Baidu Analytics



Using User Tracking for Forensics

Basics

A web server needs to track a web client by:

- by IP address
- by the HTTP referrer tag
- by a cookie saved on the target
    - Http cookie, web cookie, browser cookie
    - Cookies have gone out of fashion as insecure.
- by embedded code on the web page
    - Tracking on the website
    - Using third parties to do remote tracking

Web Analytics

- Web Page Tracking is also used by Advertisers.

The *user profile* is an important marketing tool.

User tracking for Profit

- If corporations analyse how users came across their websites and purchase their items, they will be able to entice many more to follow.
- By combining your visits to many websites many times, the analytics company will be able to understand and influence your future behaviour.

These results can be used for forensics.

Journey Mapping

Following the target as they navigate the Internet and end up purchasing an item.

- Visits to social media and search engines are *converted* into a visit to the website.
- A timeline is built based on the time stamp of every stage.
- The person is called the *actor*.
- The *scenario* identifies the expectations of the actor.
- *Opportunities* arise to entice further purchases.

Google Analytics

Two methods using two different JavaScript libraries

- *ga.js* drops a set of _utm cookies.
- *analytics.js* drops two cookies:
    - *_ga*
        - Lifetime of 2 years
    - *_gid*
        - Lifetime of 1 day

UTM Cookie Formats

- Google bought Urchin Software in 2005.
- *Google Analytics (GA)* uses UTM.
    - *UTMA* tracks dates and visits
    - *UTMB/C* indicate session expired
    - *UTMZ* is for tracking the user
        - Referer, keyword, ad campaign, etc.
    -

UTMA - The Visitor Identifier

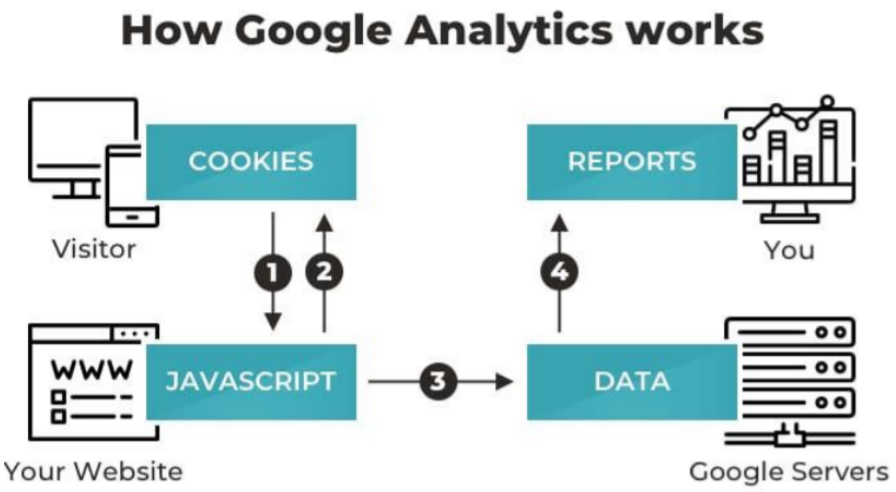
UTMB - 30 Minute session identifier

UTMC - On Exit session identifier

UTMV - Custom Variable Cookie

UTMZ - Visitor segmentation

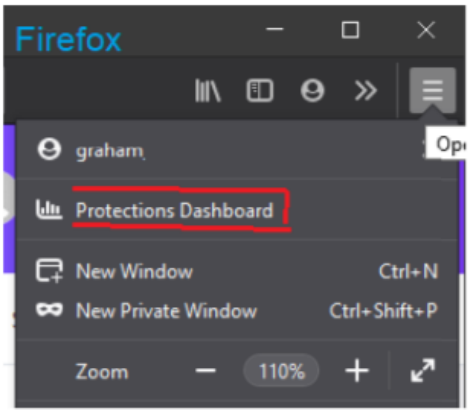
GA data flow



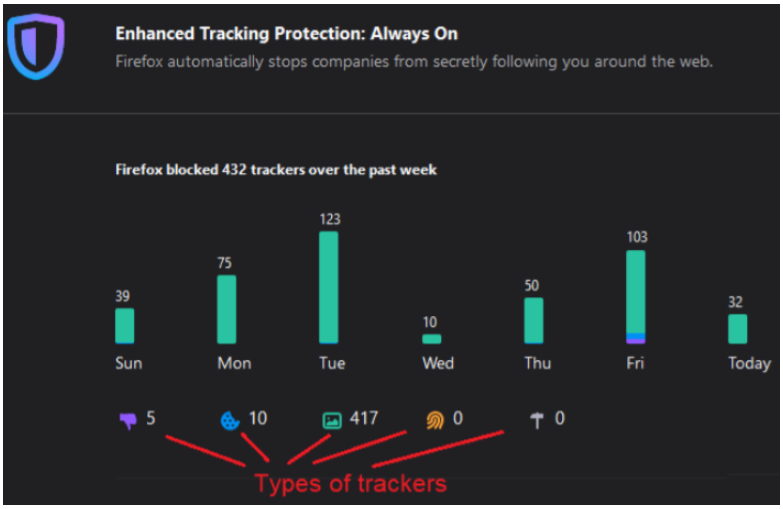
Mitigating Tracking - Firefox Protection Dashboard

Firefox can block trackers with its Protections Dashboard.

- Provides tracking protection for a desktop or mobile device.
- Provides analytics on the blocked trackers.
- Uses an addon from *Disconnect* - https://disconnect.me/



Tracker Types



- *Tracker 1 - Social Media*
  - Social networks place these trackers on other websites to follow users.
- *Tracker 2 - Cross Site Tracking Cookies*
  - Analytics companies place these trackers to follow users.
- *Tracker 3 - Tracking Content*
  - The site loads external content targeted at you.
- *Tracker 4 - Fingerprints*
  - Collects browser settings to identify you.
- *Tracker 5 - Cryptominers*
  - Uses your device to mine money.

Locating and Examining Cookie Files

Introduction - Reason for Cookies

Web Pages are transferred over the Internet using *HTTP (Hyper Text Transfer Protocol).*

HTTP is stateless - so a method of saving viewer choices is needed.

- Cookies save state on the client as a file on disk.
- Cookies are small and fast (lightweight), so they can resist Denial of Service attacks.
- Cookies are also used to save state for session key negotiation (Wireless and VPNs).

More Reasons for Cookies

Personalisation

The server remembers what you liked on your last visit.

Data Capture

The server remembers what you asked for.

Sales tracking using a shopping basket

Sales and transactions made, hence, cannot be undone.

Authentication

No need for a password for a repeated login.

Cookie Forensics

- Deleting cookies will disable many websites.
- Modern web sites only drop very basic cookies.
- Viewing cookies is a useful forensic tool, as investigators can piece together:
    - Websites visited
    - Actions taken/pages visited
    - Date of first visit
    - Date of last visit
    - Number of visits

Setting Cookies

1. The web client asks for a web page using HTTP.
    - *GET /index.html HTTP/1.1*
2. The web server *sets a cookie* when it replies
    - *HTTP/1.1 200 OK*
    - *Set-Cookie: name=value*
3. The cookie is returned each time the page is accessed.
4. The server keeps a log of cookies to track viewers.
    - *viewer = ip address + referer + cookie*

Setting Cookies - Backend process

1. Server has code to set the cookie

```
<?php
 $expire=time()+60*60*24*90;
 setcookie("user", "CEH Student",
$expire);
 ?>
```

2. Browser asks for the server page

```
GET /logon_p.php HTTP/1.1
Accept: text/html, application/xhtml
Accept-Language: en-AU,en-GB;q=0.8,e
User-Agent: Mozilla/5.0 (compatible;
Accept-Encoding: gzip, deflate
Host: 10.10.10.38:8080
Connection: Keep-Alive
Cookie: user=CEH+Student
```

3. Server sets the cookie

```
HTTP/1.1 200 OK
Date: Sun, 21 Apr 2013 20:48:10 GMT
Server: Apache/2.2.14 (Ubuntu)
X-Powered-By: PHP/5.3.2-1ubuntu4.18
Set-Cookie: user=CEH+Student; expire
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 317
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
```

4. Cookie file appears on the client's PC. Dates are in cookie format.

```
userCEH+Student10.10.10.38/153619076958723031181837412819403029 3839*
```

Cookie Types

Session cookie

No expiry date, deleted by the browser when the session ends

Persistent cookies (tracking cookies)

Expiry date in the future

Secure cookie

Sent encrypted using HTTPS

Third party cookies

Set from a different URL domain. User details are sent to the third party, and used for marketing.

- *Example* - advertisements
- Pay-Per-Click business model.
- *Can be stopped by using InPrivate/Incognito filtering.*

Google 3rd party cookies do not usually relate to the page visited. They are designed to encourage you to buy an unrelated product.

- adwords.com
- gstatic.com
- googleadservice.com

Cookie example - GA_gid

GA 1.3.1180290148.1531954273
GA = Google Analytics
1.3 = Version
1180290148 = random number session ID
1531954273 = Unix timestamp (visit)

More cookie data examples

_ga cookie

- *GA1.3.0164af9713b500100f85468f61190004e00a500d00bd0*
- SHA Hash

GUID

- Globally Unique ID
- *3b83fca5-3223-342c-459f-64e0fcf78633*
- 5 parts

%Encoded

- To treat control characters as plain text
- *%5B%5B%27SEM-GGL-SRC-FY16Q3-5463%27%2C%271531954291555%27%5D%5D*
- *[['SEM-GGL-SRC-FY16Q3-5463','1531954291555']]*

Blocking Cookies

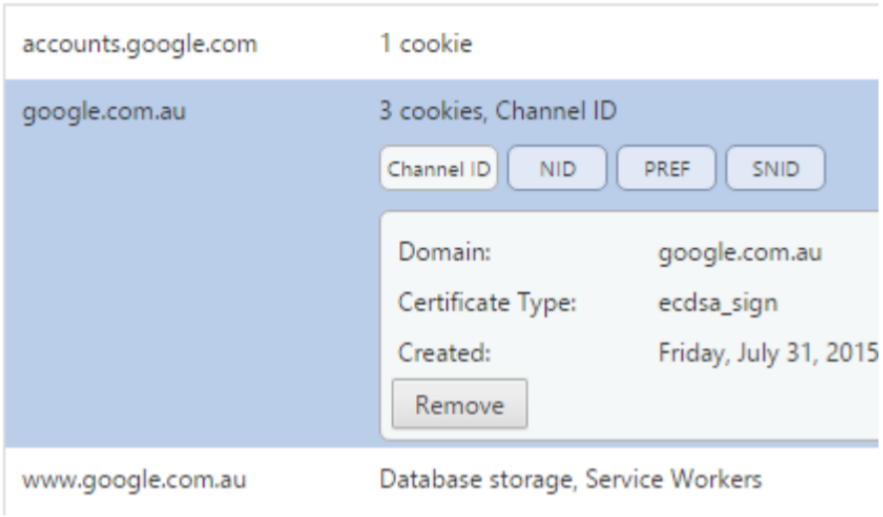
Browsers have add-ons that block certain cookies (i.e. *Ghostery* on Firefox.)

These block advertisements and stop data tracking to preserve privacy, with an in built viewer to show ad/tracking activity.

Storing Cookies

Each browser maintains its store of cookies separately. Cookies are saved in a compressed format for speed.

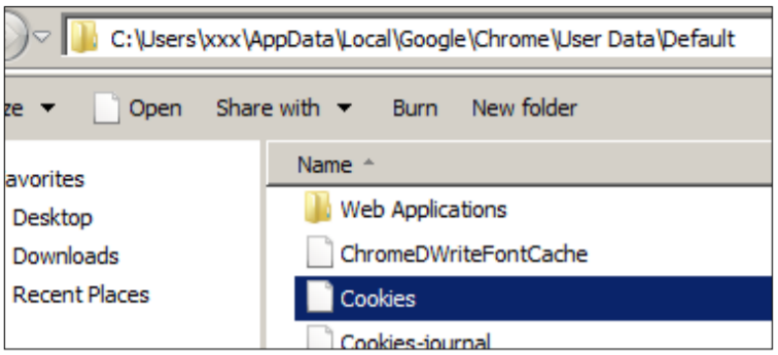
- We can use the browser *cookie manager* to view its cookies.
- Each user has a separate cookie store.



The Locally Stored Chrome Cookie File

- Keeps a list of cookies from all sites visited.
- Can see the site name in plain text.
- The user can delete these from Chrome settings.

Filepath: *C:\Users\<User>\AppData\Local\Google\Chrome\User Data\Default*



Viewing the Chrome cookie file - with *find*

- Copy the *Cookies file* to your *Windows* work folder
- Filter by your search term using *find*

```
C:\Users\graha>find "seek" C:\transfers\Cookies

---------- C:\TRANSFERS\COOKIES
..¤°▒¼▒.seek.com.au_gat_tealium_0/
.¤ý£ø4u.seek.com.au_gac_UA-63897908-1/
.¤┐OúÍ®.seek.com.au_ga/
.¤┐O▒nu.seek.com.aus_cc/
.¤°{Ñ■.seek.com.aumain/
.¤°▯■z.seek.com.au_gat_tealiumga/
.▒.¤ý£ø/¬.seek.com.au_gid/
```

Viewing the Chrome cookie file - with *grep*

- Copy the *Cookies file* to your *Linux* work folder
- Pull out the *ASCII* using *strings*
- Filter by your search term using *grep*

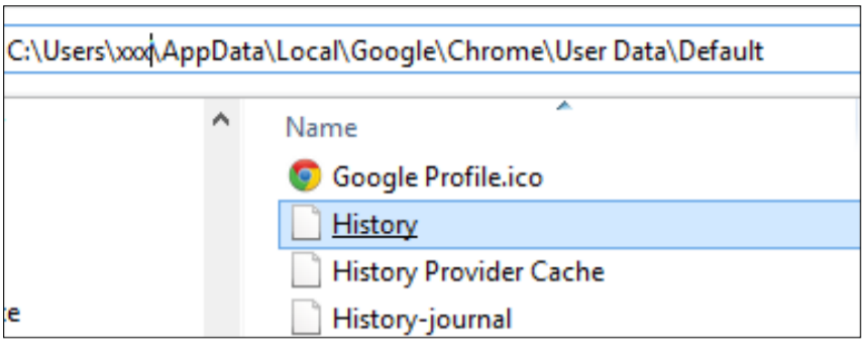
```
group11~$ strings  /mnt/c/transfers/Cookies | grep seek
.seek.com.au_gat_tealium_0/
4u.seek.com.au_gac_UA-63897908-1/
.seek.com.au_ga/
nu.seek.com.aus_cc/
.seek.com.aumain/
U.seek.com.aus_ev59/
z.seek.com.au_gat_tealiumga/
.seek.com.au_gid/
```

Locating and Examining Browser History

Basics

- The browser stores the history of visited pages.
- This is usually large (MB) so cannot read directly.
- The browser has methods of deleting web history.

Filepath: *C:\Users\<user>\AppData\Local\Google\Chrome\User Data\Default*



Viewing the Chrome web history file

- Using Linux (WSL for example)
- Copy the *history file* to your work folder
- Pull out the ascii using *strings*
- Filter by your search term using *grep*

```
C:\Forensics>strings history | grep hostworks
https://www.google.com.au/search?q=abn&oq=abn&aqs=chrome.
filetype:pdf+hostworks
filetype:pdf hostworks
filetype:pdf hostworks
site:seek.com.au hostworks
```

Chrome Top Sites
Records the most visited sites, and can indicate a suspect's interest.

```
C:\Forensics>strings.exe "Top Sites" | grep seek
http://seek.com.au/t&
http://seek.com.au/
http://www.seek.com.au/ http://seek.com.au/?
```

Hidden Web History
Unfortunately for forensics, there are easy ways for users to minimise the evidence. In Chrome, this is called *incognito* browsing.

However, going incognito does not hide browsing activity from your:

- Employer
- Internet Service Provider
- The websites you visit.

Recovering Hidden (incognito) Web History

- One likely place is volatile memory
    - Process history
    - System history
- Another is on the disk
    - Temporary files (including cached files)
    - Swap files
- In the local DNS server cache

Locating and Examining Temporary Internet Files

Basics
HTTP allows Web Browsers to *cache* recently visited pages.

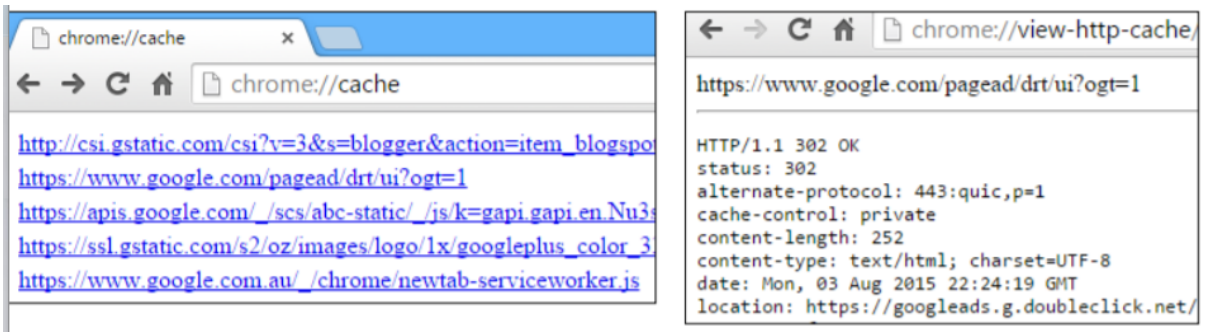
- When a viewer revisits a web page, HTTP checks the date on the cached page and devices whether to show the cached copy or refresh the page from the server.
- Caching cuts down on web traffic and speeds the rendering of the webpage.

Layout Engines
The temporary file location is chosen by the web page layout engine.

- Layout Engine for IE: *Trident*
- Layout Engine for Firefox: *Gecko*
- Layout Engine for Google Chrome: *Blink*
- Layout Engine for Edge (previously): *EdgeHTML*
- Layout Engine for Edge (now): *Blink*

Chrome Cache files
These use data files - no luck with the *strings* command.

However, the chrome decoder can be used.



Week 4 - Network Based Evidence, Packet Captures

Network abuse/attack Basics

Network Abuse

Example: A suspect downloads prohibited images. A disk examination may not find evidence of this. He may store the images on a USB so as to avoid detection on a work PC.

Furthermore, a suspect may conduct a private business on a work PC. He may avoid raising suspicions. He would then access sensitive company data and exfiltrate it to a remote internet location.

Network forensic examiners identify unusual network traffic

Intrusions into a network leave a trail:

- In the network firewall
- In the network Intrusion Detection System (IDS)
- In the network proxy server

Network forensic examiners identify compromised machines and take them offline.

Network Defences and Security

- Internet facing firewall and IDS
- *Demilitarised zone (DMZ)* network bridge
- LAN facing firewall and IDS
- Firewall and Antivirus on the hosts

Defence modes

People

- Well trained specialists dedicated to defending the network. The Guardians of the Gateway

Technology

- Strong network architecture, proven IDSs and firewalls
- Penetration testing
- Systems for log analysis

Operations

- Updating security patches
- Training and monitoring users
- Disaster recovery plans

Network Activity Protocols

Device Startup

- DHCP

Device Connection

- SSH or Telnet

Background Noise

- Switch STP (Spanning Tree Protocol)
- Routing Protocols (OSPF)
- Windows Active Directory

User Activity

- Accessing a Website
- Sending/Receiving Email
- Accessing a Work Connection (VPN)

Intruder Activity

- Using a back door

Network Based Forensics

An attack on a Digital Device can be performed *in person*, or over the digital network.

Network attacks involve:

- Opening a trapdoor on the target device
- Contacting the target device from a remote device
- Exchanging network packets to:
    - Install snooping software
    - Use that software to then retrieve sensitive information such as passwords.

Network Intrusion Detection

Network intrusion can be detected by administrators in several ways:

- Using a special Intrusion Detection hardware - IDS/IPS
- Equip a firewall with IDS features
- Have a *network based* IDS examine all network packets
- Have a *host based* IDS examine local network activity
- Record network activity in local log files
- Use a local Firewall/Virus scanner

Locating the evidence

Network evidence can be found:

- *On a suspect's device*
    - File folders, cache folders and swap files
- *On the local network*
    - Proxies, firewalls, IDS
- *On the ISP*
    - Proxies, firewalls
- *On the remote web site*
    - Logs

Missing Evidence Scenario

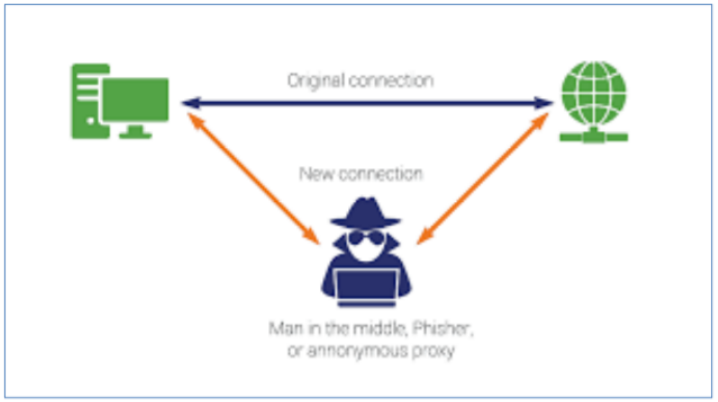
If a browser is set up to not save third party cookies, there will be no third party cookies on the disk.

However, the cookies are still sent by the server. A network packet capture will catch the cookies. Despite this, the practice of packet capturing is resource intensive.

Sequence of Accessing a Website

- DNS request
- HTTP handshake
    - Browser details
    - Server details
- HTML handshake
    - Style sheets
    - JavaScript
- Page display
    - Images, gifs and pngs
- SSL
    - SSL certificate exchange
- Plug-ins
    - Flash
- Extras
    - Cookies
    - Hit counters
    - Page tracking
    - ASP.Net

DNS Vulnerability
The DNS request/response process is subject to hacking by a man in the middle attack. Reveals the DNS request/reply to Wireshark.



However, more secure *encrypted DNS* is coming into use.

Securing DNS using the DoH client
DNS over HTTPs

```
C:\Users\graha>nslookup dns.google        dns request
Server:  mygateway
Address:  10.0.0.138                      dns server

Non-authoritative answer:
Name:    dns.google
Addresses:  2001:4860:4860::8844
          2001:4860:4860::8888
          8.8.4.4
          8.8.8.8                         dns reply

          network shell    domain name
C:\Users\graha>netsh dn show encryption server = 8.8.8.8

Encryption settings for 8.8.8.8
---------------------------------------------------------------------
DNS-over-HTTPS template    : https://dns.google/dns-query
Auto-upgrade               : no
UDP-fallback               : no
```

Text on the Internet
There are several ways text data can be saved in a database.

- A list of *Plaintext* as *.txt* files - each item is separated by a space or a TAB.
- A list of *Comma Separated Variables* as *.csv* files, each item is separated by a comma (,).
- A list of *JavaScript Object Notation (JSON)* files, each item is named with quotes (“) and separated by a comma (,)
    - “country_id”:”AU”,”city”:”Chatswood”

ASCII guide

Dec	Hex	Oct	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr
0	0	000	NULL	32	20	040	&#032;	Space	64	40	100	&#064;	@	96	60	140	&#096;	`
1	1	001	Start of Header	33	21	041	&#033;	!	65	41	101	&#065;	A	97	61	141	&#097;	a
2	2	002	Start of Text	34	22	042	&#034;	"	66	42	102	&#066;	B	98	62	142	&#098;	b
3	3	003	End of Text	35	23	043	&#035;	#	67	43	103	&#067;	C	99	63	143	&#099;	c
4	4	004	End of Transmission	36	24	044	&#036;	$	68	44	104	&#068;	D	100	64	144	&#100;	d
5	5	005	Enquiry	37	25	045	&#037;	%	69	45	105	&#069;	E	101	65	145	&#101;	e
6	6	006	Acknowledgment	38	26	046	&#038;	&	70	46	106	&#070;	F	102	66	146	&#102;	f
7	7	007	Bell	39	27	047	&#039;	'	71	47	107	&#071;	G	103	67	147	&#103;	g
8	8	010	Backspace	40	28	050	&#040;	(	72	48	110	&#072;	H	104	68	150	&#104;	h
9	9	011	Horizontal Tab	41	29	051	&#041;	)	73	49	111	&#073;	I	105	69	151	&#105;	i
10	A	012	Line feed	42	2A	052	&#042;	*	74	4A	112	&#074;	J	106	6A	152	&#106;	j
11	B	013	Vertical Tab	43	2B	053	&#043;	+	75	4B	113	&#075;	K	107	6B	153	&#107;	k
12	C	014	Form feed	44	2C	054	&#044;	,	76	4C	114	&#076;	L	108	6C	154	&#108;	l
13	D	015	Carriage return	45	2D	055	&#045;	-	77	4D	115	&#077;	M	109	6D	155	&#109;	m
14	E	016	Shift Out	46	2E	056	&#046;	.	78	4E	116	&#078;	N	110	6E	156	&#110;	n
15	F	017	Shift In	47	2F	057	&#047;	/	79	4F	117	&#079;	O	111	6F	157	&#111;	o
16	10	020	Data Link Escape	48	30	060	&#048;	0	80	50	120	&#080;	P	112	70	160	&#112;	p
17	11	021	Device Control 1	49	31	061	&#049;	1	81	51	121	&#081;	Q	113	71	161	&#113;	q
18	12	022	Device Control 2	50	32	062	&#050;	2	82	52	122	&#082;	R	114	72	162	&#114;	r
19	13	023	Device Control 3	51	33	063	&#051;	3	83	53	123	&#083;	S	115	73	163	&#115;	s
20	14	024	Device Control 4	52	34	064	&#052;	4	84	54	124	&#084;	T	116	74	164	&#116;	t
21	15	025	Negative Ack.	53	35	065	&#053;	5	85	55	125	&#085;	U	117	75	165	&#117;	u
22	16	026	Synchronous idle	54	36	066	&#054;	6	86	56	126	&#086;	V	118	76	166	&#118;	v
23	17	027	End of Trans. Block	55	37	067	&#055;	7	87	57	127	&#087;	W	119	77	167	&#119;	w
24	18	030	Cancel	56	38	070	&#056;	8	88	58	130	&#088;	X	120	78	170	&#120;	x
25	19	031	End of Medium	57	39	071	&#057;	9	89	59	131	&#089;	Y	121	79	171	&#121;	y
26	1A	032	Substitute	58	3A	072	&#058;	:	90	5A	132	&#090;	Z	122	7A	172	&#122;	z
27	1B	033	Escape	59	3B	073	&#059;	;	91	5B	133	&#091;	[	123	7B	173	&#123;	{
28	1C	034	File Separator	60	3C	074	&#060;	<	92	5C	134	&#092;	\	124	7C	174	&#124;	|
29	1D	035	Group Separator	61	3D	075	&#061;	=	93	5D	135	&#093;	]	125	7D	175	&#125;	}
30	1E	036	Record Separator	62	3E	076	&#062;	>	94	5E	136	&#094;	^	126	7E	176	&#126;	~
31	1F	037	Unit Separator	63	3F	077	&#063;	?	95	5F	137	&#095;	_	127	7F	177	&#127;	Del

asciichars.com

URL Encoding

- Also called % encoding
- Non alphanumeric characters can only be sent over the Internet using the ASCII character set.

Example: address.com/page 1/ = address.com%2Fpage%201%2F

We will see % encoding in the captured HTTP packets.

Unicode Byte Encoding

UTF-8: 1-byte for the first 127 code points (maintaining compatibility with ASCII), and an optional 1-3 bytes (4 bytes total) for other characters.

Network Based Evidence - NBE

Basics

There are four broad methods:

Full content Data

Examine every packet

Session Data

Examine TCP session data

Alert Data

Examine errors and exceptions

Statistical data

Examine unusual events/anomalies

Full Content Data
Collects every bit of every packet, on Ethernet or Wireless.

A *Packet Capture Library (libpcap)* is needed on the device network interface.

- Wireshark is a typical application.
- Usually only used after an intrusion
- Extensive disk space used.
- Excellent evidence
    - Can detect attacks on other systems
    - Can expose advanced attacks

However, encrypted packets can be a problem.

Session Data
Derived from the TCP sessions and often available during the initial intrusion.

- Indicates the time and date, as well as the parties involved.
- Can often see all the intrusion sequence
- We can look for strange IP addresses.
- We can also look for unusual ports in use, e.g. IRC.
- High traffic could mean a file transfer.

Session DNS requests are not encrypted.

Alert Data
When an IDS/IPS sees a packet that matches a virus signature or an intrusion rule, it sends an alert.

- The IPS needs tuning for best results.
    - Avoid false positives
    - Watch out for back doors.
- Usually will not detect theft of sensitive data.

Once again, encrypted packets can be a problem.

Statistical Data
Can show variations

- Top 10 websites
- Top 10 internal users
- Unusual web addresses and ports
- Which processes/services transfer the most data.

Immune to encryption.

Honeynets
A good way to find popular network attack methods is to use a *honeynet*.

- This provides awareness, information and tools.
- Honeynets comprise of honeypots and honeywalls.

Honeypot
A network device with weak defences that advertises its contents which are actually of no value

Honeywall
Monitors what attackers try to do to access a honeypot

- Which username and password did the adversary use?
- At which speed did they brute force?
- From where did they proxy from?
- What time of day did they brute-force?

Accessing the wire
Two main methods:

- Place the pcap device on the wire between the edge router and the firewall

    - Either use a hub
    - Or two interface cards as a bridge.
- Use a switch running SPAN
    - Switch Port Analyser
    - Built into Cisco switches

Using Packet Captures to baseline a device NBE

Data Sources

- Packets can come live from a device
    - From packet capture (pcap) on the network adapter
- From a .pcap file
    - From Wireshark
    - Tcpdump
    - Dumpcap
    - Text2pcap
    - Snort

Evidence of Accessing a Web Site

- Browser/server HTTP handshake
- CSS and JavaScript download
- Page download
    - Text, gifs and jpegs
    - Some may come from the local cache.
- Plugins started
- Cookies downloaded
- External Page Tracking

Searching a .pcap for URLs

- We find URLs in a .pcap file using *find*, *grep* or *wireshark*.
    - For many files or large files, these methods do not scale well.
- We can use a python script ot find URLs
- Searching for words that match a keyword dictionary.

More Protocols in Wireshark

Wireshark provides tools to dig out evidence:

- Suspect logging onto a remote site (SSH)
- Suspect using a VPN (ISAKMP, ESP, AH)
- Suspect accessing a bank web site
    - X.509 Certificates (SSL)
- Suspect using Wireless (802.11)
- Suspect using VOIP (SIP)

Wireless radio frame capture

- Captures available *Wireless Access Points (WAPs)*, whether they are broadcasting or not.
- Windows requires Wireshark in monitor mode.
    - A special USB Wireless Adapter
        - *Airpcap*
    - Or npcap with USBcap
- Linux can use software on other wireless adapters
    - Such as *aircrack-ng*, available on the Kali distro
    - Requires a USB wireless card such as TP-Link

Week 5 - Hex View of Data/File Metadata

Looking at Disk Bytes as Hex

Basics

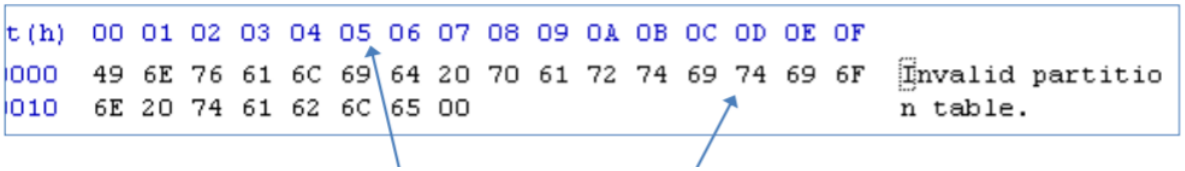
- The unit of data is the byte - 8 bits
- This can contain 2^8 = 256 combinations
- These combinations can be represented in base 16 notation (hex)

Decimal	0	1	2	3	...	10	11	12	..	15
Hex	0	1	2	3	...	A	B	C	..	F

Thus, the range of data 0 -255 is now 0 - FF in hex.

The Hex view of data

By using a hex editor:

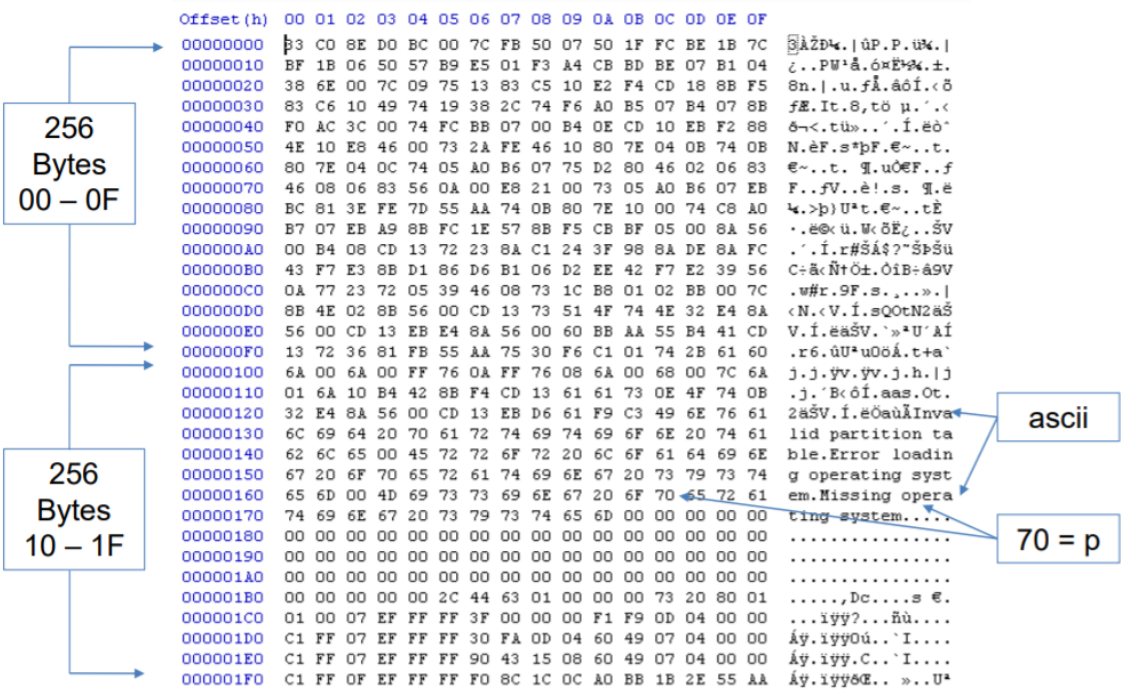


In this above image: *hex -> ascii -> plain text error message*

Hex Displays

- To display 256 bytes we use a 16 x 16 array
  - In Hex, this is an (0 - F) x (0 - F) array
- To display 512 bytes, we use a 16 x 32 array
  - In Hex, this is a (0 - F) x (0 - 1F) array

The hex editor will usually display an ASCII view as well.



Examining Image Metadata

Types of Metadata

Structural Metadata

Describes the data contained.

- How the data is structured.
- An example is the Tag:Value pair

Descriptive Metadata

Describes individual instances

- “city”:”Chatswood”

Administrative Metadata

Describes how the data is used.

- Origin, Category, Access rights

Graphic Formats

Pictures (images) can be saved as a file. There are various ways to do this:

- Bitmap - dots (pixels) as a 2D array
- Vector Graphics - define shapes such as lines (length + direction)

Bitmap editors:

- MS Paint
- Photoshop
- Gimp

Bitmap viewers:

- MS Office

Vector editors:

- CorelDraw
- Adobe Illustrator

Vector viewers

- Pdf-xchange
- Adobe Reader

Graphic Files

- Black and White Bitmaps have one bit per pixel.
    - A 640x480 pixel image requires 307200 bits = 0.37MB
- Real colour has 24 colour bites for each pixel.
    - A real colour 640x480 image needs 7.37MB.

A vector graphic of a simple image is much smaller than the pixel version.

Graphic Compression

Similar to text compression.

Consider a green line of length 100 bits. Instead of recording in the file 100 green bits, we record one green bit and a *x100 instruction*.

- Some compression is lossless - meaning uncompressing retrieves the original.
- However, other compression is lossy - meaning that uncompressing retrieves only part of the original.

Graphic file types

- Pixels
- BMP
    - No compression

    - Large and lossless
- TIFF
    - Tagged bitmap
    - Large and lossless
- GIF, PNG
    - Simple graphics, 8-bit colour
- JPEG, JPG
    - Small and lossy, used to share photos
- EXIF
    - Combines JPEG and TIFF for camera metadata
- Vectors
- SVG
    - Open standard that includes scripting.

Graphic File hex signature tags
A graphic file often has an identifying tag near the start.

```
G:\Forensics>xxd -l 16 logo.gif
0000000: 4749 4638 3961 dc00 3200 f700 00ff c35c  GIF89a..2......\

G:\Forensics>xxd -l 32 "MS Office Meta Data.jpg"
0000000: ffd8 ffe0 0010 4a46 4946 0001 0101 0096  ......JFIF......
0000010: 0096 0000 ffdb 0043 0001 0101 0101 0101  .......C........

G:\Forensics>xxd -l 32 IMAG1672a.jpg
0000000: ffd8 ffe0 0010 4a46 4946 0001 0101 0048  ......JFIF.....H
0000010: 0048 0000 ffe1 1242 4578 6966 0000 4d4d  .H.....BExif..MM
```

Examining File Metadata

Metadata sample from a Camera Image

Tag	Value
Manufacturer	CASIO
Model	QV-4000
Orientation (rotation)	top - left [8 possible values[21]]
Software	Ver1.01
Date and Time	2003:08:11 16:45:32

Magic Files
Many file formats have *headers* with metadata.

- File authoring
- Camera files have camera settings, gps location, etc.

As there are many file types, magic numbers evolved.

- Originally these were two bytes at the start of the file.
- The identifiers are now stored in a separate magic file.

```
root@kali64:~# whatis magic
magic (5)            - file command's magic pattern file
```

The magic patterns are found in: */usr/share/misc/magic*

Each file type signature is described as a comment.

Magic File Example - Text File

```
C:\Forensics_Graham>type test.txt
This is a text file
With two lines of text

C:\Forensics_Graham>xxd test.txt
0000000: 5468 6973 2069 7320 6120 7465 7874 2066  This is a text f
0000010: 696c 6520 0d0a 5769 7468 2074 776f 206c  ile ..With two l
0000020: 696e 6573 206f 6620 7465 7874 200d 0a    ines of text ..
```

- 20 = space
- 0D = Carriage Return (CR), 0A = Line Feed (LF) / new line

The End of Line marker (EOL) is *0D0A* in Windows. This text file ends with a single EOL.

Magic File examples - graphic files

```
G:\Forensics>file logo.gif
logo.gif: GIF image data, version 89a, 220 x 50

G:\Forensics>file "MS Office Meta Data.jpg"
MS Office Meta Data.jpg: JPEG image data, JFIF standard 1.01

G:\Forensics>file IMAG1672a.jpg
IMAG1672a.jpg: JPEG image data, JFIF standard 1.01
```

Magic File examples - MS Word

```
Magic entry for Microsoft Office XML

# start by checking for ZIP local file header signature
0               string          PK\003\004
# make sure the first file is correct
>0x1E           string          [Content_Types].xml
```

```
C:\Forensics>xxd -l 64 Test.docx
0000000: 504b 0304 1400 0600 0800 0000 2100 0924  PK..........!..$
0000010: 8782 8101 0000 8e05 0000 1300 0802 5b43  ..............[C
0000020: 6f6e 7465 6e74 5f54 7970 6573 5d2e 786d  ontent_Types].xm
0000030: 6c20 a204 0228 a000 0200 0000 0000 0000  l ...(..........
```

Magic File example - Windows .exe (executable) file

```
C:\Forensics>xxd -l 144 grep.exe
0000000: 4d5a 9000 0300 0000 0400 0000 ffff 0000  MZ..............
0000010: b800 0000 0000 0000 4000 0000 0000 0000  ........@.......
0000020: 0000 0000 0000 0000 0000 0000 0000 0000  ................
0000030: 0000 0000 0000 0000 0000 0000 8000 0000  ................
0000040: 0e1f ba0e 00b4 09cd 21b8 014c cd21 5468  ........!..L.!Th
0000050: 6973 2070 726f 6772 616d 2063 616e 6e6f  is program canno
0000060: 7420 6265 2072 756e 2069 6e20 444f 5320  t be run in DOS
0000070: 6d6f 6465 2e0d 0d0a 2400 0000 0000 0000  mode....$.......
0000080: 5045 0000 4c01 0700 e45a 5852 0034 0300  PE..L....ZXR.4..
```

- MZ at 00
- Jump to (80) at 0000030 (compiler dependent)
- DOS ASCII text message at 0000040
- PE (Portable Executable) at or after 80 (compiler dependent)

The *file* file

In Linux, the magic file is accessed using the *file* command.

```
C:\Forensics>file Test.docx
Test.docx: Microsoft Word 2007+
```

file * - lists all files

```
trade_secrets.txt: ISO-8859 English text, with
very long lines, with CRLF line terminators

ls.exe: PE32 executable (console) Intel 80386
(stripped to external PDB), for MS Windows

cmarko-tskintro.pdf: PDF document, version 1.4
```

File Date/Time

In Linux, you can *stat* a file to see the three date/time stamps. This command, overall, shows the basic file stats.

```
group11/mnt/c/Users/graha$ stat Sample.docx
  File: Sample.docx
  Size: 96097         Blocks: 192        IO Block: 4096
Device: eh/14d  Inode: 36873221949387872  Links
Access: (0777/-rwxrwxrwx)  Uid: ( 1000/ group11)
Access: 2020-08-07 16:38:38.358998600 +1000
Modify: 2013-09-29 21:00:12.000000000 +1000
Change: 2019-08-18 07:42:28.118143600 +1000
 Birth: -
```

- However, in Linux, a suspect can *touch* a file to change one or more of these date/time stamps.
    - In spite of this, the file header may contain another date/time stamp as *metadata*. This may be missed by the suspect when they are using the *touch* command.

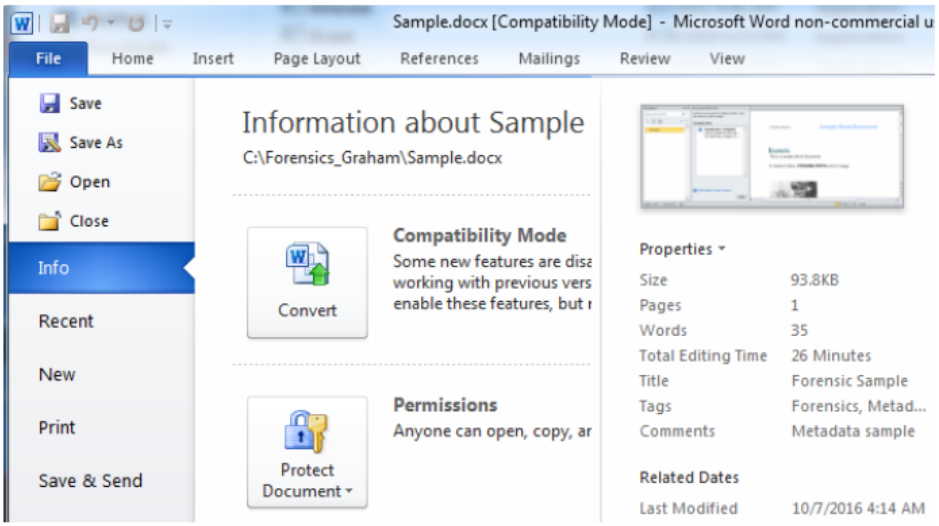
Metadata in Some Documents

MS Office Files

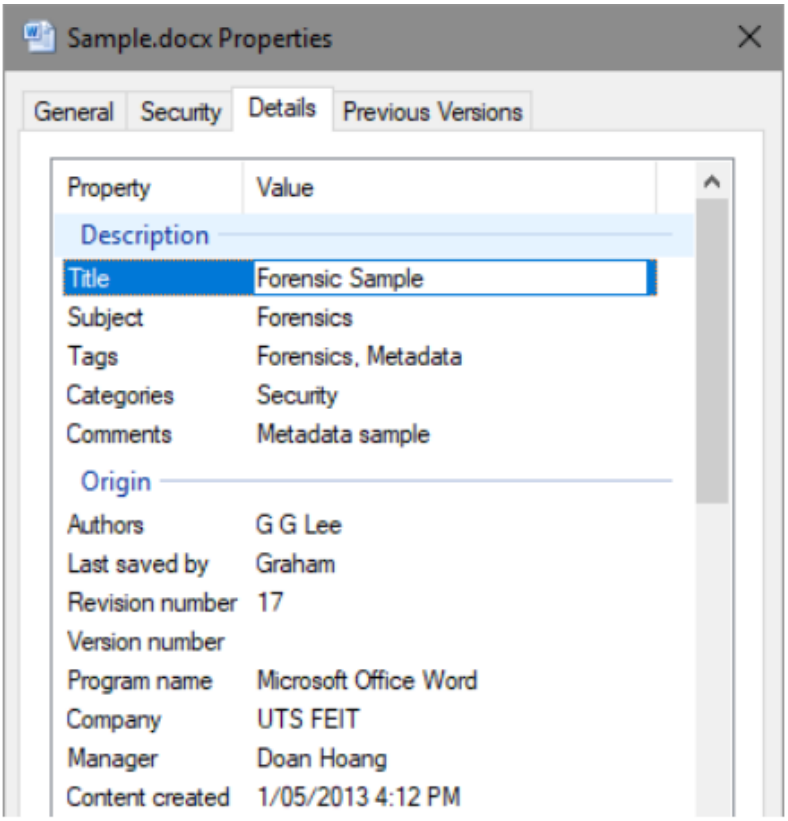
Files from MS Office are three .xml zipped files. These can be examined with the *7-zip* tool.

- *app.xml*
  - Application properties, such as page count
- *core.xml*
  - Author, date altered, print date
- *word/media*
  - Contains any images

DOCX Metadata in MS Word



In Windows 10:



PDF Files
The PDF structure is complex.

- Includes a header, objects and a trailer.
- Uses internal scripting commands.
    - */Launch* will launch a program
    - */JavaScript* will launch JavaScript
    - */OpenAction* will run a script on open

PDF Headers
The PDF Header starts with %PDF-1.x

```
C:\Forensics>xxd -l 16 "Evidence ACPO.pdf"
0000000: 2550 4446 2d31 2e35 0d25 e2e3 cfd3 0d0a  %PDF-1.5.%......
```

Adobe versions also contain the hex value (in the header) *25 e2 e3 cf d3*.

The Trailer contains an End of File marker.

```
25 45 4F 46 0D 0A                                        %EOF..
```

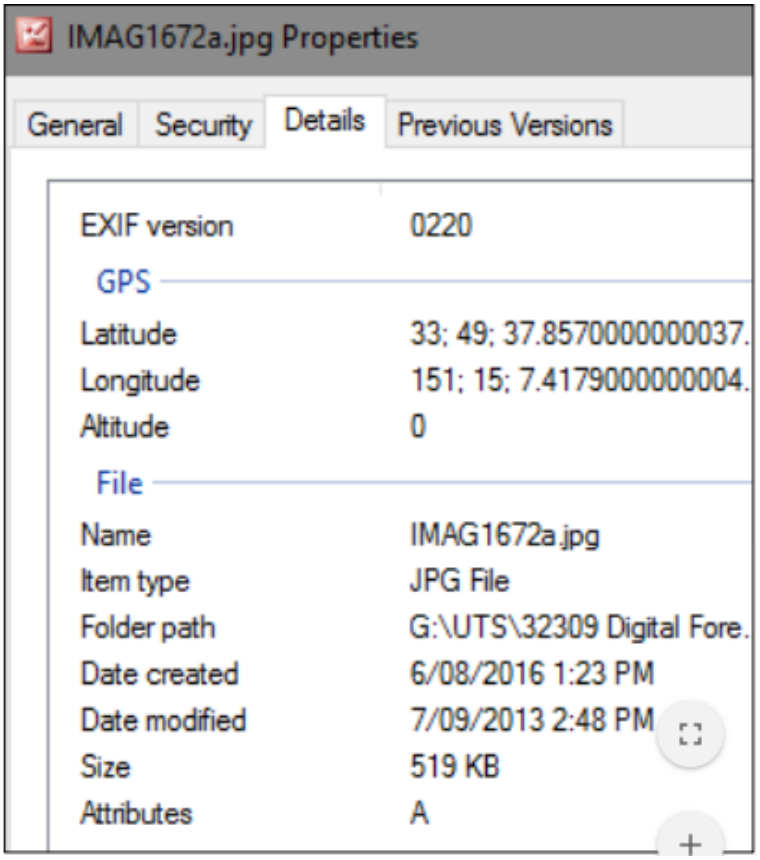
Malicious PDFs
Malware can:

- Embed an .exe into the PDF file
- Embed malicious JavaScript into the PDF.

Camera Files (i.e. EXIF)
EXIF - *exchangeable image file format*

- Used by cameras, smartphones and scanners.
- Developed from JPEG and TIFF
- Includes Geolocation

Camera files are EXIF files, but save as JPEGs.



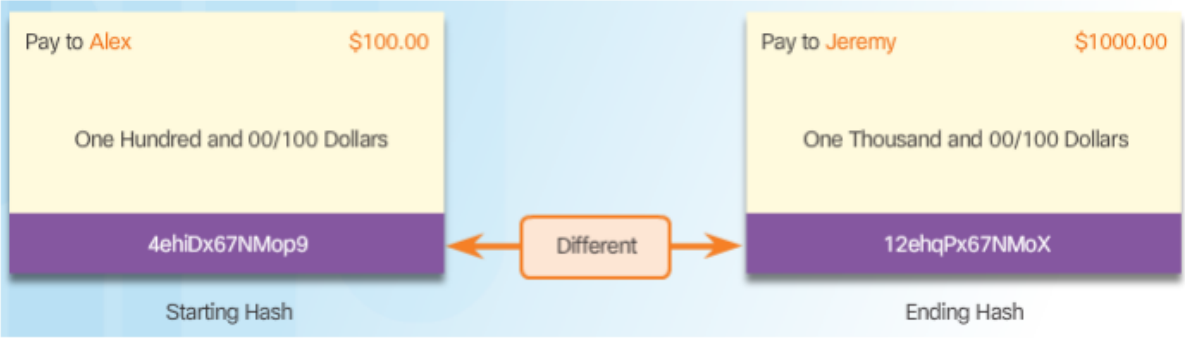
Email Metadata

- Email *.EML files can be found on the client.
- Emails contain headers that may have useful information:
    - the mail server IP used to send the email
    - IP address of the person receiving the email
    - IP addresses that the email was passed to
    - email authentication.

Hashing

Basics

Protects the integrity of a file and/or string.



Message Digest (MD5) versus Secure Hash Algorithm (SHA)

Secure Hash Algorithm has a longer hash value, meaning that it will be more difficult than it already is to experience a hash collision.

The Use of Forensic Hashing

- We can use hashing to ensure the integrity of evidence.
- We can pick up changes made to images - for instance, by a suspect or a virus.

However, in the event of a live disk, a second hash will be different as a live disk is always changing.

Thus, we use *forensic hashing* to hash many parts of a disk and keep the results in a table of hashes.

Steganography

“Hiding in plain sight”

- Invisible to the naked eye. Obfuscation of the true form of a file.

Two methods - insertion and substitution.

Insertion

In HTML, there is a *hidden* attribute. The data inside the tag is hidden.

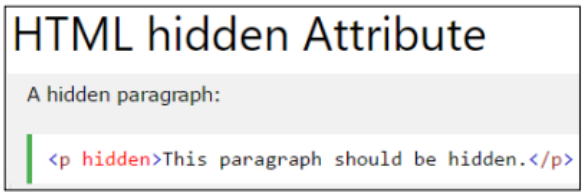
Substitution

Replaces *least significant bits (lsb)* in a bitmap image with data.

- Can also embed data in mp3 audio files.

- Other methods adjust spacing characters in text files.
- Common steno programs are password protected.

It is often hard to detect substitution stenography using an IDS, as a malicious file can be obfuscated to seem harmless, thus not alerting the systems within the IDS.



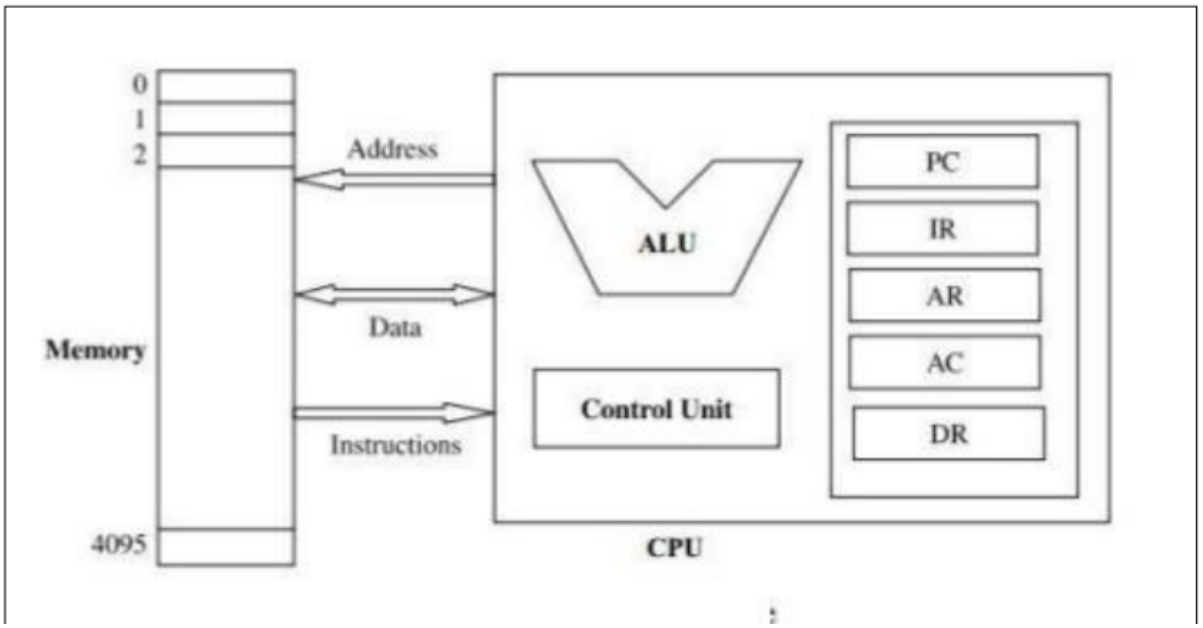
Watermarking
Commercial programs such as Photoshop can watermark in image to detect a copyright infringement.

Watermarks can record:
- The copyright owner
- The distributor
- The distribution chain
- Purchaser of the document, game or music

DRM
Digital Rights Management (DRM) supply an encryption key for paid downloads such as PDF courseware.
- The document will not open on any other device.
- This technique is also used in the Gaming industry.

Week 6 - Memory, Process and Windows Registry

CPU Memory access



The CPU

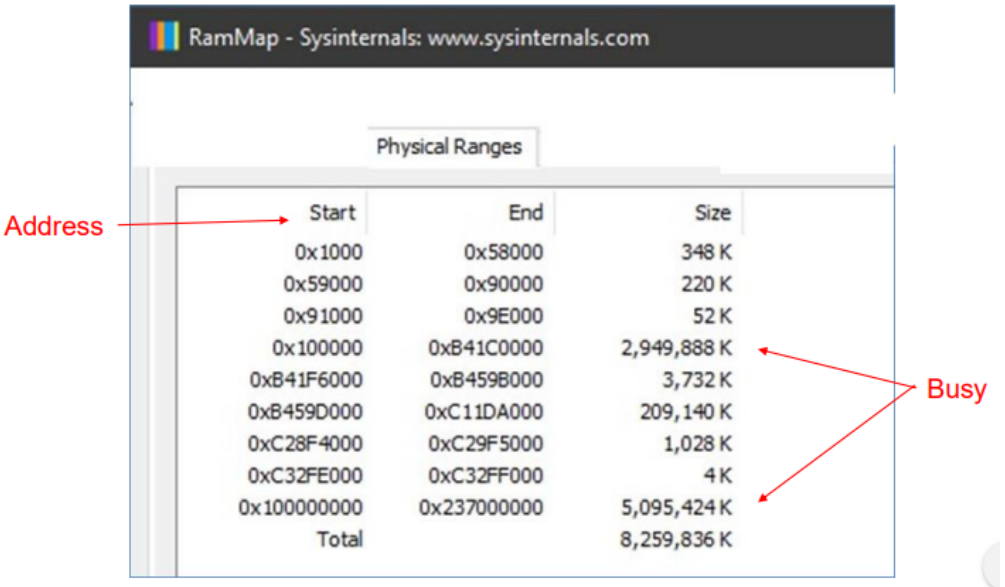
The Central Processing Unit executes *instructions* to perform actions on data.

- These instructions are kept in *memory* as *program segments*.
- The data is also kept in memory (data segments)

Unlike *disk storage,* memory in RAM is volatile.

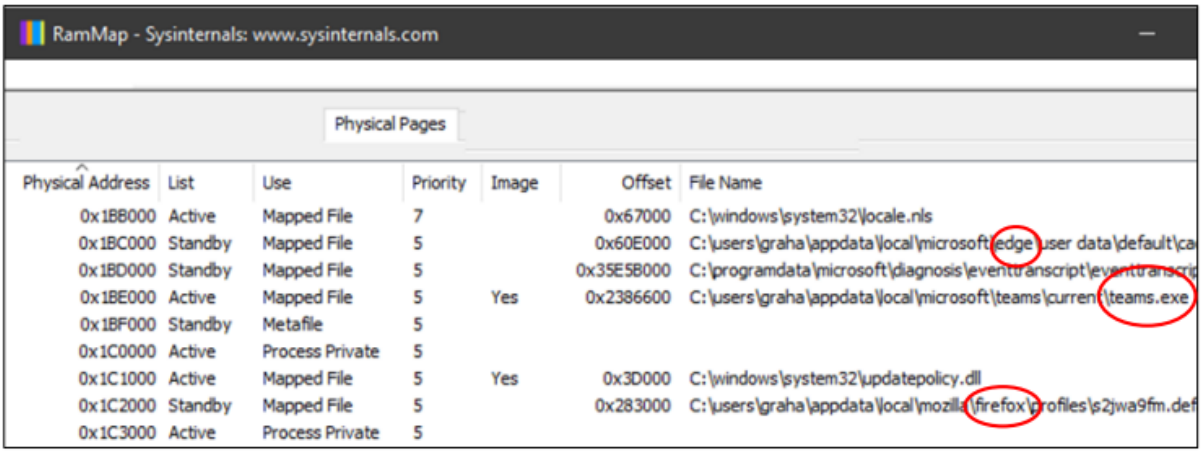
Physical Memory

Physical Memory Range



Physical Pages

A file on the disk can be mapped into a memory page.



How large is Memory?

IA-32 Intel CPUs can access 4GB of memory. However, there is a technique called *Physical Access Extension (PAE)*

The OS may limit RAM as a sales incentive.

- Win10 memory limits:

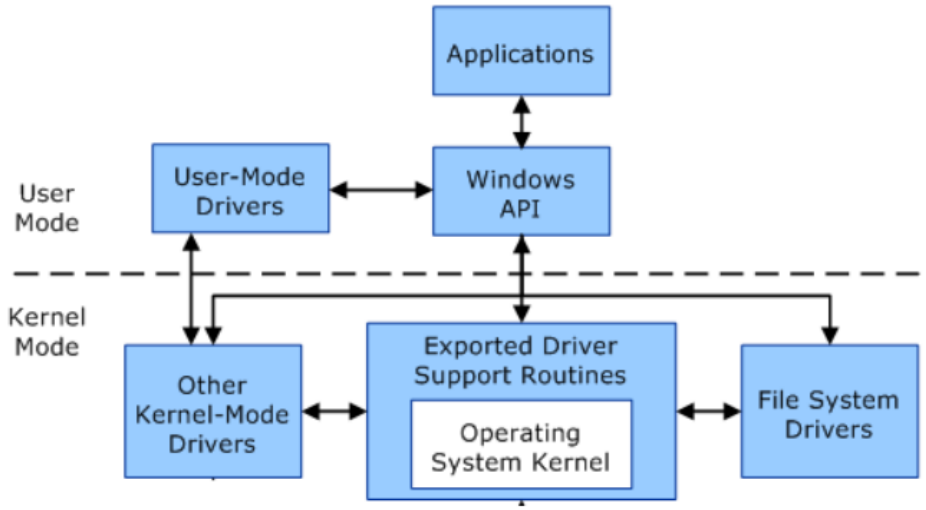
Version	Limit on X86	Limit on X64
Windows 10 Enterprise	4 GB	2TB
Windows 10 Education	4 GB	2TB
Windows 10 Pro	4 GB	2TB
Windows 10 Home	4 GB	128GB
Apr 20, 2018		

What writes to Memory?

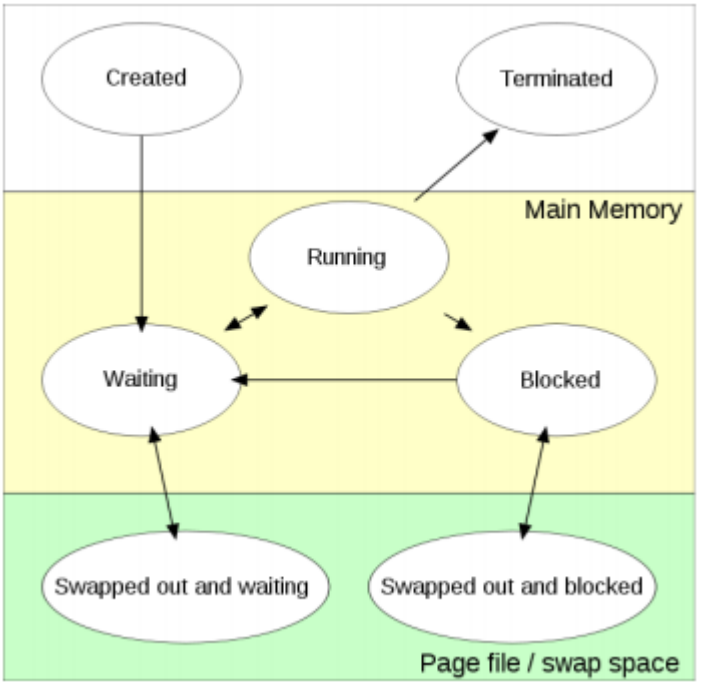
- The *Memory Management Unit (MMU)* handles memory requests.
- Alternatively, there is a *Translation Look aside Buffer (TLB)* that may hold memory data.
- In addition, some devices like graphics cards have *Direct Memory Access (DMA).*

Operating System modes

- The core OS runs in *kernel mode*.
    - This can access most of the RAM
    - This includes may drivers
    - All kernel mode processes can see each other's RAM.
- The user apps run in *user mode*.
    - RAM access is restricted
    - Each user mode process runs in its own sand box
    - A user mode process cannot access kernel mode RAM.



Processes



Basics

A process is a running program launched from an *.exe*

- Every *task* in a PC runs as a process.

Forensics examines processes to locate evidence.

Data Structures in Memory

To recover information from memory, investigators must know how it is stored.

- *Arrays* - usually of a fixed size
- *Bit Maps* - sparse arrays (e.g. TCP ports in use)
- *Records* - name:value pairs
- *Strings* - often 00 terminated
- *Linked Lists*
- *Hash Tables*
- *Hierarchical Trees*

Address Space Layout Randomization (ASLR)

A technique to reduce memory hijacking.

- Prevents an attacker from reliably jumping to an address in memory.

ASLR *randomly arranges the address space positions* of key data areas of a process.

Process memory footprint

Each process has artifacts that identify it in RAM.

- Open file handles
- Recent DLLs used
- Memory mappings
- Network connections (sockets) e.g. 29463:22
- Privileges

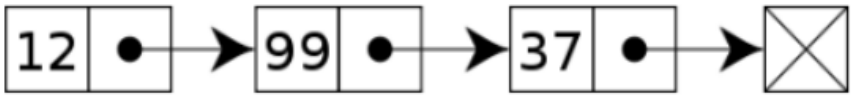
Process Source

- Task Manager reveals many processes:
  - How did they start?
  - Who published them?
  - When were they written?
- We can see running processes with:

- *TaskList* (Built-In Windows)
- *PsList* (SysInternals)

The Linked List

For Task Manager to keep a detailed track of processes (tasks), it must use a *linked list of nodes.*

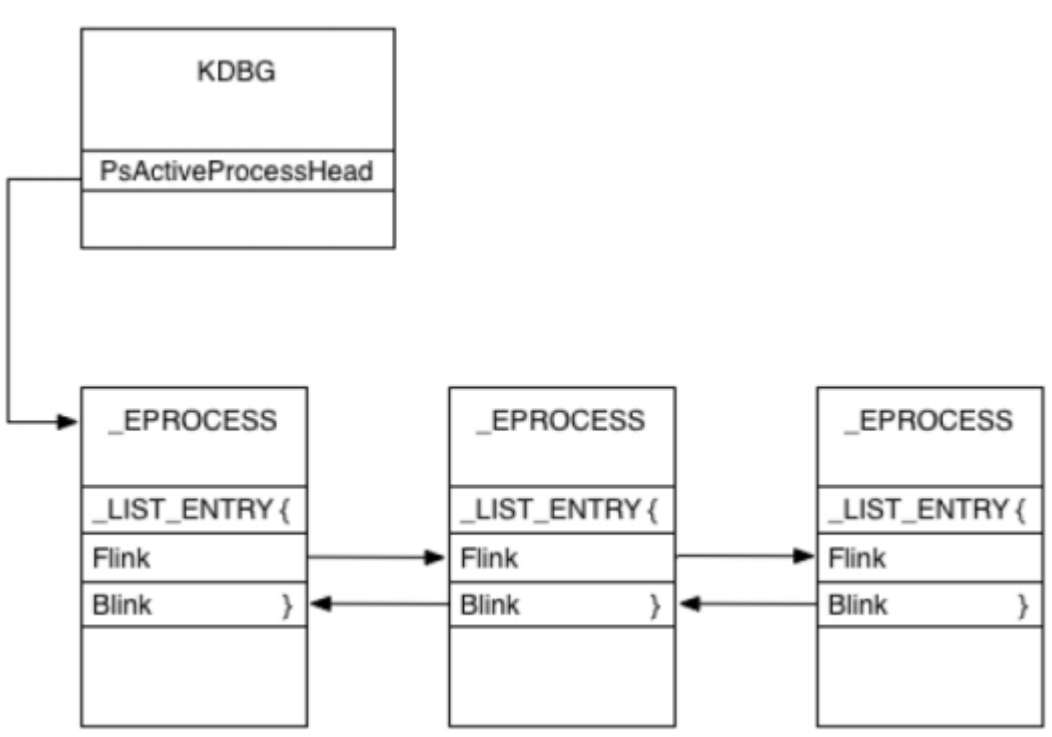


Each node in the list has a *value* and a *pointer* to the next node. The last node is linked to a *terminator*.

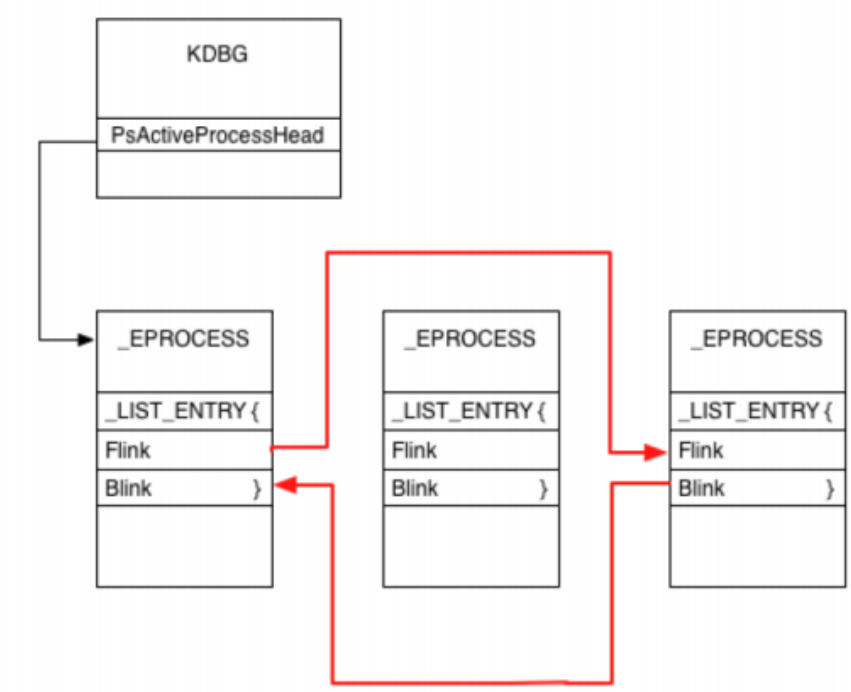
Listing Processes

- *Task Manager* displays a list of processes.
  - Starting at *PsActiveProcessHead*
  - Then links to each _EPROCESS structure
  - Active processes are displayed.
- The *Executive Process* list has more processes.
  - Active, Hidden, Deleted
- Some tools can dump all these processes.
  - A virus can hide an evil process by manipulating the list.

Walking the list using forward and backward links



A rooted list - unlisted process



Executable file process

1. The linked program is compiled into an *exe*.
2. When the .exe is clicked, a dynamic linker reads the .exe file, and loads its pieces into memory.
3. The linker links the file DLL calls into the *running DLLs loaded into memory*.
4. Code is loaded into a *read only, executable* region.
5. Data is loaded into a *data* region.

Windows DLLs

A *Dynamic Link Library (DLL)* is a piece of code that can be shared by one or more processes.

- Windows has thousands of DLLs stored on disk.
- It is difficult to spot a DLL introduced by malware.
- Worse still, malware can alter an *existing* DLL.
    - This can, however, be detected by examining the DLL hash.

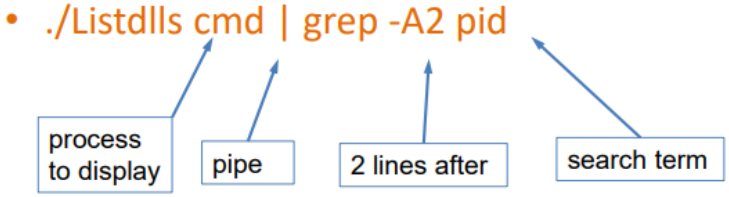
We can view *running DLLs* with:

- *Listdlls* (SysInternals)
- *Tasklist* (Windows)

Viewing a process source file

Listdlls shows how a process was launched.

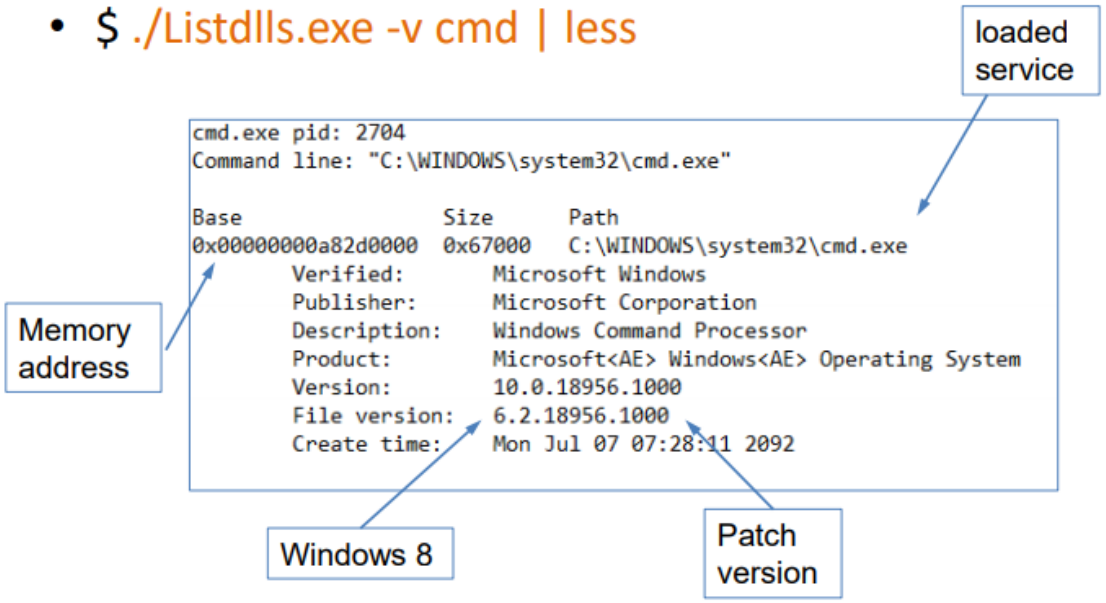
- We use *grep* to filter out the lines of interest.




```
$ ./Listdlls.exe cmd | grep -A2 pid
cmd.exe pid: 2704
Command line: "C:\WINDOWS\system32\cmd.exe"
```

Viewing DLL version detail

./Listdlls.exe -v cmd | less



Viewing DLLs with TaskList - WIndows

tasklist /m /fi "imagename eq cmd.exe"

- The /m option lists modules (DLLs)
- The /fi option filters by name or PID.

```
group11~$ tasklist.exe /m /fi "imagename eq cmd.exe"

Image Name                     PID Modules
========================= ======== ============================================
cmd.exe                       5800 ntdll.dll, KERNEL32.DLL, KERNELBASE.dll,
                                   msvcrt.dll, combase.dll, ucrtbase.dll,
                                   RPCRT4.dll, winbrand.dll, sechost.dll,
                                   apisethost.appexecutionalias.dll,
                                   msvcp_win.dll, kernel.appcore.dll,
                                   daxexec.dll, advapi32.dll, FLTLIB.DLL,
                                   shcore.dll, profapi.dll, container.dll,
                                   AppXDeploymentClient.dll,
                                   windows.storage.dll, IPHLPAPI.DLL,
                                   capauthz.dll, OLEAUT32.dll, WINTRUST.dll,
                                   CRYPT32.dll, MSASN1.dll, ntmarta.dll,
                                   windows.staterepositorycore.dll,
                                   bcryptPrimitives.dll
```

Services

Long running processes that have no user interface.

- Many services start automatically at boot
- Similar to daemons in Linux
- Some services are used for networking
    - *Webclient*
    - *Remote Procedure Calls (rpc)*
- Services can be run by *Service Host Processes*

- *svchost.exe*

Viewing processes running Services

TaskList.exe /svc

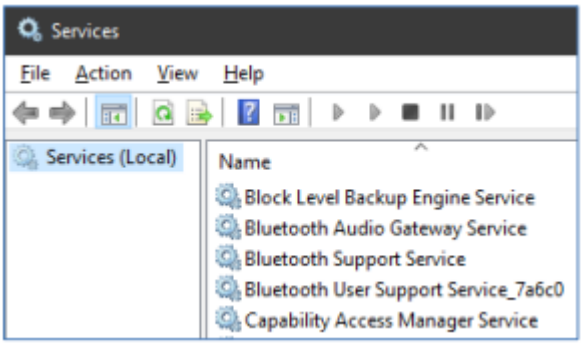
```
Image Name                     PID Services
========================= ====== ============================================
System Idle Process              0 N/A
System                           4 N/A
smss.exe                       936 N/A
csrss.exe                     1016 N/A
winlogon.exe                  1040 N/A
services.exe                  1084 Eventlog, PlugPlay
lsass.exe                     1096 PolicyAgent, ProtectedStorage, SamSs
svchost.exe                   1276 DcomLaunch, TermService
svchost.exe                   1384 RpcSs
svchost.exe                   1480 AudioSrv, BITS, Browser, CryptSvc, Dhcp,
                                   dmserver, ERSvc, EventSystem,
                                   FastUserSwitchingCompatibility, helpsvc,
                                   HidServ, lanmanserver, LanmanWorkstation,
                                   Netman, Nla, RasMan, Schedule, seclogon,
                                   SENS, SharedAccess, ShellHWDetection,
                                   srservice, TapiSrv, Themes, TrkWks, W32Time,
                                   winmgmt, wscsvc, wuauserv, WZCSVC
svchost.exe                   1600 Dnscache
svchost.exe                   1676 LmHosts, RemoteRegistry, SSDPSRV
spoolsv.exe                   1848 Spooler
```

More ways to view Services

- Using the Service Controller SC
  - *SC query type=service*

```
C:\Users\graha>sc query type=service | find /I "Bluetooth"
DISPLAY_NAME: Bluetooth Audio Gateway Service
DISPLAY_NAME: Bluetooth Support Service
SERVICE_NAME: BluetoothUserService_7a6c0
DISPLAY_NAME: Bluetooth User Support Service_7a6c0
```

- Use the service snap in
  - *Services.msc*



How Processes lead to Forensic Evidence

Basics

Memory accesses are far faster than disk accesses.

- A process opens the files it requires and places the contents in memory.
- It *decodes encryption (ssl and vpn)* in memory
- Passwords are also placed in memory.

Memory dumping is an important forensic activity. However, memory addressing is complicated, and requires specialised tools.

Memory Addressing

1. Request to read a virtual address
2. Translate to a physical memory address
3. Translate to file offset, decompress (if necessary)
4. Seek to, and read from, the file offset

Memory in Windows

Memory data may be:

- Incomplete
- Randomly organised
- Partly overwritten
- Repeated in different locations
- Changed by memory managers at any instant

Dumping all Memory

Memory can be quite large (8 - 24GB). Thus, we need 8 - 24GB disk space for the dump.

- It is not advisable to dump onto the system disk, as this may upset paging and swap files.
- The act of dumping may interfere with Memory Managers

To dump all memory: visit *System* > *About* > *Advanced* > *Startup and Recovery*

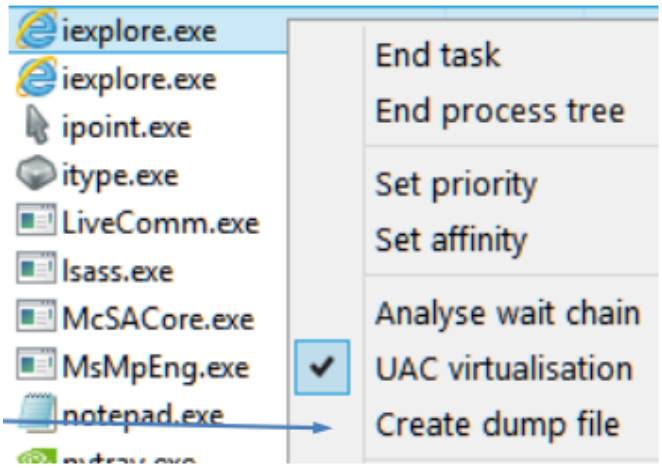
Searching all Memory

We use the *Volatility* tool add-on for Python.

- Volatility can analyse memory dumps from Windows, Linux, Mac OSX and Android ARM.
- It can also recover:
    - Process lists
    - Network Connections
    - Passwords
    - Web Sessions

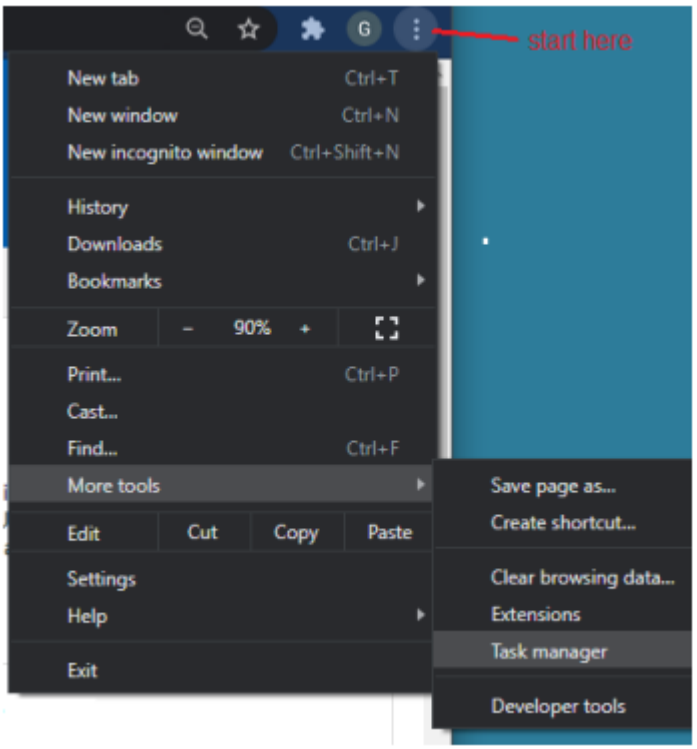
Dumping Process Memory - Example

Dumping a process in Windows 10 can be done, using *Task Manager*.

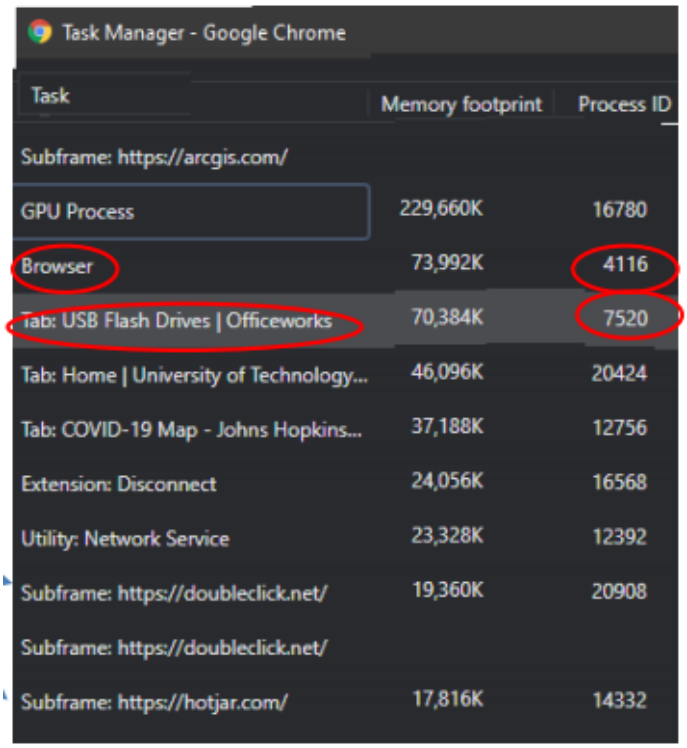


In Chrome:

1. Open Chrome.
2. Select the target website.
3. Open Chrome Task Manager.

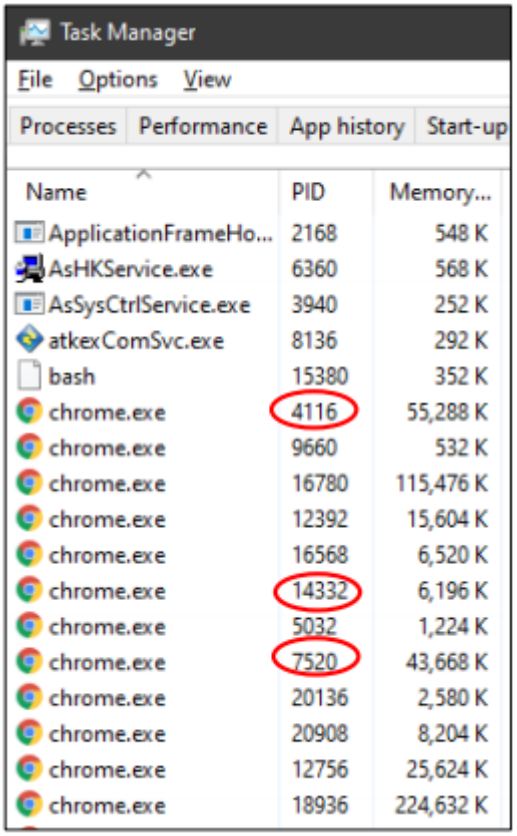


4. Note the PIDs you want
    - Browser
    - Your tab
5. Note the tracker processes.



6. Open Windows Task Manager

7. Dump the processes that match your desired PID.



Searching Process Memory

- We use *strings* to extract text in the binary dump.
  - *strings chrome.dmp > chrome.txt*
- We then search the text file using *grep*.
  - *grep passwd chrome.txt*
    - Looks for login passwords.
  - *Grep Set-Cookie chrome.txt*
    - Looks for cookies from websites.

Memory search filters

Memory is disorganised, and you will see unwanted hits (noise). Use filters to combat noise.

grep -i bazaar chrome.txt | cut -c 1-120 | grep -i officew -m20 | uniq

Wireshark text dump

Shows third party website pages.

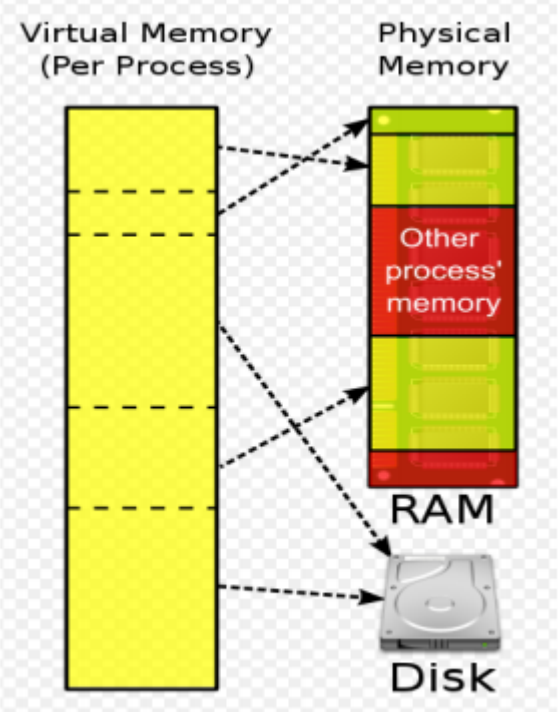
Memory text dump

Seeing running Javascript on the client:

```
$ grep -i bazaar chrome.txt  | cut -c 1-120 | grep -i -m20 officew | uniq
https://apps.bazaarvoice.com/deploy/officeworks-au/.../rating_summary-config.js
https://apps.bazaarvoice.com/deploy/officeworks-au/..../layouts
https://apps.bazaarvoice.com/deploy/officeworks-au/.../ratings-config.js
https://apps.bazaarvoice.com/deploy/officeworks-au/.../spotlights-config.js
https://apps.bazaarvoice.com/deploy/officeworks-au/.../swat-submission-config.js
https://display.ugc.bazaarvoice.com/./officeworks-au/../scripts/secondary.jsA
https://display.ugc.bazaarvoice.com/./officeworks-au/../scripts/bv-primary.js
https://apps.bazaarvoice.com/deploy/officeworks-au/.../layouts/rating_summary.json
https://apps.bazaarvoice.com/deploy/officeworks-au/.../bv.js?build=1537
https://apps.bazaarvoice.com/deploy/officeworks-au/.../reviews-config.js
```

Virtual Memory

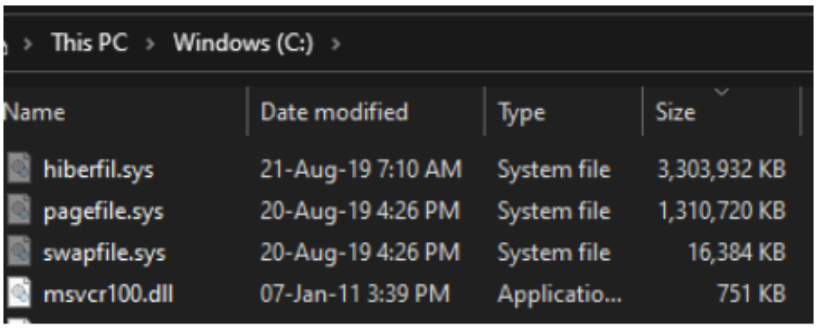
- The CPU would like to access all its programs in RAM.
- However, there is not enough RAM and it is volatile.
- Thus, unused RAM is swapped to disk files.



Memory on Disk

We can recover memory from the disk.

- Virtual memory page files (25% of RAM)
- Hibernation files (75% of RAM)
- Windows 10 swap files
- Crash files



Memory may also contain:

- Parts of the Windows Registry
- Parts of the Disk File Table
- Terminated processes
- Malware

The Windows Registry

Basics

A hierarchical database storing *configuration* settings.

- Very fast access (like cookies)
- Considered the 'brain' of Windows.
- Stored in *C:\windows\system32\config*

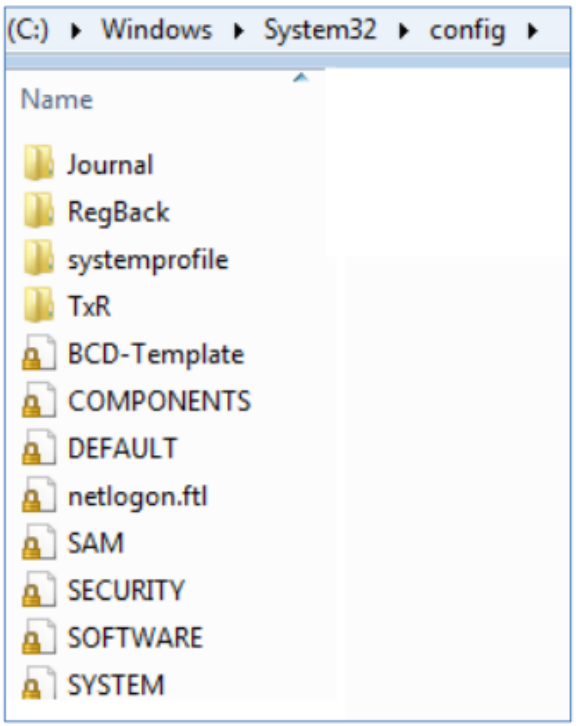
Each branch, or *hive* is called a *Handle to a Key (HKEY)*

- Only two master keys are stored on the disk.
    - HKLM and HKU.

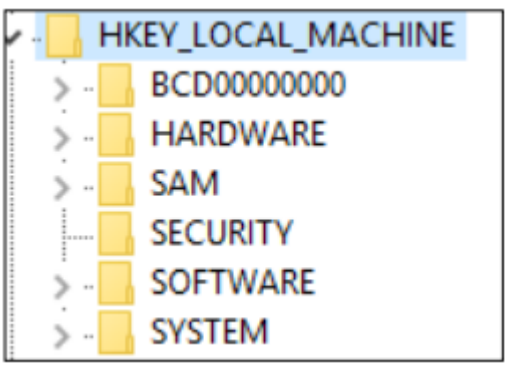
Viewing a live registry can be dangerous. A trivial change to a registry key can cause instant failure.

Registry Files

Files on Disk



Matching Registry Keys



HKLM Windows Registry Hives

HKey_Classes_Root (HKCR) - link to subkey in HKLM

- Contains file extension associations (*.exe, *.docx…)
- Software classes

HKey_Local_Machine (HKLM)

- Hardware
- Access passwords (SAM
- Installed software
- Device Driver Configs

HKey_Current_Config (HKCC) - link to subkey in HKLM

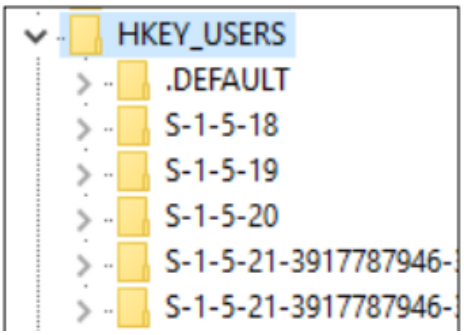
- Hardware Profiles

HKey_Current_User (HKCU) - link to subkey in HKU

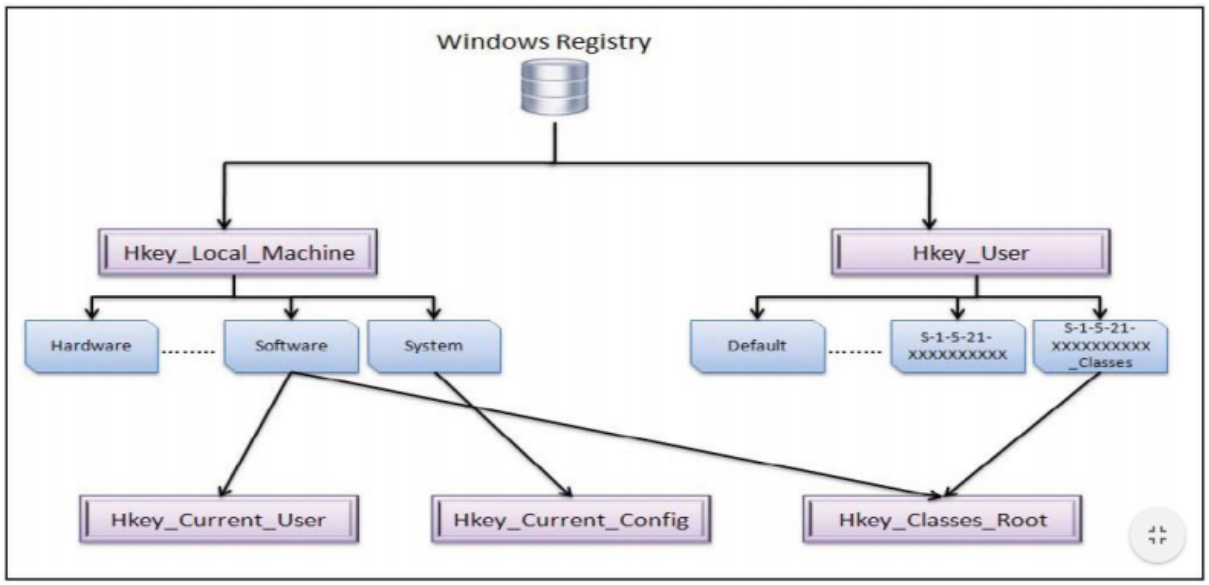
- NTUser.Dat in Documents and Settings

HKey_Users (HKU) - Master key on disk

- List of Users



Root Key Links



User Hives

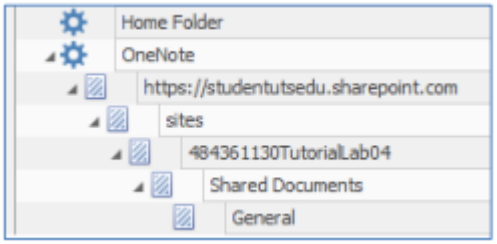
- NTUSR.DAT
- UsrClass.Dat
- Both used by ShellBags.
    - You will need a tool to see the ShellBags, such as ShellBags Explorer.

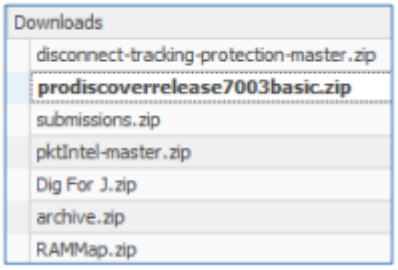
Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\hivelist

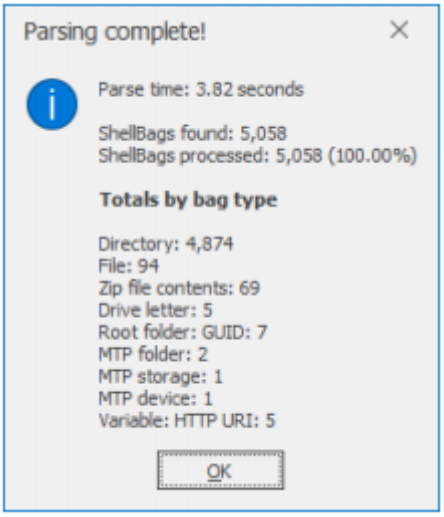
Name	Type	Data
(Default)	REG_SZ	(value not set)
\REGISTRY\MACHINE\BCD00000000	REG_SZ	\Device\HarddiskVolume1\EFI\Microsoft\Boot\BCD
\REGISTRY\MACHINE\HARDWARE	REG_SZ	
\REGISTRY\MACHINE\SAM	REG_SZ	\Device\HarddiskVolume3\Windows\System32\config\SAM
\REGISTRY\MACHINE\SECURITY	REG_SZ	\Device\HarddiskVolume3\Windows\System32\config\SECURITY
\REGISTRY\MACHINE\SOFTWARE	REG_SZ	\Device\HarddiskVolume3\Windows\System32\config\SOFTWARE
\REGISTRY\MACHINE\SYSTEM	REG_SZ	\Device\HarddiskVolume3\Windows\System32\config\SYSTEM
\REGISTRY\USER\.DEFAULT	REG_SZ	\Device\HarddiskVolume3\Windows\System32\config\DEFAULT
\REGISTRY\USER\S-1-5-19	REG_SZ	\Device\HarddiskVolume3\Windows\ServiceProfiles\LocalService\NTUSER.DAT
\REGISTRY\USER\S-1-5-20	REG_SZ	\Device\HarddiskVolume3\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
\REGISTRY\USER\S-1-5-21-215816962...	REG_SZ	\Device\HarddiskVolume3\Users\graha\NTUSER.DAT
\REGISTRY\USER\S-1-5-21-215816962...	REG_SZ	\Device\HarddiskVolume3\Users\graha\AppData\Local\Microsoft\Windows\UsrClass.dat

ShellBags Explorer

- Decodes the registry ShellBags.







Registry Keys

- A database of *tag:value* pairs
- The data in the value part can be of three types
  - *REG_BINARY*
    - Data is application dependent
  - *REG_DWORD*
    - Numbers, 1 = Active, 0 = Not Active
    - DWORD = double word = 32 bits
  - *REG_SZ*
    - ASCII string

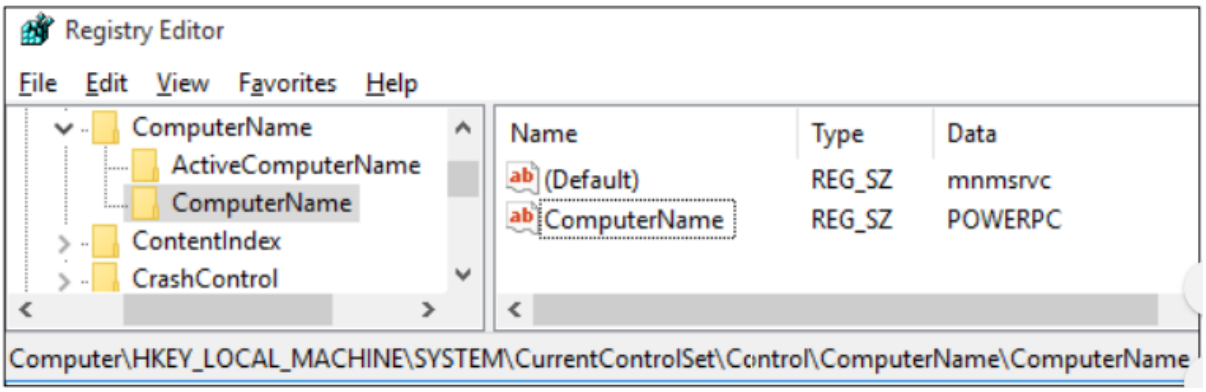
Sample Key

In the case where we want to find the Computer Name:

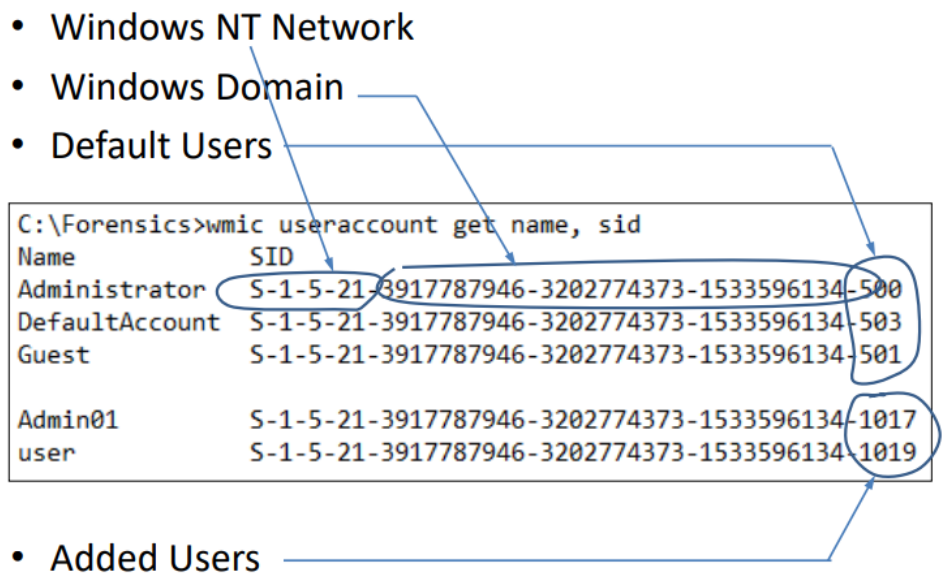
- We use the CLI for WMI (WMIC)

```
C:\Forensics>wmic computersystem get name
Name
POWERPC
```

Where is this stored in the Registry?



Sample Key 2 - Windows Users



Registry Issues

- Complex and Undocumented
- Easy to misinterpret and draw incorrect conclusions
- Many tools automate the analysis
- Need to be sure that the tool is correct and complete
- Deep analysis relies on three things:
    - Timelining
    - Baselining
    - Backup Analysis

Understanding GUIDs

GUIDs and UUIDs

UUID - Universally Unique Identifier

A 128-bit number used to identify information in Linux file systems.

GUID - Globally Unique Identifier

Used in Windows file systems.

- Many versions, here is version 1
- {4d36e967-e325-11ce-bfc1-08002be10318}
- {time of day-month-year-variant-MAC address}

UUID/GUID types

v1 GUIDs

- Have a '1' at the start of the third group.

– {72631e54-78a4-11d0-bcf7-00aa00b7b32a}

- Use the Gregorian calendar time (0=15 Oct 1582)
- The third group = 1xxx where xxx is the date code
- Use the user's NIC MAC address as the last 6 bytes

v4 GUIDs

- Have a '4' at the start of the third group
- Have 8, 9, A or B at the start of the fourth group.

– {53d29ef7-377c-4d14-864b-eb3a85769359}

- Use a *Psuedo Random Number Generator* for all other bits.

UUID/GUID Version 1 Date codes

UUID Dates - Version 1		
Third Group	Date low	Date high
11b2	1/01/1970	15/10/1970
11b8	1/04/1975	14/05/1982
11c0	20/05/1982	11/04/1983
11c8	9/07/1989	31/05/1990
11d0	27/08/1996	19/07/1997
11d8	16/10/2003	6/09/2004
11e0	4/12/2010	20/10/2011
11e3	8/08/2013	30/06/2014
11e6	11/04/2016	3/03/2017

UUID/GUID Version 1 MAC OUI codes

```
MAC OUIs
--------
00:0C:29 VMware, Inc.(VM)
00:1d:7d GIGA-BYTE TECHNOLOGY CO.,LTD.(Motherboard)
00:aa:00 Intel (NIC or CPU)
08:00:2b DEC (Digital Equipment Corporation - Unix)
2C:44:FD Hewlett Packard (PC)
```

Windows GUIDs

- Used to identify components in Windows hardware
  - Disk drives and partitions
- Used to identify software
  - Drivers
  - Class objects

```
typedef struct _GUID {
  DWORD Data1;          # 4 bytes = 32 bits
  WORD  Data2;          # 2 bytes = 16 bits
  WORD  Data3;          # 2 bytes = 16 bits
  BYTE  Data4[8];       # 8 byte array = 64 bits
} GUID;
Total Length = 128 bits (like IPv6)
```

Sample v1 GUIDs

```
Class = GPS
ClassGuid = {6bdd1fc3-810f-11d0-bec7-08002be2092f}

Class = DiskDrive
ClassGuid = {4d36e967-e325-11ce-bfc1-08002be10318}

Class = Net (Network Adapter)
ClassGuid = {4d36e972-e325-11ce-bfc1-08002be10318}

Class = Printer
ClassGuid = {4d36e979-e325-11ce-bfc1-08002be10318}
```

Linux UUIDs

- Used to identify block devices (disks)
- Look in */dev/disk*

```
root@kali:~#ls /dev/disk
by-id   by-label    by path     by-uuid

root@kali:~# ls -l /dev/disk/by-label/
total 0
lrwxrwxrwx 1 root root 10 Aug 19 04:03 FORENSICS -> ../../sdb1
lrwxrwxrwx 1 root root  9 Aug 19 03:55 Kali\x20Live -> ../../sr0

root@kali:~# ls -l /dev/disk/by-uuid/
total 0
lrwxrwxrwx 1 root root 10 Aug 19 04:03 24D7-B629 -> ../../sdb1
lrwxrwxrwx 1 root root 10 Aug 19 03:55 8a833949-3596-4c15-932b-0573f630307c -> ../../sda1
lrwxrwxrwx 1 root root 10 Aug 19 03:55 ebfc84f5-4e38-47ab-b451-2f683c549b6d -> ../../sda5
```

- It is also possible to generate your own UUIDs:

```
bash-4.1$ uuidgen -t
5c15d34e-879c-11e7-9cd3-2c44fd18e75f
bash-4.1$
```

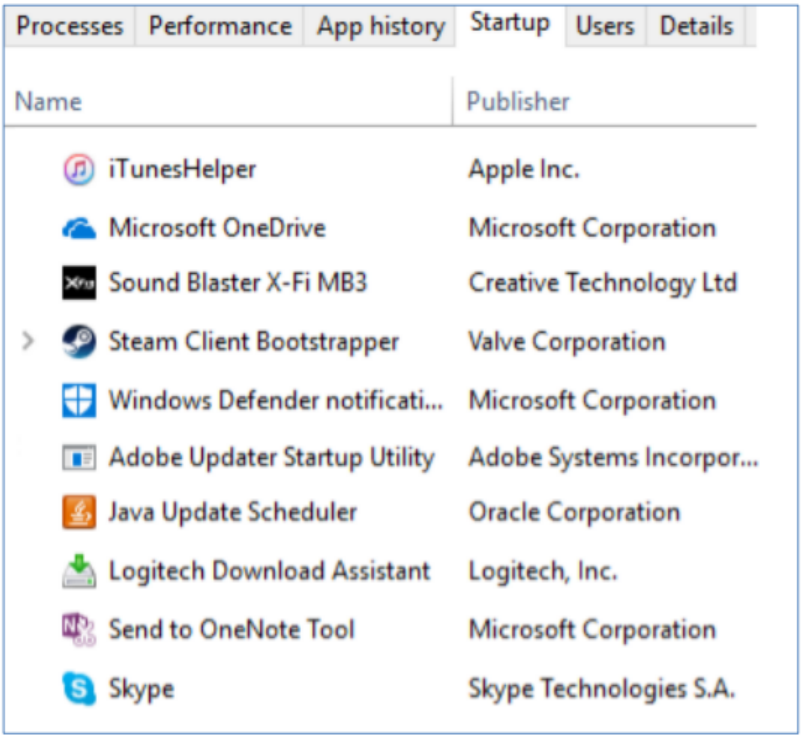
Identifying Registry Keys of Forensic Interest

Useful Registry Keys

Windows GUIDs

- **Windows 7**
- {F4E57C4B-2036-45F0-A9AB-443BCFE33D9F}
- {CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}
- **Windows 8**
- {FA99DFC7-6AC2-453A-A5E2-5E2AFF4507BD}
- {F4E57C4B-2036-45F0-A9AB-443BCFE33D9F}
- {F2A1CB5A-E3CC-4A2E-AF9D-505A7009D442}
- {CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}
- {CAA59E3C-4792-41A5-9909-6A6A8D32490E}
- {B267E3AD-A825-4A09-82B9-EEC22AA3B847}
- {A3D53349-6E61-4557-8FC7-0028EDCEEBF6}
- {9E04CAB2-CC14-11DF-BB8C-A2F1DED72085}

Autostart in Task Manager



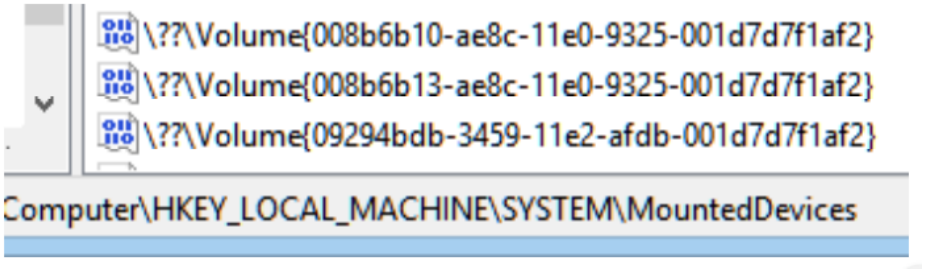
Autostart/Autorun - Registry

```
HKEY_LOCAL_MACHINE\SOFTWARE
\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\SOFTWARE
\Microsoft\Windows\CurrentVersion\RunOnce

HKEY_LOCAL_MACHINE\SOFTWARE
\Wow6432Node\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\SOFTWARE
\Wow6432Node\Microsoft\Windows\CurrentVersion\RunOnce
```

Disk GUIDs

- Kept in the registry to map drive letters (C:)

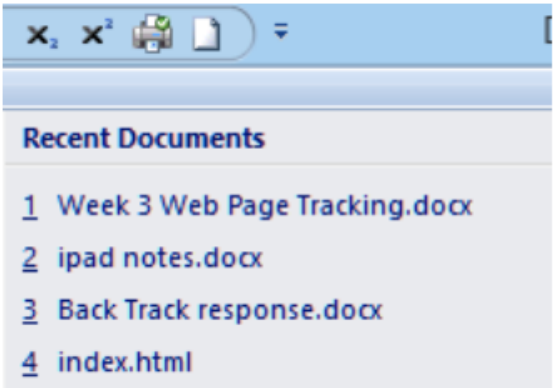
HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices



MRUs

Windows keeps several *Most Recently Used lists (MRUs)*. This includes:

- Apps started
- Web Pages visited
- Office docs opened



These indicate what the suspect did recently.

MRUs can be found in the registry.

Time Zone

It is advisable to know the time zone of when the suspect's disk was seized. Thus, you can then build a timeline around the suspicious event.

Time Zone information is stored in the registry path *HKLM\SYSTEM\CurrentControlSet\Control\TimeZoneInformation.*

TimeZoneKeyName	REG_SZ	AUS Eastern Standard Time

The USBStor Key

Records every device connected by USB, and is backed up at each restore point.

Stored in the registry path *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR*

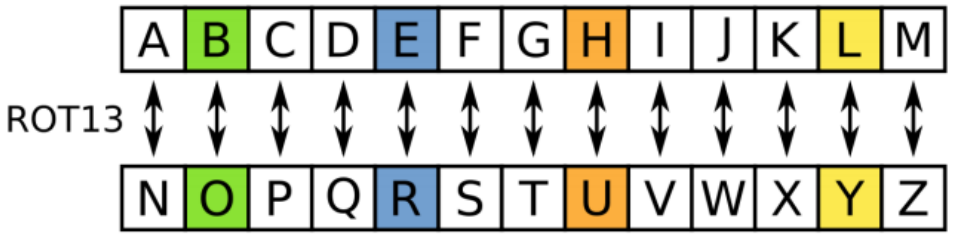
Userassist Keys

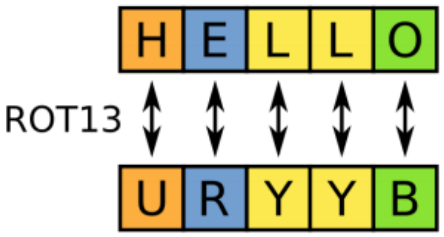
- User Assist tracks programs executed.
- The count and the last use date are stored.
- However, they do not count *.exe* files run from the command line.

Stored in: *HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist*
This key is ROT13 encoded.

```
pzq.rkr = cmd.exe, ertrqvg.rkr = regedit.exe
\Npprffbevrf\Cnvag.yax = \Accessories\Paint.lnk
```

ROT13 = Rotate by 13 Characters





Sample UserAssist Keys

```
PnabavpnyTebhcYvzvgrq.HohaghbaJvaqbjf_79euxc1saqtfp!hohagh
CanonicalGroupLimited.UbuntuonWindows_79rhkp1fndgsc!ubuntu
```

```
{6D809377-6AF0-444B-8957-A3773F02200E}\Wireshark\Wireshark.exe
{6Q809377-6NS0-444O-8957-N3773S02200R}\Jverfunex\Jverfunex.rkr
```

```
C:\Users\graha\Desktop\putty.exe
{7C5A40EF-A0FB-4BFC-874A-C0F2E0B9FA8E}\Nmap\zenmap.exe
{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\msiexec.exe
P:\Hfref\tenun\Qrfxgbc\chggl.rkr
{7P5N40RS-N0SO-4OSP-874N-P0S2R0O9SN8R}\Aznc\mraznc.rkr
{1NP14R77-02R7-4R5Q-O744-2RO1NR5198O7}\zfvrkrp.rkr
```

Week 7 - Windows Artifacts

Understanding Windows Artifacts

Scenario

We are asked to examine a digital device that we suspect has been involved in an attack.

- We suspect that there may be evidence left + traces of any malware that was used.
- We wish to capture the evidence immediately.

First, the *volatile* evidence is captured, and then the *non-volatile evidence.*

Device Variation

Each device has completely different artifacts.

OS

- Windows
- Apple
- Mac OS X
- Android

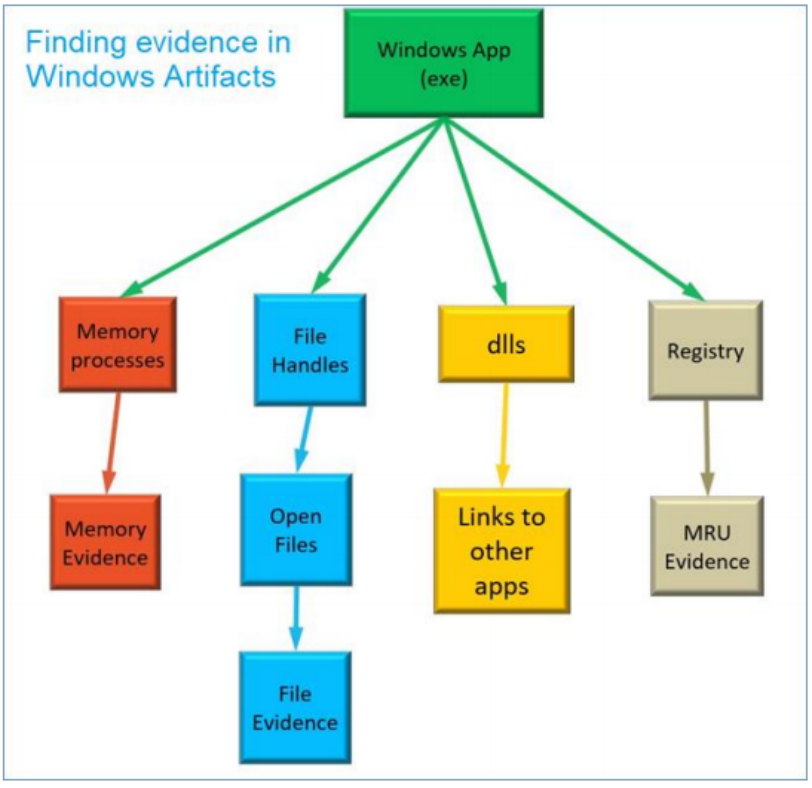
Virtualisation

- Native Host
- Virtual Machine
- Cloud-based services

Installed Apps

- Browsers
- MS Office
- VPN technologies

Evidence in Windows Artifacts



Using the Web Client for Evidence
We use a browser to identify the device - *device fingerprinting*. The HTTP request string is an example.

Browser Characteristic	bits of identifying information
User Agent	10.14
HTTP_ACCEPT Headers	9.55
Browser Plugin Details	15.38
Time Zone	7.15
Screen Size and Color Depth	4.5
System Fonts	19.08
Are Cookies Enabled?	0.43
Limited supercookie test	0.96

Windows Profiling
An important forensics process where we collect state information from *normal* behaviour, and *abnormal* behaviour is considered as being of forensic interest.

- Collecting and averaging behaviour for a variety of combinations
- Varying:
    - browsers
    - applications
    - users
    - time of day

Windows Artifact Tools

- *WMI (Windows Management Instrumentation)* to scan a PC and determine its configuration
    - i.e. *wmic bios get serialnumber*
- Additionally, we can use *Python* or *Windows PowerShell* to run commands.
- Forensic tools can also be used:
    - OSForensics
    - ProDiscover
    - Autopsy
    - Encase

Identifying Volatile Forensic Data

Basics - Volatile Forensics
Examiners use a routine in their initial investigation - a profile check to detect unusual artifacts.

- Date and Time
- Current network sessions
- Running Processes
- Prefetch activity

Volatile Evidence Collection Items

- *Date and Time* of the investigation
    - Easy to obtain from built-in Windows commands
    - Includes the current time zone
- Checking current *network connections*
    - Using the built-in *netstat* command
- We will see many connections

- Browsing and cloud services
- How do we know which ones are normal?

Open TCP and UDP ports

- *Netstat* shows open ports *listening*
  - Forensic tools are then used to link the open ports to the executable program that initially launched them.
- *.Exe* files are examined to see if they have been altered.
  - How are they examined?
    - Looking at the file publisher information
    - Looking at the *published file hash sets*
    - Some forensic tools have a copy of these hashes in an SQLite database.

Netstat on Win10 - idle, no user apps open:

```
Netstat on Windows 10 (idle)
---------------------
C:\WINDOWS\system32>netstat -bno
  Proto  Local Address          Foreign Address        State           PID
  TCP    10.10.10.3:19702       111.221.29.162:443     ESTABLISHED     10548
 [OneDrive.exe]     MIcrosoft cloud file hosting service
  TCP    10.10.10.3:19724       111.221.29.106:443     ESTABLISHED     3476
  WpnService         Windows push notification service
 [svchost.exe]
  TCP    10.10.10.3:19797       111.221.29.254:443     ESTABLISHED     3216
  DiagTrack          Diagnostic Tracking service
 [svchost.exe]
-----------------
nslookup 111.221.29.xxx
Name:    xxx.wns.windows.com
-----------------
```

Processes, Services and DLLs

These are of forensic interest when chasing malware.

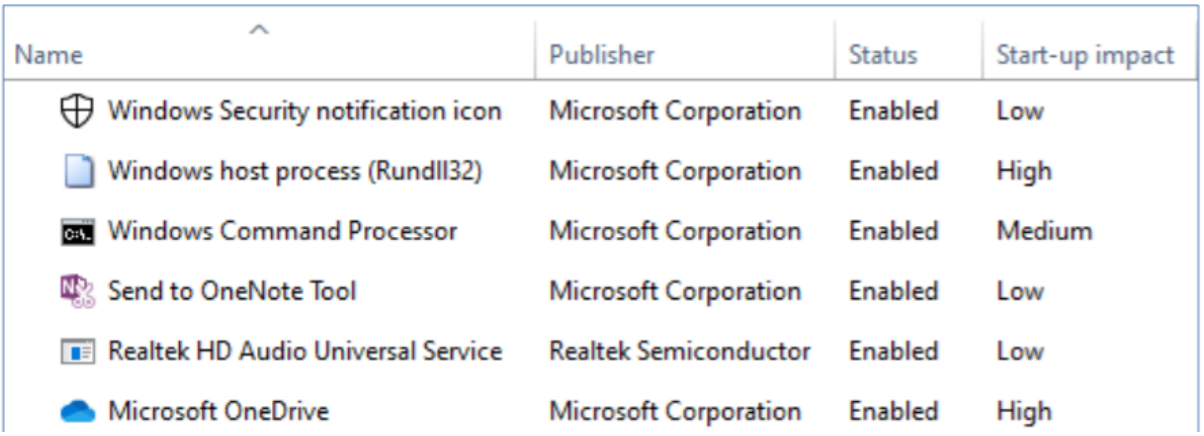
To observe processes, services and DLLS, the *pslist* and *listdlls* tools are used. We search for two things:

- Strange process names
- Strange *.exe* locations

Viewing DLLs

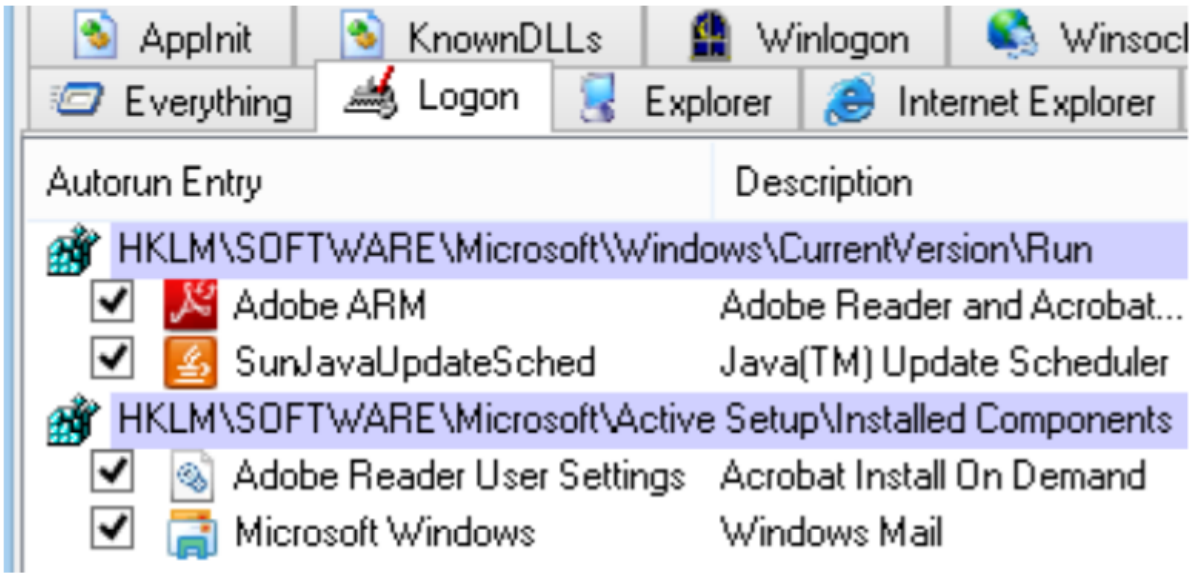
```
C:\Forensics_Graham>Listdlls.exe cmd.exe

Listdlls v3.2 - Listdlls
Copyright (C) 1997-2016 Mark Russinovich
Sysinternals

------------------------------------------------------------------------------
cmd.exe pid: 8800
Command line: "C:\WINDOWS\system32\cmd.exe"                    dll description

Base               Size      Path
0x0000000057960000  0x68000   C:\WINDOWS\system32\cmd.exe          Windows Command Processor
0x00000000a71b0000  0x1f9000  C:\WINDOWS\SYSTEM32\ntdll.dll        NT Layer dll
0x00000000a66e0000  0xbc000   C:\WINDOWS\System32\KERNEL32.DLL     Windows BASE API Client dll
0x00000000a4b20000  0x2cc000  C:\WINDOWS\System32\KERNELBASE.dll   Windows BASE API Client dll
0x00000000a6380000  0xa1000   C:\WINDOWS\System32\msvcrt.dll       Windows C Runtime dll
0x00000000a69b0000  0x356000  C:\WINDOWS\System32\combase.dll      MS COM for windows
0x00000000a4f50000  0x100000  C:\WINDOWS\System32\ucrtbase.dll     C run time library
0x00000000a6fe0000  0x11b000  C:\WINDOWS\System32\RPCRT4.dll      Remote Procedure Call run time
0x000000008e1f0000  0x37000   C:\WINDOWS\SYSTEM32\winbrand.dll    Windows Branding
0x00000000a5230000  0xad000   C:\WINDOWS\System32\shcore.dll      ?
0x00000000a5ae0000  0x9b000   C:\WINDOWS\System32\sechost.dll     Host for SCM/LSA lookup

C:\Forensics_Graham>Listdlls.exe cmd.exe | find /c "dll"        There are 11 dlls in cmd.exe
11
```

AutoStart/Autorun



Name	Publisher	Status	Start-up impact
Windows Security notification icon	Microsoft Corporation	Enabled	Low
Windows host process (Rundll32)	Microsoft Corporation	Enabled	High
Windows Command Processor	Microsoft Corporation	Enabled	Medium
Send to OneNote Tool	Microsoft Corporation	Enabled	Low
Realtek HD Audio Universal Service	Realtek Semiconductor	Enabled	Low
Microsoft OneDrive	Microsoft Corporation	Enabled	High

Alternatively, the *SysInternals Autoruns* tool can also be used:



Prefetch

When an app runs, it needs various objects loaded into memory.

Prefetch collects this information and *preloads* these objects for the next time the app starts.

- Kept in *C:\Windows\prefetch*

```
prefetch file name      | times ran | last run | path\appname
IEXPLORE.EXE-4B6C9215.pf | 139       | 11/11/13 | \INTERNET EXPLORER\IEXPLORE.EXE
WINWORD.EXE-7D220BFE.pf  | 113       | 11/11/13 | \MICROSOFT OFFICE\OFFICE14\WINWORD.EXE
ACRORD32.EXE-D066635E.pf | 111       | 11/11/13 | \ADOBE\READER 11.0\READER\ACRORD32.EXE
```

- The hash includes the name, date and file path.
- Provides evidence of when an app was used, as well as how often it was opened.

Non-Volatile Forensics

Basics

Examiners run a profile check to detect unusual artifacts.

- OS Patch level
- Browser add-ons
- User Accounts
- Timelines
- MRUs
- Registry
- Restore Points
- Logs

Collecting System Data

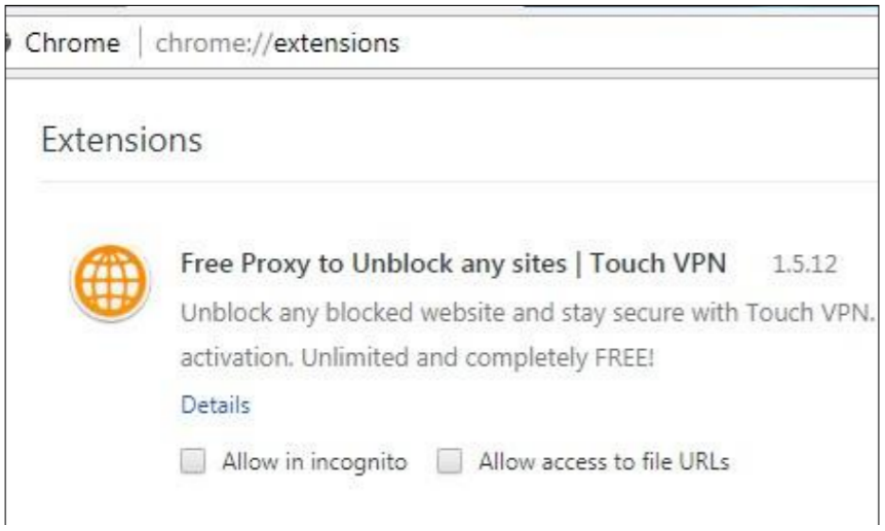
- The attacks possible on a device depend heavily on which OS patches have been applied.
- This includes patches for applications i.e:
    - Browsers
    - Office

We use the forensic tool *PsInfo* or similar.

Browser Add-ons

- Customised browsers can reveal a lot about the suspect.
- Check for:
    - Anonymous proxies
    - VPNs
    - TOR

Chrome Extensions



Viewing User Accounts with WMIC

- *WMIC - Windows Management Instrumentation Command*
- Can see Windows Internals
    - *wmic alias list brief* - show all available commands
    - *wmic useraccount list brief* - show *common* item headings
    - *wmic useraccount get disabled* - show *selected* items

```
wmic alias list brief
FriendlyName
------------
NICConfig
SysDriver
TapeDrive
NTEventLog
UserAccount
```

```
wmic useraccount get disabled, name
Disabled  Name
TRUE      Administrator
TRUE      DefaultAccount
FALSE     graha
FALSE     group11
TRUE      Guest
TRUE      WDAGUtilityAccount
```

Find the last login for a user
Use a *pipe* (|) to pass the output of *net user* into *find*.

```
C:\Users\graha>net user group11 | find "Last"
Last logon                     9/01/2018 4:31:10 PM

C:\Users\graha>net user graha | find "Last"
Last logon                     Never
```

If the answer is *Never*, the user logged in is using a Microsoft cloud account.

Timelines

- Timelines track the Incident events step by step.
- You may find suspicious events in a log file.
- Other evidence may point to the suspect's activity around this time.
- It is of forensic interest to assemble all activity around this time.
    - On the PC, network and phones
    - You must allow for different server Time Zones.

Collecting a Time Line
Previous investigations will reveal the date and time of attacks. We can collect date and time info about every file on the device.

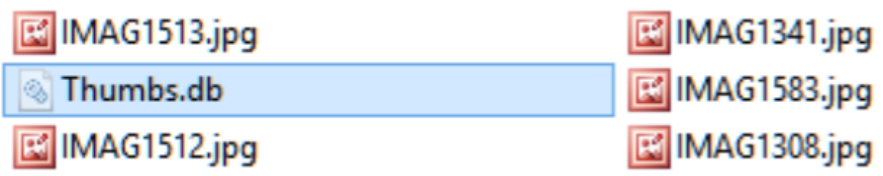
We can then examine the files in use during the attack.

There are three dates for each file:

- Created
- Modified
- Opened

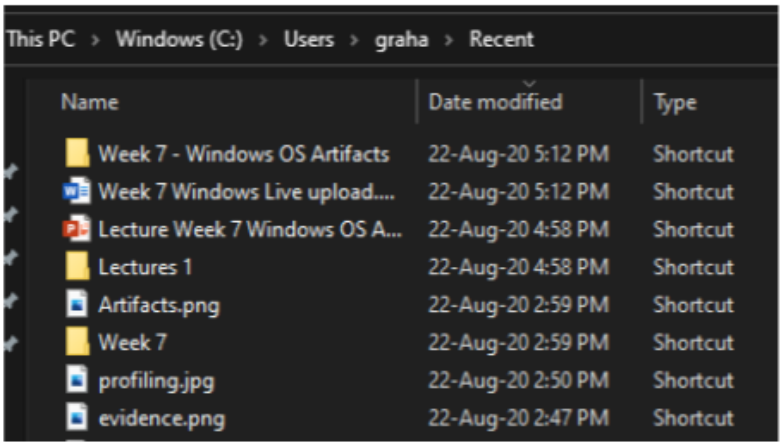
A Linux utility called *find* is used to examine file data. This is then exported to Excel for sorting.

Thumbnail Caches
Windows can create a *Thumbs.db* of image files in each directory for quick viewing.



Deleting an image does *not* delete its entry in thumbs.db - hence, this can be used for forensic evidence.

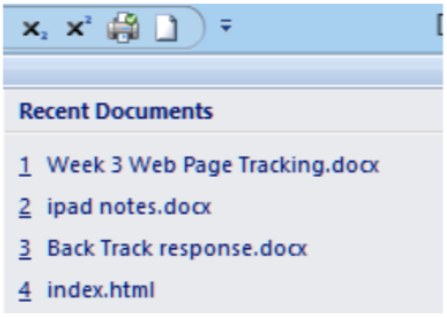
Recent Files
A list of recently opened data files and folders can be found in *C:\Users|<name of user>\Recent*.



To see recently used apps, use *UserAssist*.

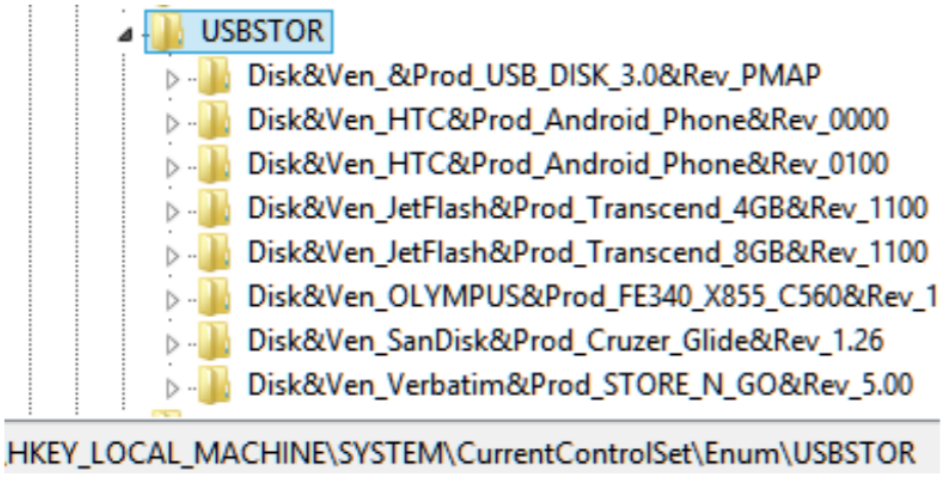
The Windows Registry
- AutoStart/AutoRun
- UserAssist - Records the number of uses of certain *.exes*
- USBStor - Records USB devices used
- List of *Most Recently Used (MRU)* items

MRUs
Keeps track of:
- Files opened
- Apps started
- Web Pages visited
- Office docs opened



These all indicate what the suspect did recently.

The USBStor key
Records every device connected by USB, and is backed up at each restore point.



USB Oblivion

A caveat to using the Windows Registry USBStor key to find digital evidence is the fact that suspects can use certain tools to remove most traces of USB usage from the registry.

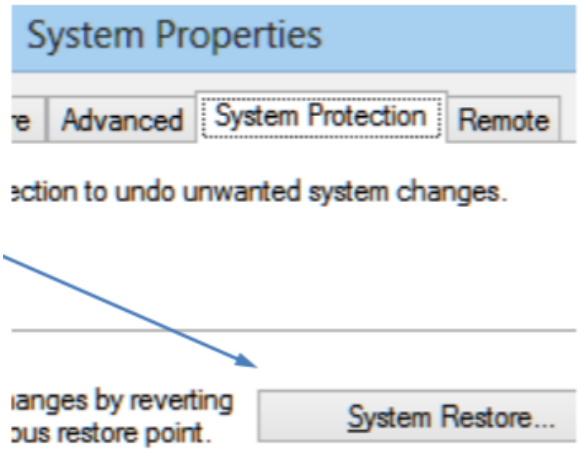
- This tool is called *USB Oblivion.*

The act of running this tool is forensic evidence against the suspect.

Restore Points

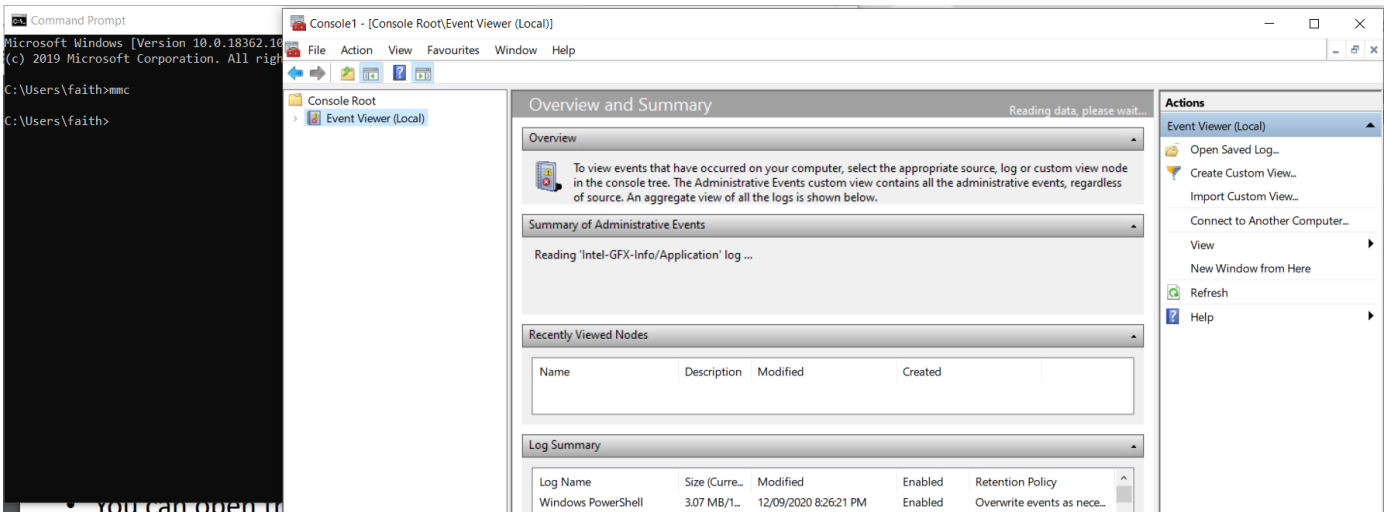
Saves a snapshot of registry and system configurations. Used before trying something dangerous.

- Can rollback if something goes wrong.
- Find Restore in System Properties
- Can recover deleted apps and registry keys



Windows Logs

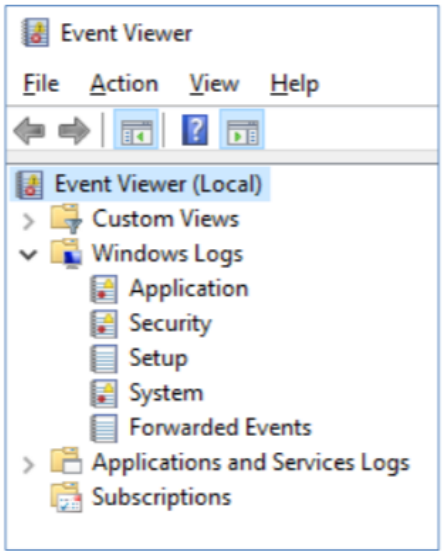
- Integrated into the Operating System
- They come with their own GUI viewer
    - Runs as the *Event Viewer* snap-in for the MMC - *Microsoft Management Console*



Windows logging system

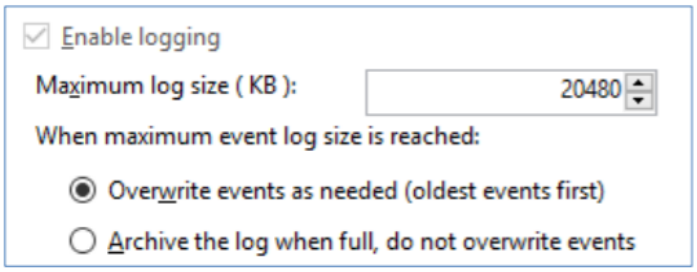
There are three main logs:

- Application
- Security
- System



Not all logging is enabled by default.

Logs default to 20MB and then roll over. (Right click and select properties)



Computer Profiling

Basics

Once we have examined a device's artifacts and its forensics data, we can reconstruct the user's activity. From this activity, we can abstract a view of the user.

This is called *computer profiling.*

- User level view of the device
- We use this computer profile to confirm or deny allegations about the user.
- When we have a new device to examine, we can use previous profiles to focus on key areas of investigation.

Hypothesis Testing

Using the computer profile, the investigator hypothesizes an action by the subject.

- E.g. downloading a pornographic image.
    - The hypothesis is then tested using forensic examination
    - Investigator's job is to attribute the download to one particular person.

Some computer profiles

- *Innocent*
    - Nothing to see, 'as new' install.

- *Media professional*
    - Image manipulation
    - Heavy social media activity
- *IT Professional*
    - Use of Linux, VMs and VPNs
- *Hiding from forensics*
    - Dark web
    - Metadata scrubbing
    - Secure deletion
- *Logons detected*
    - Private (home)
    - Work (company)
    - Educational (school, university)
    - Restricted (dark web)
- *Other people - non login*
    - Contacts (i.e. friends in divorce investigations, customers in illegally obtained data sales)
- *Apps installed*
    - Photo manipulation (photoshop, GIMP)
- *Incognito Browsers and search engines used*
    - Chrome Incognito
    - Duck Duck Go
    - TOR browser
- *Linux VMs installed*
    - Ubuntu
    - Kali
- *Use of VPNs*
    - OpenVpn
    - TOR

Week 8 - Linux Artifacts

Understanding Basic Linux

Linux OS components

The Kernel

- Talks to the CPU and Hardware
- Modular

Operating System Tools (GNU)

- Compilers and libraries
- Shell and command line tools

User interface (environment)

- GUI
- Touch

Applications

- Do useful things like run a web server

Linux OS Distributions (distros)

Assembling the components in a particular flavour

Debian GNU (1993) descendants

- Knoppix
- Ubuntu (2004)
    - Backtrack, Kali

Red Hat (1994-2004) descendants

- Fedora (free)
- CentOS (2003)
- Red Hat Enterprise (RHEL) (mainly in USA)

Open SUSE (Novell) 1994

- Mainly in Europe

Debian Releases

4.0	Etch	2007-04-08	Etch, the Etch-A-Sketch
5.0	Lenny	2009-02-14	Lenny, the binoculars
6.0	Squeeze	2011-02-06	Squeeze toy aliens
7	Wheezy	2013-05-04	Wheezy the penguin
8	Jessie	2015-04-26	Jessie the cowgirl
9	Stretch	2017-06-17	Rubber octopus from Toy Story 3
10	Buster	2019-07-06	Andy's pet dog
11	Bullseye	Not yet released	Woody's horse
	Sid	"unstable"	The next door neighbour

Downloading OS Distros

Usually downloaded as a DVD Image (.iso) file or a VM.

Debian

- Includes Kali, Knoppix and Ubuntu
- Use the *apt* package manager

Fedora

- Includes CentOS, RHEL
- Uses the *yum* package manager

Apt and Yum communicate with official Linux servers to retrieve package updates for the user's installed distro.

Linux user interfaces

The GUI interfaces

- Use *X* Windows (also called X11) for bit mapped displays
- KDE (Windows-like)
- Gnome (Apple-like)
- Blackbox (minimal X11)

The CLI interface

- Command line interface
- Like the Cisco CLI and Windows CMD
- Use a shell, usually *bash*

Linux Applications

- Usually issued as a *package*
- Downloaded and installed using a package manager.

The package manager also specifies a package format:

- *RPM* for Red Hat
- *APT* for Debian using the *dpkg* format

In regards to package *licencing*:

- All code must be compatible with version 2 of the *GNU General Public License (GPLv2)*
- All code must be signed
- All commands support the - *-help* option

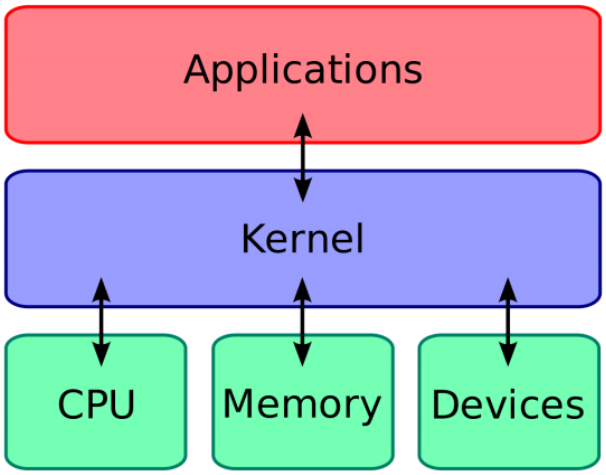
Note that small companies often use Linux as the server platform (web server, mail server, FTP server, DB server). Thus, it is a target for application-based attacks.

Linux Security Features

- *iptables* - the Linux Firewall
- Linux also implements Virus Scanners, such as *clamav*.
- Use of Private Key Certificates for networking
    - OpenSSL
    - SSH
    - OpenVPN, etc
- Sudo (superuser do)
    - privilege escalation as required
    - Must only be used when explicitly needed.

Understanding the Linux System

The Linux Kernel



The WSL Kernel (for Windows)

Here, the Kernel is modified to run on Windows - *Windows Subsystem for Linux (WSL)*

The Linux System

Bootloader

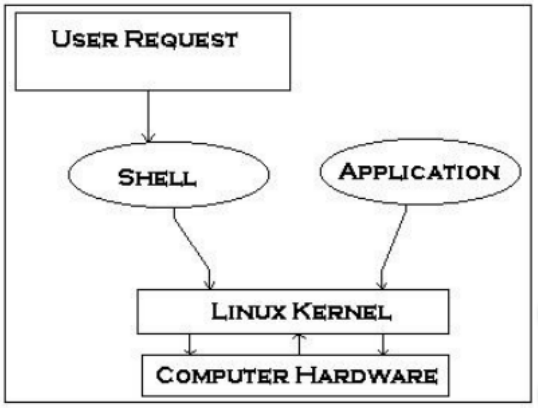
- GNU, GRUB or LILO - various types of loaders
- Loads the kernel from a file into RAM.

Init

- The top of the process tree, launched by the kernel
- Launches other processes

Libraries

- Contain code used by other processes
- GNU C library (*glibc*)
- *Like DLLs in Windows*



The GNU C Compiler (gcc)

- Compiles and links the programs used by Linux.

User Interface

- A shell or GUI

Shells

- The CLI to talk to the kernel
- Takes commands from *stdin*
- Supports regular expressions
- Keeps a history

Shell Versions

Bash

Bourne Again Shell, part of GNU Linux

Csh

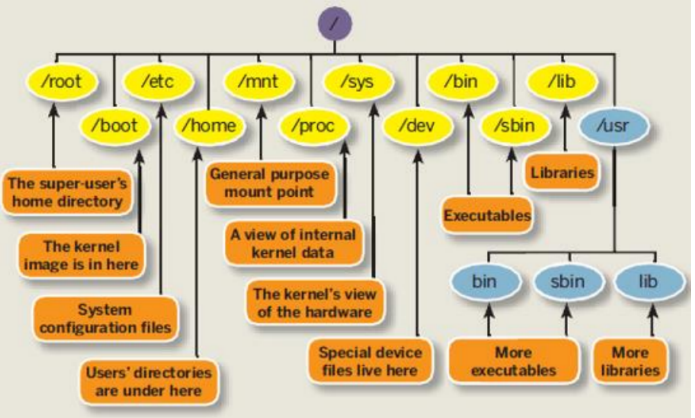
The C shell - uses C syntax

Sh

- System default shell
- Often the *Dash* shell for speed (*Derived Almquist Shell*)

Other Shells

- Korn Shell
- Tenex C shell (*tcsh*)



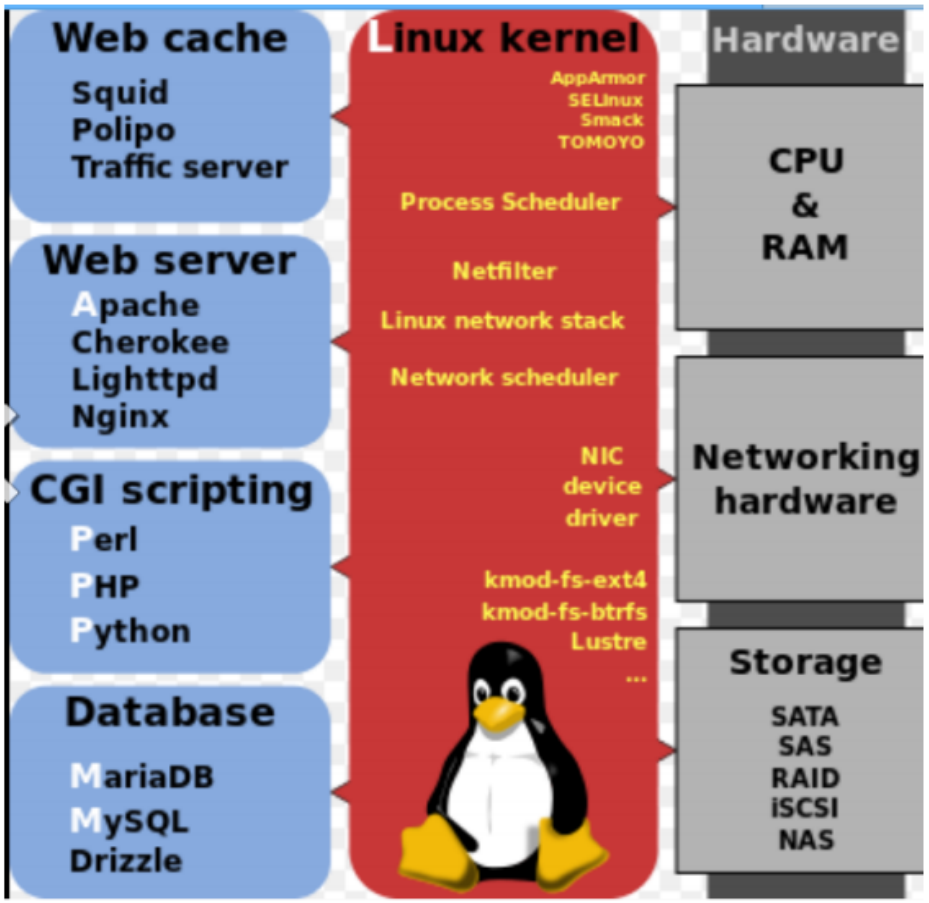
Linux File Structure

- */* - root
- */bin* - binaries (executables)
- */lib* - libraries (DLLs)
- */dev* - devices (disks, RAM, USB)
- */etc* - config file (registry)
- */home* - user folders
- */var* - logs, swap files, mailboxes, caches

LAMP

Many small Linux Servers run *LAMP*. It is an open source solution stack that works - high availability, heavy duty.

- *Linux (L) OS*
  - Debian and Ubuntu share 60% of Linux web servers
- *Apache2 (A) Web Server* (56%), some are NGINX (25%)
- *MySQL (M) Database* - also PostgreSQL
- *PHP (P)* Dynamic Web Pages - also Perl



Locating Volatile Evidence

Linux forensic shell commands

- Many tools are similar to the Windows suite.
    - Netstat, ifconfig (ipconfig), date, ping
- Some are Unix based
    - *ps* (process), *df* (mount points), *du* (disk usage)
    - *uname* (OS version), *w* (logged on users)
- Some require root/administrator privilege (sudo)
    - *fdisk* (similar to Windows Registry)*, crontab, viewing logs*
- Some will not work on Linux VMs such as WSL.

Sudo

- SuperUser Do (sudo)
- Ubuntu supports restricting dangerous commands to the SuperUser called *root*
- To run a root command as User, just prefix *sudo* before the command.
    - *sudo ifconfig*

Who can Sudo

- Root is *su,* so root can sudo.
- To see who else can *sudo*, look at the sudo group.

Volatile Evidence

Users logged on remotely

- *-w*

Running processes

- *ps -af*
    - Local processes
- *ps -Af*
    - System processes

Services

- *service - -status-all*
- *ls /etc/init.d*
    - Startup processes

Folders of Forensic Interest

- */etc/passwd* - usernames (who has a password, but not the actual passwords)
- */etc/shadow* - password hashes
- */etc/init.d* - services
- */var/www* - web server pages
- */var/log* - log files
- */var/lib/mysql* - database data files, also PostgreSQL

Locating Non-Volatile System Evidence

Basics

Linux keeps memory information in */proc*. This is a virtual folder - a link to memory.

- */proc/cmdline* shows how the boot image is loaded
    - A.k.a - the boot file name.
- */proc/cpuinfo* shows some CPU details.
    - CPU model, speed and flags.
- */proc/meminfo* shows the memory manager details.
    - Total memory, swap file details, Virtual Memory details

/proc/ examples

```
$ cat /proc/cpuinfo | grep processor
processor       : 0
processor       : 1
processor       : 2
processor       : 3

$ cat /proc/cpuinfo | grep 'model name' | uniq
model name      : Intel(R) Core(TM) i5-7400 CPU @ 3.00GHz

$ cat /proc/meminfo | grep Mem
MemTotal:       8259836 kB
MemFree:        4543520 kB
```

Locating Non-Volatile Log Files

Linux log files

The main logs are in */var/log*.

Apache2 logs (web server access)

- access.log
  - Web visitors
- ssl-access.log
  - SSL visitors

Authentication logs

- *auth.log*
  - In situations where users require a password to be authorized - i.e. SSL or SSH.

Mail logs

- *mail.log*

Database logs

- *mysql.log*

Line-printer logs

- *lpr.log*

Application logging

- *Syslog*

journalctl

Modern Linux uses *systemctl* instead of *init.d* to launch processes.

Systemctl has its own log - *journalctl*. Check with *hostnamectl* for device details.

```
ubuntu$hostnamectl
   Static hostname: ip-172-26-5-152
         Icon name: computer-vm
           Chassis: vm
        Machine ID: 0eaed5dbe31548e38562f54383ed1376
           Boot ID: 8fbc0ef2f0fa497fb36226922559bc0a
    Virtualization: xen
  Operating System: Ubuntu 16.04.3 LTS
            Kernel: Linux 4.4.0-1049-aws
      Architecture: x86-64
```

Journalctl Options

- *journalctl | grep -i GHz*
  - Look for CPU info
- *journalctl | -u ssh*
  - Look for unit
- *journalctl -t sshd*
  - Look for syslog identifier
- *journalctl -t dhclient | grep bound*
  - To interface
- *journalctl -p 3*
  - Set warning level
- *journalctl - -since -1w*
  - Open archive, w = week

Obtaining Forensic Findings from a User Logon

- Identify the usernames - *SID*
  - *SID* - security identifier.
  - When the user requests access to a resource, their Security Identifier is checked and access is denied or granted depending on the SID.
- Find the user sessions - *time and PID (process id)*
- Find the log entries

Identifying the Usernames

- Users are registered in the */etc/passwd* file
  - Many are system users with no shell
- People have a shell called *bash*.
  - *Bourne-again shell (bash)* replaces the original *Bourne* shell.

```
root@kali:~# whatis cat
cat (1)              - concatenate files and print on the standard output
```

```
root@kali:~# cat /etc/passwd | grep bash
root:x:0:0:root:/root:/bin/bash
postgres:x:114:125:PostgreSQL administrator,,,:
group11:x:1000:1000:,,,:/home/group11:/bin/bash
```

Find the user sessions

To see the logged in sessions, use *last*.

```
root@kali:~# whatis last
last (1)             - show listing of last logged in users
```

```
root@kali:~# last root
root     pts/2        192.168.198.1    Tue Jan  9 23:18   still logged in
root     pts/1        192.168.198.1    Tue Jan  9 23:00   still logged in
root     pts/0        :0.0             Wed Sep 21 00:54 - down   (00:01)
root     tty7         :0               Wed Sep 21 00:54 - down   (00:01)
```

```
root@kali:~# last group11
group11  pts/0        :0.0             Tue Jan  9 22:40   still logged in
group11  tty7         :0               Tue Jan  9 22:40   still logged in
```

- A special user called *reboot*

```
root@kali:~# last reboot
reboot   system boot  3.7-trunk-686-pa Tue Jan  9 22:40 - 23:59  (01:19)
reboot   system boot  3.7-trunk-686-pa Wed Sep 21 00:54 - 00:55  (00:01)
reboot   system boot  3.7-trunk-686-pa Sat Sep 17 04:09 - 04:10  (00:00)
reboot   system boot  3.7-trunk-686-pa Sat Sep 17 04:05 - 04:09  (00:03)
```

- See *system* with -x

```
$last -x -n 20
...<output snipped>
runlevel (to lvl 5)   4.4.0-1049-aws   Sun Feb 11 08:25   still running
reboot   system boot  4.4.0-1049-aws   Sun Feb 11 08:25   still running

$last --help | grep '\-n,'
 -n, --limit <number> how many lines to show
$last --help | grep '\-x,'
 -x, --system         display system shutdown entries and run level changes
```

Find the log entries

- *cat* /var/log/auth.log | *grep* gdm
  - *Gnome display manager (gdm)* is a *local* login.

```
root@kali:~# cat /var/log/auth.log | grep gdm3:session | grep group11
Jan  9 22:40:29 kali gdm3][3129]: pam_unix(gdm3:session): session opened for user group11
```

To see *remote* logins, use ssh.

```
root@kali:~# cat /var/log/auth.log | grep sshd

Jan  9 22:40:07 kali sshd[3315]: Server listening on 0.0.0.0 port 22.
Jan  9 22:40:07 kali sshd[3315]: Server listening on :: port 22.
Jan  9 23:00:43 kali sshd[3773]: Accepted password for root from 192.168.198.1 port 2047 ssh2
Jan  9 23:00:43 kali sshd[3773]: pam_unix(sshd:session): session opened for user root by (uid=0)
Jan  9 23:18:15 kali sshd[5430]: Accepted password for root from 192.168.198.1 port 2077 ssh2
Jan  9 23:18:15 kali sshd[5430]: pam_unix(sshd:session): session opened for user root by (uid=0)
```

- DHCP activity
    - Get the current IP address details - MAC address
    - Find the log entries - MAC or ip
    - Use *ifconfig* for any network-related investigation

```
root@kali:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:b3:49:7b
          inet addr:192.168.198.128  Bcast:192.168.198.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:feb3:497b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4942 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3747 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:466146 (455.2 KiB)  TX bytes:2670120 (2.5 MiB)
          Interrupt:19 Base address:0x2000
```

- Address *offer* from the DHCP server
    - *cat /var/log/syslog | grep -i dhcpoffer*

```
root@kali:~# cat /var/log/syslog | grep -i dhcpoffer
Sep 17 04:05:21 kali dhclient: DHCPOFFER from 192.168.28.254
Sep 21 00:54:08 kali dhclient: DHCPOFFER from 192.168.3.254
Jan  9 22:40:05 kali dhclient: DHCPOFFER from 192.168.198.254
```

- We then look for a following *preinit* to *bound* operation.

```
root@kali:~# cat /var/log/syslog | grep -i -A1 preinit
Jan  9 22:40:05 kali NetworkManager[2224]: <info> (eth0): DHCPv4 state changed
Jan  9 22:40:05 kali NetworkManager[2224]: <info>   address 192.168.198.128
```

Rotating log files

- Server log files fill up quickly
- To keep old logs as long as possible, we use *rotation.*
- When a log is *full* or *expired*, a *copy* is made.
- When the copy limit is reached, new logs *overwrite* older logs.
- Typically, the log files of a busy server only last 14 days.

Logrotate

/etc/logrotate.conf looks after two log files:

- *wtmp* for logins
- *btmp* for bad (failed) logins

Other apps look after their own logs in *etc/logrotate.d*

```
$ls /etc/logrotate.d
apache2  apt   lxd           rsyslog  ufw
apport   dpkg  mysql-server  samba
```

apache2 is typical.

```
$cat /etc/logrotate.d/apache2
/var/log/apache2/*.log {
        weekly
        missingok
        rotate 13
        compress
        delaycompress
        notifempty
```

Sample logrotate activity

```
$sudo logrotate -d /etc/logrotate.d/apache2
reading config file /etc/logrotate.d/apache2

Handling 1 logs

rotating pattern: /var/log/apache2/*.log  weekly (13 rotations)
empty log files are not rotated, old logs are removed
considering log /var/log/apache2/access.log
  log does not need rotating
considering log /var/log/apache2/error.log
  log does not need rotating
considering log /var/log/apache2/other_vhosts_access.log
  log does not need rotating
not running prerotate script, since no logs will be rotated
not running postrotate script, since no logs were rotated
```

-d = debug

Zipped log files

Often, a log file is compressed to save space. Zipped log files can be accessed in *access.log.2.gz*

- The file needs to be unzipped using *gunzip* before using *cat*.

```
ubuntu$ls
access.log      access.log.3.gz
access.log.1    access.log.2.gz
ubuntu$sudo gunzip access.log.2.gz
ubuntu$ls
access.log      access.log.3.gz
access.log.1    access.log.2
```

Tarballs

A *tarball* is a group or archive of files that are bundled together using the *tar* command and have the .tar file extension.

- If your tar file is *compressed* using a gzip compressor, use this command to uncompress it:

```
$ tar xvzf file.tar.gz
Where,
  x: This option tells tar to extract the files.
  v: The "v" stands for "verbose." This option will
     list all of the files one by one in the archive.
  z: The z option is very important and tells the tar
     command to uncompress the file (gzip).
  f: This options tells tar that you are going to give  it a
     file name to work with.
```

Shell logs

An intruder or suspect may log in and open a bash shell.

- Here, they may run shell scripts with malicious intent.
- In this case, their activities are recorded in their *.bash_history* file.

```
The suspect logs in
The suspect types the following
----
ls
date
./getinfo.sh
cat info.txt
whoami
----
The suspect logs out
----
We look at the suspect's shell history file
cd /home/group11
cat .bash_history
We see all
```

Week 9 - Disk Data

Classifying Disks

Basics

Hard Disk

High capacity at a low cost

USB Flash Drives

Portable between all Operating Systems

Solid State Drive SSDs

No moving parts

Disk Blocks

- The disk is formatted into *blocks* - default is *512* bytes.
- The disk file system sees these blocks as *sectors* - default is also *512* bytes
- The file system counts these blocks using a sequential system (LBA)
- File system allocates *clusters* of sectors to a file or other disk object - default size is *4096* bytes
- Clusters are allocated by finding unused or deleted blocks
- File Table pointers keep track of the file *segments*.

File Systems and Formatting

Formatting

Low level formatting

- Place disk sectors on the disk
- Done at the disk factory

Partitioning

- Breaks the disk into sections
- Place data structures on the disk

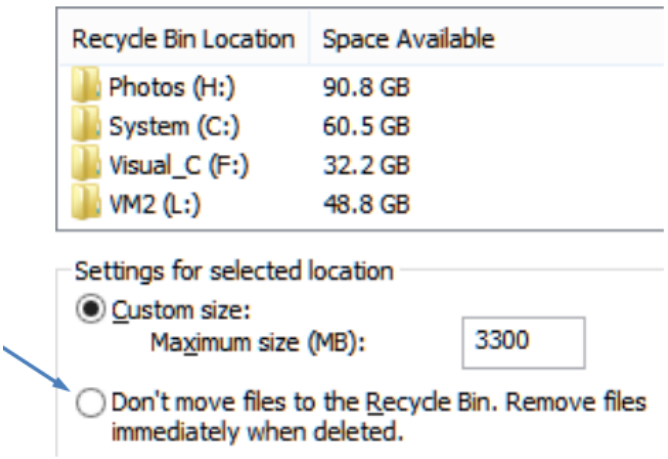
High level formatting

- Adds file structures to the partition
- Operating system dependent

The Recycle Bin

- When a file is deleted, it is moved to the *Recycle Bin* on fixed drives (not USB). It is a great source of forensic evidence.
- There is a Recycle Bin for each Drive Letter
- There is a Recycle Bin for each user
- Files can be deleted in the Recycle Bin.

Bypassing the Recycle Bin



- Deleted files can be recovered with TSK tools.
- Over time, parts of a deleted file can be overwritten by new files.

Erasing Files
- A high level format or a repartition will *not* erase data.
    - It will only remove data pointers.
- Low level formats will usually erase the data
    - Writing zeros may not destroy all previous data
    - Specific bit patterns are more effective (i.e. 01010101)
    - Some secure systems write random data
    - Writing several times improves erasure.

File Carving
- Unallocated disk space may contain fragments from previous files.
- However, the links to the parts of the file are lost.
- The file needs to be reassembled by hand (or with a tool).
- We start by searching for file headers.

USB Flash Drives
USB Flash Drives have removable read/write storage. Their available capacity is around 8 - 132GB.
- Low cost, small size, reasonable reliability
- Replace RW DVD
- Power is drawn from the host device
- Serial interface like SATA
- On chip error checking and wear leveling
    - Limited number of erases
    - Similar to SSDs
- *Typical USB 3.1 rates*: 700MB/s for sequential reads.

DRAM
Also known as *Dynamic Random Access Memory*.

Known as the ‘working memory,’ of computers, as well as the long-term memory in flash drives.

While writing data to DRAM is fast and low-energy, the data is volatile and must be continuously ‘refreshed’ to avoid it being lost.
- Inconvenient and inefficient
- Flash stores data robustly, but writing and erasing is slow, energy intensive and deteriorates it, making it unsuitable for working memory.

SSDs
- Solid State Drives (SSDs) are replacing HDDs.
- No moving parts - they are smaller, lighter and quieter
- Small form factors such as *M.2*
- Cost more - $200 for 256GB
- Uses Flash NAND chips
- Faster reads but have trouble writing
- *Triple Level Cache (TLC)* allows higher density

SSD TRIM
- Deletion is handled by the SSD controller, not the OS.
- When the file system wants to delete a file, a *TRIM* signal is sent to the SSD controller.
- If power is removed, deletion will continue when SSD power is restored, even if removed from the laptop.
- A read after *TRIM* can be set to return data *(DRAT)*
- A read after *TRIM* can be set to return zeros *(RZAT)*
    - The data may still be on the disk.

Trim Check

```
TRIM check v0.7 - Written by Vladimir Panteleev
https://github.com/CyberShadow/trimcheck

Loading continuation data from C:\Forensics_Graham\trimcheck-cont.json...
  Drive path   :  \\.\C:
  Offset       :  54400016384
  Random data  :  F0 EA D8 F2 2A 14 1F 63 AD DA 08 71 0E E3 A0 7E...

Reading raw volume data...
  Opening \\.\C:...
  Seeking to position 54400016384...
  Reading 16384 bytes...
  First 16 bytes: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00...
Data is empty (filled with 0x00 bytes).

CONCLUSION: TRIM appears to be WORKING!

Press Enter to exit...
```

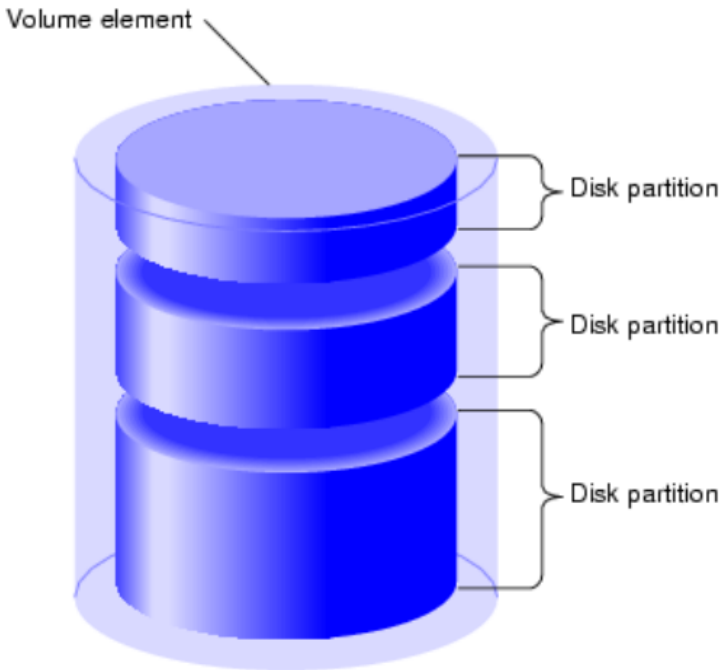
SSD Forensics

- Clearing unallocated blocks is slow
- The SSD controller performs random garbage collection independent of the file system
    - Even when disconnected from the PC
- *Wear levelling* means multiple file copies may exist - and their location is continually changing

Understanding Partitions

Disk volumes and partitions

- Disks are split into disjoint (non overlapping) *partitions*.
- Each *volume* has its own file system.



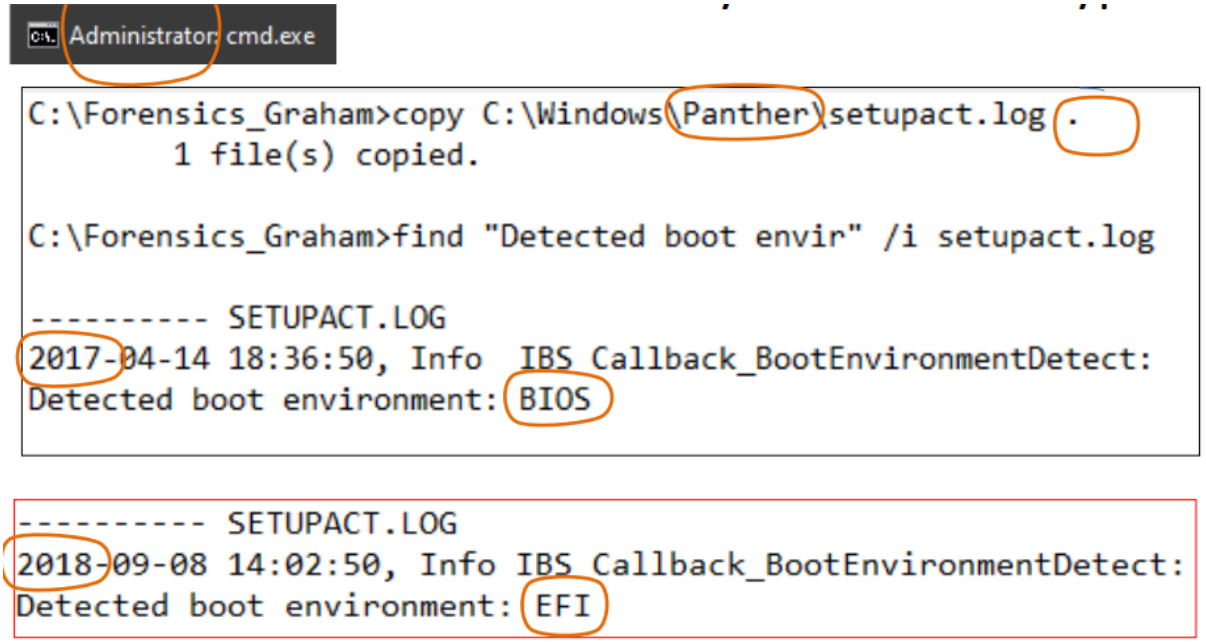
Partitioning schemes

- Partitions can be the older *BIOS* based
- Each disk may be divided into partitions
- There are four primary partitions.
    - Typically, the first partition contains the OS.
- Partitions can:
    - be the newer *UEFI* based, which is more secure, and
    - use a *GUID partition table (GPT)*.

Unified Extensible Firmware Interface (UEFI)

- Boots any OS (Windows or Linux)
- Uses a *Boot Manager* instead of the BIOS Boot Sector
- Can use an EFI system partition instead of the MBR
- CPU independent (Intel or Motorola)
- Can load the OS over a network or from USB
- Supports large disks (over 2 TB)
- The OS can talk to the UEFI once loaded.

Detecting your Disk Boot Type



Master Boot Record (MBR)

- BIOS style partitions use an MBR
- The first 512 byte sector of a disk is the *MBR*
- 440 bytes for a *boot sector* which boots to the OS in a partition. The BIOS boots to this sector
- 4 bytes (32 bits) for the disk signature
- 64 bytes for *4 partition tables*.

GUID partition tables (GPT)

- An alternate disk signature is a *GUID* as used by UEFI
    - *Global Unique Identifier (GUID)* - a random hash
    - Kept in the registry to map drive letters (C:)
    - *HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices*

Gdisk (fdisk for GPT)

```
Command (? for help): ?
b       back up GPT data to a file
c       change a partition's name
d       delete a partition
i       show detailed information on a partition
l       list known partition types
n       add a new partition
o       create a new empty GUID partition table (GPT)
p       print the partition table
q       quit without saving changes
r       recovery and transformation options (experts
s       sort partitions
t       change a partition's type code
v       verify disk
w       write table to disk and exit
x       extra functionality (experts only)
?       print this menu
```

Viewing the GPT Disk

gdisk64.exe 0: ← 0 is the first disk

```
Command (? for help): p
Disk 0:: 500118192 sectors, 238.5 GiB
Sector size (logical): 512 bytes
Disk identifier (GUID): EC7E7C0B-56E5-4F2E-B37C-DFE192FCC523
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 500118158
Partitions will be aligned on 2048-sector boundaries
Total free space is 2055106 sectors (1003.5 MiB)

Number Start (sector) End (sector)  Size     Code Name
   1            2048       206847   100.0 MiB EF00 EFI system partition
   2          206848       239615   16.0 MiB  0C01 Microsoft reserved .
   3          239616    316718762   150.9 GiB 0700 Basic data partition
   4       316719104    424648703   51.5 GiB  0700 Basic data partition
   5       426698752    498378751   34.2 GiB  0700 Basic data partition
   6       498380800    500117503   848.0 MiB 2700
```

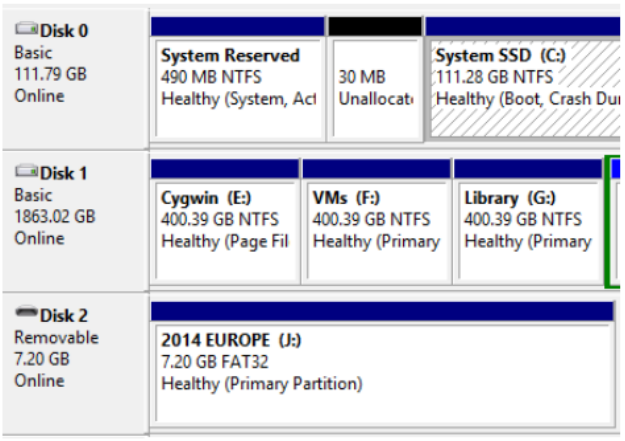
Viewing the GPT partitions

```
Command (? for help): i
Partition number (1-6): 1
Partition GUID code: C12A7328-F81F-11D2-BA4
Partition unique GUID: 0B3FFFDA-A04F-4496-B
First sector: 2048 (at 1024.0 KiB)
Last sector: 206847 (at 101.0 MiB)
Partition size: 204800 sectors (100.0 MiB)
Attribute flags: 8000000000000000
Partition name: 'EFI system partition'
```

Disks as seen by Windows

Each partition is identified by a letter C, D, E, Z, etc.

Volume	Layout	Type	File Sys
2014 EUROPE (J:)	Simple	Basic	FAT32
Cygwin (E:)	Simple	Basic	NTFS
HD Unused (I:)	Simple	Basic	NTFS
Library (G:)	Simple	Basic	NTFS
Photos (H:)	Simple	Basic	NTFS
System Reserved	Simple	Basic	NTFS
System SSD (C:)	Simple	Basic	NTFS
VMs (F:)	Simple	Basic	NTFS



Disks as seen by WMIC

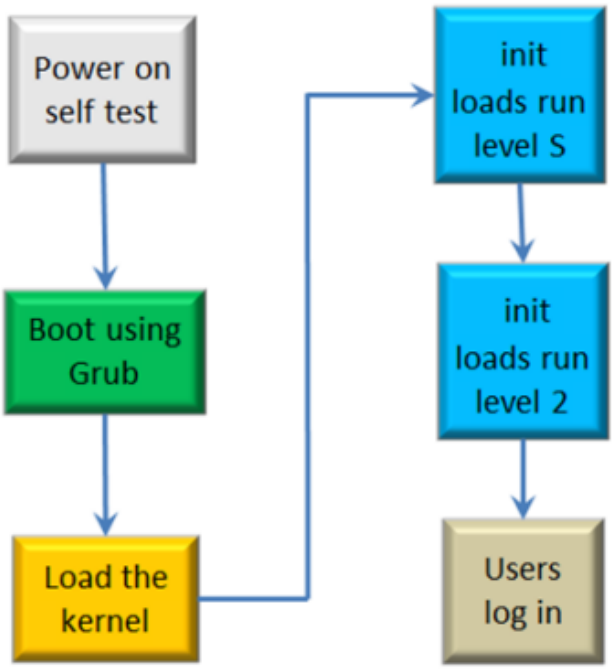
wmic diskdrive list brief

```
C:\Users\graha>wmic diskdrive list brief
Caption                  DeviceID            Model                  Partitions  Size
HFS256G39TND-N210A       \\.\PHYSICALDRIVE0  HFS256G39TND-N210A     5           256052966400
TDK LoR Platinum 3.0 USB Devi \\.\PHYSICALDRIVE1 TDK LoR Platinum 3.0 USB  2           7723537920
```

Understanding the Boot Process

Basics

- To ensure the integrity of the file system, we need to guarantee the boot process.
    - Power ON Self Test (POST)
    - Basic I/O System (BIOS)
    - File System Loader
    - Init the Operating System (OS)
    - Pass control to the OS



Secure Boot

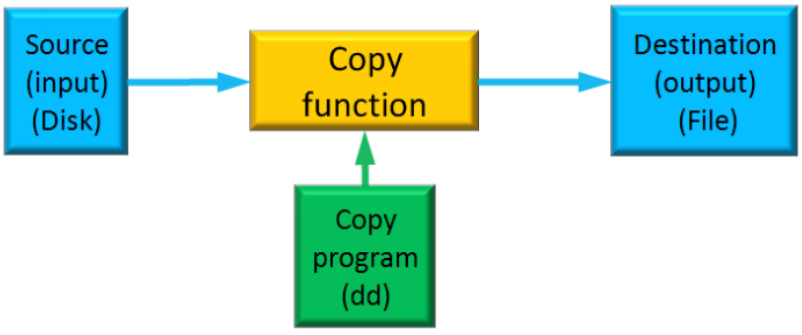
- A *Boot virus rootkit* can install and hide from the OS - very dangerous.
- Secure Boot checks a signed certificate in the UEFI.
- Microsoft own the certificate (which may be an issue for Linux)
    - Thus, the Linux distributor uses a *shim* to allow UEFI to call their boot loader.
- The Linux distributor buys a certificate from Microsoft to sign their boot loader.
- If the boot loader hash matches the certificate, it will load.

Data Acquisition Principles

Disk Acquisition Options

1. Physically extract the disk and copy it to another disk
2. Boot from a USB or CD and copy the disk
    - We can make a disk to disk copy
    - Or a disk to file copy
3. Either way, we will need to mount the evidence disk as read only and block writes from the OS.

Disk Acquisition Process



Forensic Investigation Options

1. Boot from the disk copy
2. From our forensic laptop OS, we can mount the disk
3. We can open a VM and load the disk image file.

Forensics Tool Testing

- The US National Institute of Standards and Technology (NIST) test available forensic tools
- Their approval helps get tool results accepted in court.

Becoming Familiar with Disk Acquisition Tools

Basics

- A tool will be needed to capture a disk into a file.
- Forensics tools such as ProDiscover and Autopsy for Windows have built-in disk acquisition.
- You can also use specialist tools
    - Access Data FTK Imager
- You can also use the *Linux Data Dump Tool (dd).*
    - However, this will not work on a VM such as Windows WSL.

Disk Acquisition

- We can capture the whole suspect's disk, but we will also require the same space to be free on the investigator's disk.
- Can capture a partition (Volume).
- Can be much less time and space required
- Once we have the image, we can examine it.
- Tools such as *The Sleuth Kit (TSK)* are used.
    - Built into Linux

Data Dump - dd

- Default tool for *disk to file copy*.
- Uses a raw, sector by sector copy
- Copies text, binaries and hex files all with ease.
- Will pick up hidden data.

dd --help will list all available command options.

dd variants

- *dd*
    - The original dd, part of *GNU Core Utilities*
- *ddrescue*
    - Additional support for bad sectors
- *dc3dd*
    - Support for forensics, including *on-the-fly* hashing
- *dcfldd*
    - Enhanced for government use

Understanding Partition Details

Viewing the Partition Table

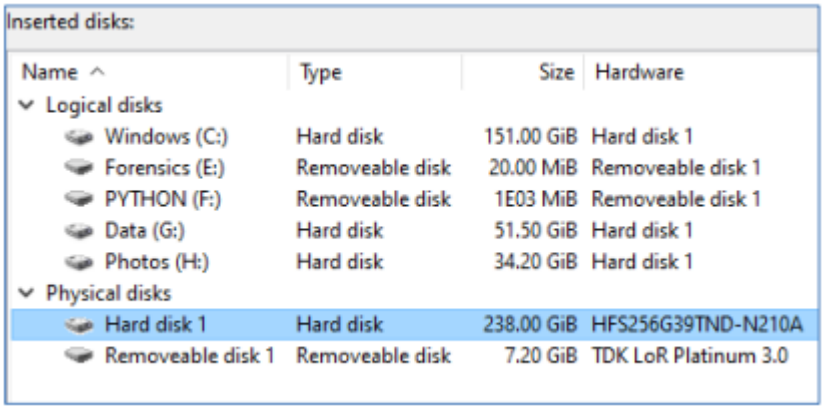
- There may be hidden partitions, so we look at the table
- We can see the live partition table
    - Windows Disk Manager
    - Or HXD Editor
- We can see the acquired image partition table
    - Extract the first two blocks with *dd* and view with *xxd*
    - Or view the first two blocks with the *HXD* Editor

Viewing the Live Partition Tables

Windows Disk Manager

Volume	Layout	Type	File System	Status
(Disk 0 partition 1)	Simple	Basic		Healthy (EFI System Partition)
(Disk 0 partition 6)	Simple	Basic	NTFS	Healthy (Recovery Partition)
Data (G:)	Simple	Basic	NTFS	Healthy (Basic Data Partition)
Forensics (E:)	Simple	Basic	NTFS	Healthy (Primary Partition)
Photos (H:)	Simple	Basic	NTFS	Healthy (Basic Data Partition)
PYTHON (F:)	Simple	Basic	FAT32	Healthy (Primary Partition)
Windows (C:)	Simple	Basic	NTFS	Healthy (Boot, Page File,

HXD Editor

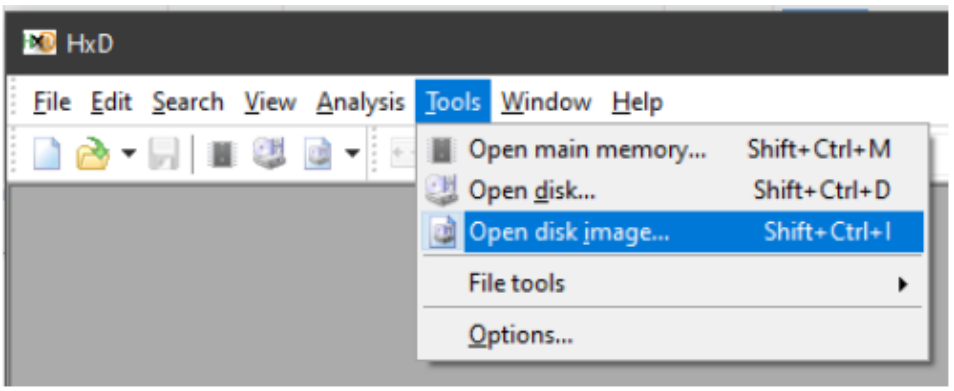


Viewing the Acquired Image USB BIOS Partition Table

dd and xxd

```
group11/mnt/c/Forensics_Graham$ dd if=USB1.001 count=1
of=USBSector3.dd
1+0 records in
1+0 records out
512 bytes copied, 0.0024833 s, 206 kB/s

group11/mnt/c/Forensics_Graham$ xxd USBSector3.dd
```

HxD

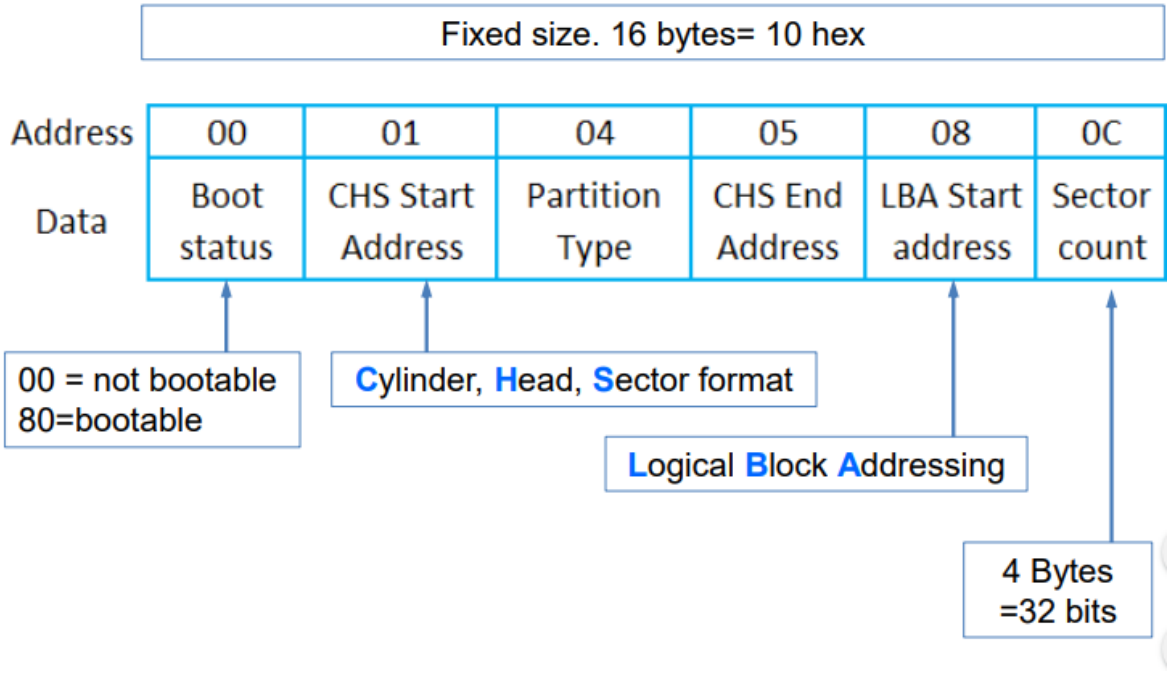


MBR (Master Boot Record) Data Structure

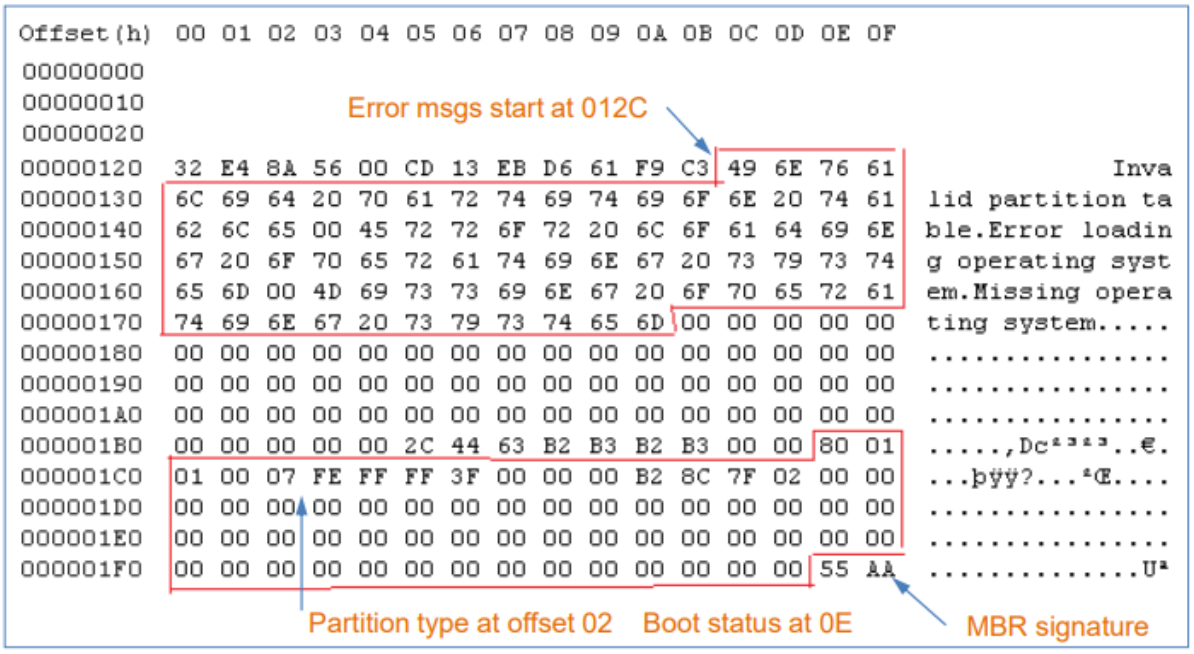
Structure of a master boot record

Address			Description		Size in bytes
Hex	Oct	Dec			
0000	0000	0	code area		**440** (max. 446)
01B8	0670	440	disk signature (optional)		**4**
01BC	0674	444	Usually nulls; 0x0000		**2**
01BE	0676	446	**Table of primary partitions** (Four 16-byte entries, IBM partition table scheme)		**64**
01FE	0776	510	55h	MBR signature; 0xAA55	**2**
01FF	0777	511	AAh		
MBR, total size: 446 + 64 + 2 =					**512**

MBR Partition Table Data Structure



Sample MBR
Note the four *partitions* starting at *01BE*.



Partition Types

- 07 NTFS
- 0B FAT32 CHS
- 0C FAT32 LBA
- OE FAT16 LBA
- 0F Extended

Other Types

- 82 Linux swap
- 83 Linux native partition (*EXT4*)

Viewing a Partition (Logical Disk)

- Drive Letters in Windows (C:, D: etc.)
- Live Partition - use *fsutil*

```
C:\Users\graha>fsutil fsinfo volumeinfo e:
Volume Name : Forensics
Volume Serial Number : 0xf0039552
Max Component Length : 255
File System Name : NTFS
Is ReadWrite
```

- Image Partition - use *fsstat*

```
$ fsstat USBFolder.001
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: NTFS
Volume Serial Number: F8F003D1F0039552
OEM Name: NTFS
Volume Name: Forensics
Version: Windows XP
```

File System Types

Basics

File Allication Table (FAT32)

- Simple, used on USBs, phones and cameras
- 4GB file size limit

Windows New Technology File System (NTFS)

- ACL permission control
- Encryption using *EFS (Encrypting File System)*
- Compression
- Quotas
- Linux mount points

FAT32

Layout

- *Sector 0* - Boot Sector
- *Sector 1* - File System Information
- *Sector 6* - Backup of Sector 0

FAT Tables (2 for redundancy)

- Contain file poitners to data sectors
- Variable length, can be many sectors

Data Region - all the rest of the partition

Contents	Boot Sector	FS Information Sector (FAT32 only)	More reserved sectors (optional)	File Allocation Table #1	File Allocation Table #2	Data Region (for files and directories) ... (To end of partition or disk)
Size in sectors	(number of reserved sectors)			(number of FATs)*(sectors per FAT)		NumberOfClusters*SectorsPerCluster

FAT sizes

- USB partitions may use FAT32
  - Minimum disk is 320MB
  - Max. disk is 32GB
- Small partitions may use FAT16
  - Minimum disk is 16MB
  - Maximum disk - 2GB

NTFS

- Only WIndows can format as NTFS (license)
- However, Linux systems can read and write to NTFS.

Basics

- Large file systems, currently cpas at 256TB
- *Reliability* - aimed at file server use
- *Recoverability* - journaling file system
- Multiple copies of the *Master File Tables (MFT)* on disk
- Hot fixing - if a bad sector is detected, the data is silently moved and the sector marked as bad.

NTFS Features

- *USN Journal*
  - records changes to files
- *ADS - Alternate Data Streams*
  - for Macintosh fork support
  - Used by malware to hide code

    - Do not copy to FAT32 partitions
- *VSS - Volume Shadow Copy*
    - Allows backup of a locked file.
- *MFT - Metadata File Tables*
    - File name starts with a $.

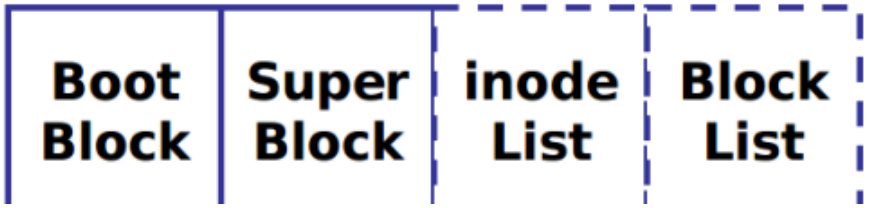
NTFS Metafiles

- Define and Organise the file system
- Hidden from the user
- *$MFT* - master list of all file names
- *$LogFile* - The NTFS Log, a transactional system to allow rollback of metadata changes
- *$Volume* - Volume description
- *$Boot* - Volume boot record
- *$BadClus* - a list of known bad clusters
- *$Secure* - the ACL database
- *$UpCase* - Uppercase name version

Segment Number	File Name
0	$MFT
1	$MFTMirr
2	$LogFile
3	$Volume
4	$AttrDef
5	.
6	$Bitmap
7	$Boot
8	$BadClus
9	$Secure
10	$UpCase

Linux File System on disk

Overview



- Boot block contains the bootstrap code as in Windows
- Superblock contains the disk metadata
- Inode contains a pointer to a data block
- Data block of 512 byte sectors

fdisk on Linux

```
root@kali:~# fdisk -l

Disk /dev/sda: 21.5 GB, 21474836480 bytes
255 heads, 63 sectors/track, 2610 cylinders, total 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000dff5c

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048    40136703    20067328   83  Linux
/dev/sda2        40138750    41940991      901121    5  Extended
/dev/sda5        40138752    41940991      901120   82  Linux swap
```

Default Linux file system - Ext4

- Kernel 2.6.28 was released with the ext4 filesystem
    - - 25 Dec 2008
- Used on Android from version 2.3
- Easy upgrade from Ext3 and Ext2
- Can handle very large files, 16TB
- It is faster as it joins contiguous blocks as an Extent
- Journal checksums for reliability
- Good for SSD

```
root@kali:~# file -sL /dev/sda1

/dev/sda1: sticky Linux rev 1.0 ext4 filesystem data,
UUID=8a833949-3596-4c15-932b-0573f630307c
(needs journal recovery) (extents) (large files) (huge files)
```

Finding Hidden Data on Disks

Viewing the FAT32 image file detail with fls

- Runs fls against the volume image.

- fls USBFolderx

r/r = file

d/d = folder

* = deleted file

```
$ fls USBFolder2.001
r/r 3:  PYTHON      (Volume Label Entry)
d/d 6:  System Volume Information
r/r * 9:        Trade_Secrets.txt
r/r 11: Sample.pdf
...
r/r 17: MS Office Meta Data.jpg
r/r * 18:       _s.exe
r/r * 19:       _s2.exe
r/r * 20:       _ogo.gif
r/r 21: test
r/r 22: test1
...
r/r 28: Flowers.txt
v/v 32636931:   $MBR
v/v 32636932:   $FAT1
```

Viewing the NTFS image file detail with fls

- Runs fls against the volume.

- fls USBFolderx

r/r = file

d/d = folder

-/r = deleted file

```
group11/mnt/c/Forensics_Graham$ fls USBFolder.001
r/r 4-128-1:    $AttrDef
r/r 8-128-2:    $BadClus
...
r/r 50-128-1:   strings.exe
d/d 36-144-1:   System Volume Information
r/- * 0:        Trade_Secrets.txt
-/r * 39-128-3: Trade_Secrets.txt
-/r * 45-128-3: logo.gif
-/r * 46-128-1: ls2.exe
d/d 256:        $OrphanFiles
```

Recovering deleted files

- Run *fls* against the Volume folder.
- Note the *inode* number for the deleted file.
- Run *icat* against the inode.

The Forensic Process, Cybercrime, Australian Law and Legal Issues

Understanding the Forensic Process

Basics

On being notified of a possible Security Breach, the investigator must decide:

- Is it a *Civil or Criminal* case?
- Who has *jurisdiction* of the case?
    - Is it in the workplace?
    - Is it on public property?
    - Has a crime been committed?

Criminal Law

- Deals with acts of intentional harm
- Such acts are offences against us all
- The offences are listed as *crimes* in a *criminal code*
- To be convicted, it must be proven that the person committed the crime (authentication)
- It must also be proven that the person *meant* to commit the crime

Standard of Proof

Beyond a Reasonable Doubt

- The judge or jury must be almost certain.

Civil Law

- Civil cases may not involve law enforcement
- Civil cases may involve a Court Order
    - For example, child custody
- The court order may impact on what the investigator can do.
    - There may be time limits on the evidence.
- Forensics procedures and techniques can vary significantly from case to case.

Standard of Proof

Balance of Probabilities

- It is more likely than not that the defendant causes harm or loss.

Crime Scene Preparation

1. *Notify* Decision Makers and Acquire Authorization
2. Risk assessment
    - Privacy issues
3. Obtain a *search warrant*
4. Issue *subpoenas*
5. Document the procedure to be followed

Search and Seizure

- A police officer may apply for a warrant to search if:
    - They have reasonable grounds for believing that there is or will be on the premises:
        - A thing connected with a particular crime
        - A thing stolen or otherwise unlawfully obtained.
- A *warrant* authorises a police officer to enter premises, and search only for those things listed
    - However, a warrant is not required to search a person or package for a dangerous article, or a thing used in a crime.

Subpoenas

- If a person refuses to produce documents or give evidence, a party may request the Court to issue a subpoena for:

    - Production
    - Evidence
    - Production and to give evidence

Evaluating and Securing the Scene

- Gather the preliminary information at the scene
- First Responder duties
    - Ensure the safety of all persons
    - Protect the integrity of all evidence
    - If it is off, leave it off - if it is on, leave it on
    - Remove all persons from the scene
    - Conduct preliminary interviews

Collecting the Evidence

- Search and Seizure
- Photograph the scene
    - Include all peripherals
- Collect Physical Evidence
    - Books, notes, passwords
- Collect Electronic Evidence
    - Look for a backdoor Trojan

Examples of things that become Evidence

- Documents detailing a Crime
- Financial documents detailing a crime
    - Orders, invoices and Receipts
- Illegal images
- Web history of visits to a website relating to a crime
- Web searches relating to the performing of a crime

Digital Evidence

Acquisition methods depend on the *type* of digital evidence.

- It can be *transient*
    - Reside in memory
    - I.e. open network ports
- It can be *fragile*
    - Easily altered
    - Date/time stamp
- It can be *active*
    - Current open files
    - Current TCP session
- It can be *archived*
    - Backup copy
- It can be *residual*
    - Fragments left after file deletion
- It can be *metadata*
    - Date about the data
    - Date/time stamps, owner, printer used
- Access can be *temporary*
    - Encrypted file system
    - Requires access to the private key/password

Acquiring a Forensic Duplication

- We need to examine the non volatile evidence stored on a hard disk, to ensure that the evidence is not *altered.*
- A *copy* of the disk needs to be analysed.
- The original disk is now *evidence* - so it must be sealed.
- A second copy may be necessary to return the device to service.

Legal Issues in taking a copy

- *Cybercrime Act 2001 section 3K*
    - Use of equipment (laptop) to examine or process things (make a copy of the suspect's disk)
    - Equipment may be brought to the warrant premises
    - Things may be moved for examination (take the device away)
    - Time limit on moving a thing - 14 days

Testifying as an Expert Witness

Definition of "Expert"

A person who, by virtue of education, training, skill or experience, is believed to have expertise and specialised knowledge in a particular subject.

They are considered so by their peers.

Only an expert witness can give an opinion in court.

A Summary of the Forensic Process

Suspicious item is found

1. What is the item?
2. How did it get there?
3. When was it placed there?
4. Who put it there?
5. Why was it placed there?

Is it forensic evidence?

Steps

1. Identification
    - Criminal or Civil?
2. Preparation
    - Warrants and Subpoenas
3. Evaluate and Secure the Scene
4. Collect the Evidence
5. Secure the Evidence
    - Make hashed copies
6. Acquire the data
    - Disk images
    - Decryption
    - Unhiding
7. Analyse the data
    - Keyword searches
8. Prepare the Final Report

Understanding Cybercrime

Facets of Cybercrime

1. Involves digital devices in the commission of a crime
    - Online fraud
    - Cyberstalking
    - Cyber terrorism
2. Directed at digital devices themselves
    - Hacking
    - Malware
    - Botnets
3. Incidental to the commission of other crimes
    - Communications about a crime
    - Purchasing stolen credit cards

Cybercrime laws

- Older Commonwealth laws refer to a *communication service.*
- Any new form of communication is a like service.

OECD countries approved the *Convention on Cybercrime* in 2004.

Australia signed the convention in 2012. The law came into force in March 2013.

Legal View of Cybercrime

- Council of Europe Convention on Cybercrime
- Three parts:
    - Access to a computer system or interception of data transmission
    - Committed intentionally
    - Illegal or without right (not permitted by the owner)

Cybercrime Motivation

Main Factors

- *Scale*: The internet can access 3 billion people
- *Accessibility*: Smart phones are everywhere.
- *Anonymity:* Proxy servers, VPNs and TOR
- *Data Portability*: A smartphone can hold gigabytes of data

Other Factors

- The availability of large numbers of *victims*
- The supply of large numbers of *offenders*
- Absence of capable *guardians*
    - No surveillance of data
    - Lack of network activity observation

Australian Law regarding Digital Forensics

Right of Privacy

- The right of natural persons to protect their personal life from invasion, and to control the flow of their personal information.
- Privacy is not an absolute
- Is balanced against other competing rights and duties.

Privacy Concepts

Information privacy

The handling of personal data such as:

- Credit information
- Medical records
- Government records

Also called data protection.

Bodily privacy

- Protection against invasive procedures such as:
    - Genetic tests
    - Drug testing
    - Cavity searches

Privacy of communications

The security of:

- Mail
- Telephones
- E-mail

Territorial privacy

- Limits intrusion into the workplace or public space.

- This includes:
    - Searches
    - Video surveillance
    - ID checks.

Australian Privacy Principles (APPs)
These became law on the 12th of March 2014.

- Companies must comply with these principles.
- Once approved, the company is given an APP code

APP 1
Open and transparent management of personal information

- APP entities must manage personal information in an open and transparent way.
- Includes having a clearly expressed and up to date *APP privacy policy*.

APP 2
Anonymity and pseudonymity

- Requires APP entities to give individuals the option of not identifying themselves, or of using a pseudonym.
- Limited exceptions apply, however.

APP 3
Collection of solicited personal information

- Outlines when an APP entity can collect personal information that is solicited.
- It applies higher standards to the collection of 'sensitive' information.

APP 4
Dealing with unsolicited personal information

- Outlines how APP entities must deal with unsolicited personal information.

APP 5
Notification of the collection of personal information

- Outlines when, and in what circumstances, an APP entity that collects personal information must notify an individual of certain matters.

APP 6
Use or disclosure of personal information

- Outlines the circumstances in which an APP entity may use or disclose personal information that it holds.

APP 7
Direct marketing

- An organisation may only use or disclose personal information for direct marketing purposes, if certain conditions are met.

APP 8
Cross-border disclosure of personal information

- Outlines the steps an APP entity must take to protect personal information before it is disclosed overseas.

APP 9
Adoption, use or disclosure of government related identifiers.

- Outlines the limited circumstances when an organisation may adopt a govt. related identifier of an individual as its own identifier, or use or disclose a government-related identifier of an individual.

APP 10
Quality of personal information

- An APP entity must take reasonable steps to ensure the personal information it collects is

APP 11

Security of personal information

- An APP entity must take reasonable steps to protect personal information it holds from misuse, interference and loss, and from unauthorised access, modification or disclosure.
- An entity has obligations to destroy or de-identify personal information in certain circumstances.

APP 12

Access to personal information

- Outlines an APP entity's obligations when an individual requests to be given access to personal information held about them by the entity.
- This includes a requirement to provide access, unless a specific exception applies.

APP 13

Correction of personal information

- Outlines an APP entity's obligations in relation to correcting the personal information it holds about individuals.

Security Example - APP 11

- A company stores their customer data with *Amazon Web Services (AWS)*
- The data gets hacked and damaging personal information is made public.
- Who is to blame?

Data Retention

Telecommunications (Interception and Access) amendment (Data Retention) Bill - March 2015

- Requires telecommunications service providers to retain telecommunications metadata for 2 years.
- Conflicts with the *Right to Privacy Act 1988*.

Phone calls

- Data not retained:
    - The conversation
- Data retained:
    - The caller number, and the called number
    - Call duration
    - Caller location
        - Exchange line for fixed lines
        - Mobile cell tower for mobile calls
    - GPS data

Emails

- Data not retained:
    - Email content
- Data retained:
    - Sender and the recipients
    - Message size
    - Sender location
    - Receipt acknowledgement
    - GPS data

Principles of computer-based electronic evidence

1. No action taken by law enforcement agencies or their agents, should change data held on a computer or storage media which may subsequently be relied upon in court.
2. In circumstances where a person finds it necessary to access original data held on a computer or on storage media, that person must be competent to do so, and be able to give evidence explaining the relevance and the implications of their actions.
3. An audit trail or other record of all processes applied to computer-based electronic evidence should be created and preserved. An independent third party should be able to examine those processes and achieve the same result.

4. The person in charge of the investigation (the case officer) has overall responsibility for ensuring that the law and these principles are adhered to.

Common Legal Issues in Digital Forensics

Case Study

- A person was charged with assaulting police after heavily armed officers ordered her out of bed in the early hours of Sep 18 2014 - largest counter terrorism raid in Australia's history
- It emerged that the woman was not named on the search warrant, and was not shown paperwork accompanying the warrant as required by law.
- While police had the authority to enter the premises, they did not have the authority to:
    - arrest
    - detain
    - or search the woman without reasonable suspicion.
- On Monday, the magistrate found that searching the woman was unlawful, and that the officers were not acting in the execution of their duties.
    - Thus, the charges against her must be dismissed.
- The woman is likely to seek an order that NSW Police pay her legal costs.

Search Warrants

- An investigator may discover something outside the warrant. It may be evidence of a reportable offence.

Adverse Interference

- A legal interference, adverse to the concerned party, drawn from silence or absence of requested evidence.
- The jury can infer that the evidence would have been adverse to (the defendant), and adopt the plaintiff's reasonable interpretation of what the document would have said.
- Examples include encrypted emails and incognito browsers.

Exclusionary Evidence

- There are some legal principles that prevent certain evidence being used in a court.
    - E.g. *Contamination*
        - The evidence may have come from another user, or even the investigator.
- An illegal and improperly done step in an investigation may void all following evidence.

Privacy

- The investigator may be restricted in searching due to the privacy laws.

Awareness of Ethical Responsibilities

Ethics

Rules you internalise and use to measure your performance.

- Standards that others apply to you
- Standards that your profession applies to you.
    - Also called rules of conduct

Relevant Organisation: *The International Society of Forensic Computer Examiners*

Purpose of Ethics

- Maintaining one's balance in a difficult situation
- Maintaining one's self respect
- Maintaining the respect of others
- Protecting oneself against legal challenge
- Identify and control one's bias when presenting evidence.

Ethical Principles

- To have nothing to hide
- Presenting unbiased evidence
- Complying with the rules of evidence

- Preserving confidentiality
- Disclosing all fees and charges
- Disclosing any conflict of interest.