

Forensics Cyber Security

Summary

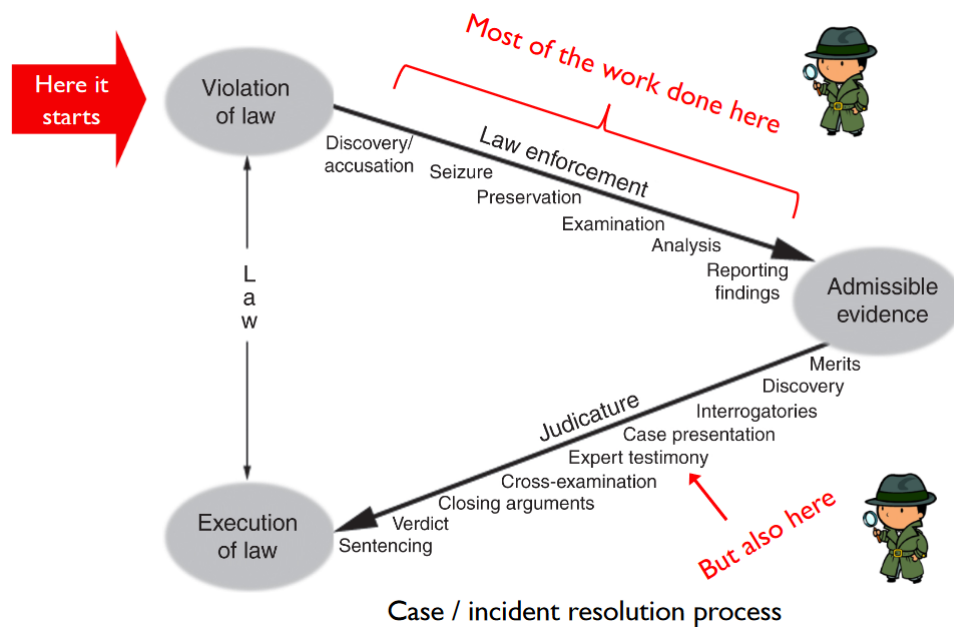
Contents

1	Investigation Process	3
1.1	Investigation Models	3
1.1.1	Kruse and Heiser (2001)	3
1.1.2	The Scientific Method	5
1.2	Acquiring Evidence	5
1.3	Analysing Data	6
1.3.1	Data Encoding	6
2	Recovering Hidden Artifacts	7
2.1	Steganography	7
2.1.1	Least Significant Bit	7
2.1.2	Steganalysis	7
2.2	Watermarking	8
2.2.1	LSB Embedding	8
2.2.2	Common Investigation	9
2.2.3	Steganography vs. Watermarking	10
2.3	Memory Analysis	10
2.3.1	Memory Acquisition	10
2.3.2	Virtual Addresses	11
2.3.3	Forensic Interpretation of Memory Dumps	11
2.3.4	Identifying Malicious Processes and Network connexions	11
2.4	Storage And Volume Analysis	11
2.4.1	Layout	12
2.4.2	HDD And SSD	12
2.4.3	Evidence From File Systems	12
2.4.4	File Carving	16
2.5	Evidence In Operating Systems	17
2.5.1	Location of Artifacts	17
2.6	Time Tracking	20
2.6.1	Time Tracking In File Systems	20
2.6.2	Timestomping	21
2.6.3	Digital Stratigraphy	21
2.7	Network Traffic Analysis	22
2.7.1	Packet Analysis Techniques	22
2.7.2	Flow Analysis Techniques	22
2.7.3	Protocol Analysis Techniques	23
2.8	Pitfalls in Network Addressing	23
2.8.1	MAC addresses	23

2.8.2	IP addresses	23
2.8.3	Ports	24
2.9	Network Scanning	24
2.9.1	Traceroute	24
2.9.2	Nmap	24
2.9.3	Other Techniques	24
2.9.4	Correlating Events Across the Network	25
3	Specialized Techniques and Tools	26
3.1	Wireless Network Forensics	26
3.1.1	Man-In-The-Middle Attacks	26
3.1.2	Other Typical Wi-Fi Attacks	26
3.1.3	Cellular Networks and Location Tracking	26
3.2	Email Forensics	27
3.2.1	Email Investigations	28
3.2.2	Antiforensics	28
3.3	Web Forensics	29
3.3.1	Code Injection Attacks	29
3.4	Deep Web And Anonymity	29
3.4.1	Google Hacking	29
3.4.2	Anonymity	29
3.4.3	Anonymity Networks	30
3.4.4	Hidden Services And The Dark Web	31
3.4.5	Investigation of Tor Traffic	32
3.5	Botnets	33
3.5.1	Infecting Botnet Nodes	34
3.5.2	Trail Obfuscation Techniques	34
3.5.3	Investigating Botnets	34
3.6	Rootkits	35
3.6.1	Rootkit Tools	35
3.7	Malware Analysis	35
3.7.1	Analysis Techniques	36
3.7.2	Anti-Analysis Techniques	36
3.7.3	Creating a Safe Environment	36
3.8	Cryptocurrency Investigations	36
3.8.1	Bitcoin	36
3.8.2	Bitcoin Investigations	37
3.9	Mobile Forensics	38
3.9.1	Evidence From Android Devices	38
3.9.2	Evidence Extraction	39
3.9.3	Android App Reverse Engineering	40
3.10	Cloud Forensics	40
3.10.1	Virtual Machines	40

Chapter 1

Investigation Process

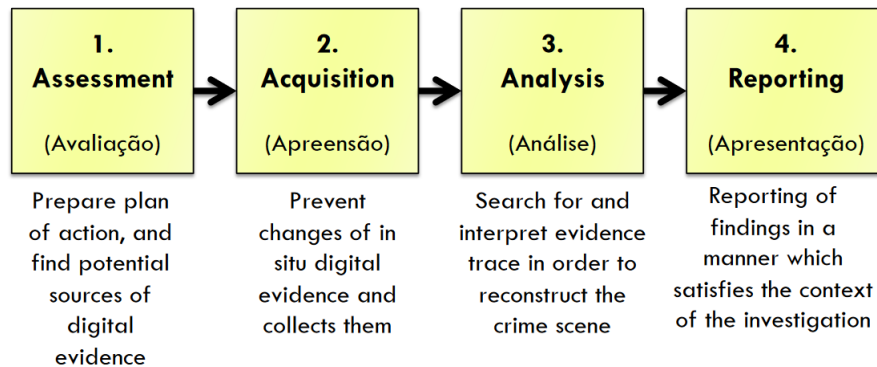


1.1 Investigation Models

An investigation model is a predefined pattern of activities when performing an investigation to generate admissible evidence. The four main guidelines for admissibility of digital evidence is that it must be relevant, authentic, credible and collected legally.

1.1.1 Kruse and Heiser (2001)

Is the *de facto* model used today



Assessment

This step consists of defining the scope and likely venue of the examination, the stakeholders, the likely sources of evidence, the forensic tools, personnel and legal documentation required.

The authorization level is set by the investigation type:

- Internal investigations are sponsored by an organization
- Civil investigations require the involvement of courts
- Criminal investigations involve the courts

For internal investigations, a signed letter of agreement outlining the scope of the investigation along with contractual details is needed. For civil and criminal investigations, a court order is needed prior to starting.

Acquisition

This step consists of acquiring and preventing changes to possible evidence. Some highlights include:

- Maintaining a chain of custody
- Generating integrity checks

Analysis

In this step, the artifacts obtained in the last step are analysed in order to extract all material evidence. It is common practice to reconstruct events (temporal, relational and functional).

Reporting

The final step consists of generating the work product of the analysis, documentation. The 5 levels of documentation are:

- General case documentation

- Procedural documentation
- Process documentation
- Case timeline
- Evidence chain of custody

1.1.2 The Scientific Method

In practice, digital investigators need to complement investigative models with a methodology that guides them in the right direction, allows them to maintain the flexibility to handle diverse situations and preserves the rigors of forensic science.

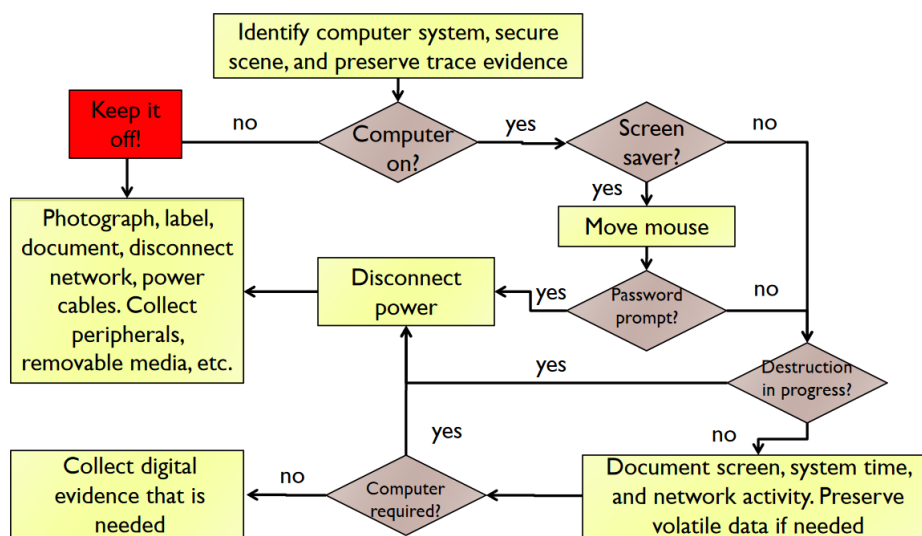
- Observation
- Hypothesis
- Testing
- Conclusion

1.2 Acquiring Evidence

When acquiring digital evidence, the main concern is to preserve it in a way that it maintains an accurate representation of the original data and that it is as complete as possible.

The general procedure when finding storage devices is to collect them or, if they can't be taken, perform data extraction on the spot. When extracting data, one must be careful as to not perform any writes, usually by using a software or hardware write blocker.

Sometimes it will not be possible to obtain an exact physical copy, bit by bit, of the entire source device, due to accessibility, space or time constraints. In this case, one can try to perform a logical acquisition of files (selecting and copying only relevant files) or volumes (copying the contents of an encrypted volume that is being used, unencrypted, on a powered computer)



Mobile Devices

Mobile devices are unique since persistent memory can't be removed and requires the original OS to be read. In this case, one should bring the device, along with power and data cables.

If the device is powered on, it should be isolated as soon as possible, by activating airplane mode, removing sim cards or placing it inside a Faraday bag.

1.3 Analysing Data

1.3.1 Data Encoding

ASCII is the common code text representation, however, there are some ASCII extensions made to support more characters, such as UTF-8, UTF-16 or UTF-32.

Sometimes it may also be necessary to encode binary objects into text. Base64 is a popular encoding scheme.

Decoding Unknown Files

File types have defined magic numbers, constant numerical or text value used to identify a file format or protocol. In cases where the file is corrupted certain bytes may have to be altered in order to repair it.

Chapter 2

Recovering Hidden Artifacts

2.1 Steganography

Steganography is the art and science of communicating in a way that hides the existence of a message. Digital steganography works by encoding secret bits in files, such as photos or audio files, with secret data.

2.1.1 Least Significant Bit

Using this method, only the LSB of a pixel is used to encode information. More bits can be used in order to hide more information, but using too many can visually corrupt the image. Encoding on a single higher bit can be used to create a watermark.

When using LSB, one can use consecutive pixels or take advantage of a (fixed or random) interval.

2.1.2 Steganalysis

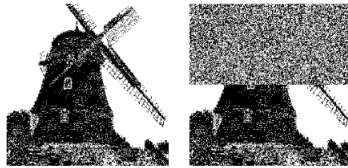
Steganalysis is the art and science of detecting hidden data. It's methods rely on the inspection of file properties or contents and the detection of signature patterns. Some approaches consist of:

- Visual attacks

Sometimes, a simple observation shows
grainy noise introduced into the steganogram



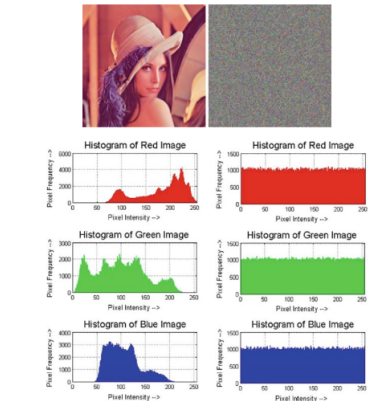
Windmill as carrier medium (left), and steganogram generated using EzStego (right)



Blue plane 0: left, nothing embedded; right, continuous embedded payload in 50% of the carrier (EzStego)

- Statistical attacks

Illustration of R/G/B histograms for two images: a normal photo (left), a randomly synthesized image with max entropy signal, causing flat frequency distributions (right)



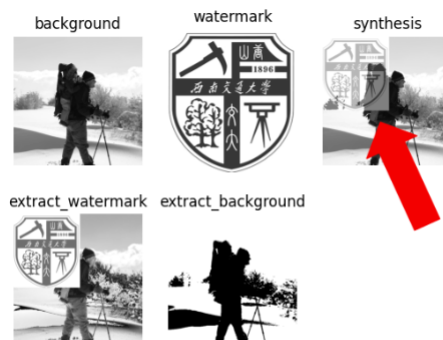
- Machine Learning

2.2 Watermarking

Watermarking is about establishing identity/ownership of digital content to prevent unauthorized use.

2.2.1 LSB Embedding

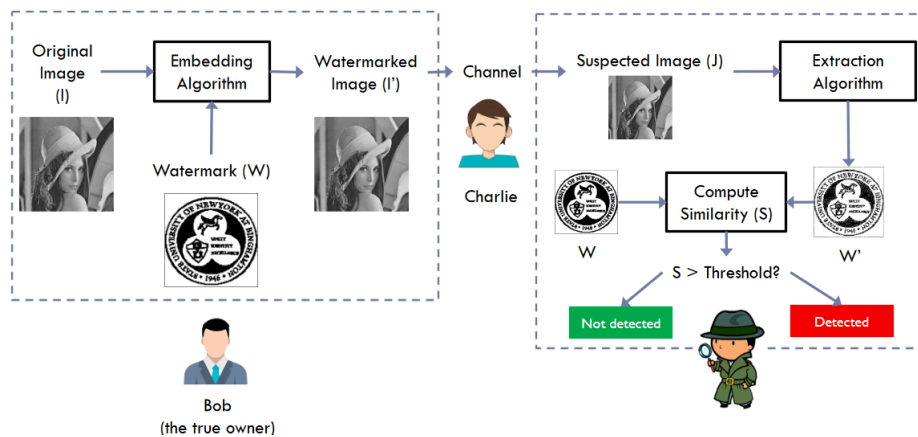
Watermarks can be applied using LSB, as mentioned before, or a higher bit in order to make it visible.



Interestingly, by embedding the watermark in a higher LSB bit (LSB 7 bit), we can make it visible. This is one way to generate visible watermarks

2.2.2 Common Investigation

Watermarks are frequently used in proprietary content that should not be altered. If such a file is tampered with, even if the changes are not directly visible, they will likely reflect on the hidden watermark.



2.2.3 Steganography vs. Watermarking

- ▶ Both techniques **hide** a message ***m*** in some cover data ***d***, to obtain ***d'***, practically indistinguishable from ***d***
- ▶ However, they have different goals:

Steganography

- ▶ An eavesdropper must not be able to **detect** the presence of ***m*** in ***d'***
- ▶ Primarily for 1-to-1 communication
- ▶ Robustness not typically an issue
- ▶ Capacity desired for message is large
- ▶ Always invisible
- ▶ Typically dependent on file format

Watermarking

- ▶ An eavesdropper cannot remove or replace ***m*** in ***d'***
- ▶ Primarily for 1-to-many communication
- ▶ Robustness of watermark is a main issue
- ▶ Known watermark may be there
- ▶ Can be visible or invisible
- ▶ Watermark can be considered to be an extended data attribute

2.3 Memory Analysis

Physical memory dumps may be analysed in order to recover relevant data.

2.3.1 Memory Acquisition

There are several software and hardware memory acquisition techniques. The two main factors can be used to help make a decision are:

- Atomicity: how close to the present memory state can the forensic memory snapshot be retrieved
- Availability: whether the tools necessary to perform memory acquisition are available or not

Some software-based approaches include crash dumps, the hibernation file, operating system injection and virtual machine imaging. As for hardware-based approaches, there are hardware cards that can obtain forensic image of a computer's RAM and special buses that can read volatile memory. It is also important to note the difference between warm and cold boots, since RAM retains memory during reboots as long as power is provided:

- Warm boots refer to reboot methods in which power is never removed from the memory module (e.g., press reset button)
- Cold boot refers to reboot methods in which power is removed from the memory module (e.g., pull the plug and reboot)

2.3.2 Virtual Addresses

Virtual addresses may be converted into physical addresses, like the following example:

Goal: Find the corresponding physical address to inspect data in close spatial proximity

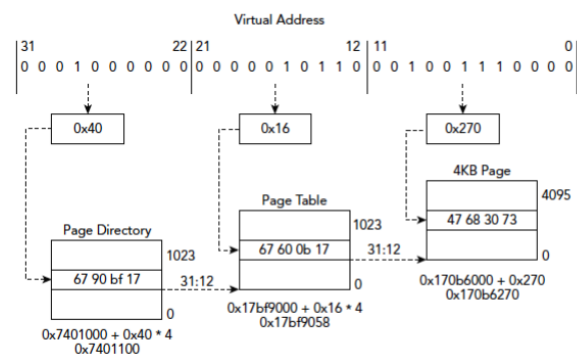
Convert the VA to binary and decompose it into the relevant offsets (table, left)

Calculate the PA of the PDE by multiplying the page directory index by the size of the entry (4 bytes) and then adding the page directory base, 0x7401000

Read the value from physical memory stored at the PDE address, i.e., value is 0x17bf9067 (in little endian format!); from bits 31:12 of the PDE learn the PA for the base of the PT and add the offset to calculate PTE's PA (0x17bf9058)

Read the value from that address, select bits 31:12 to find the PA for the page, and compute the offset using the bits 11:0 from the VA, resulting in the PA: 0x170b6270

Paging Structure	VA Bits	Binary	Hex
Page directory index	Bits 31:22	0001000000	0x40
Page table index	Bits 21:12	0000010110	0x16
Address offset	Bits 11:0	001001110000	0x270



2.3.3 Forensic Interpretation of Memory Dumps

Once a memory dump has been performed, it is necessary to interpret the data structures in the raw memory sample. Some methods used include:

- Tree/list traversal: find index into lists/trees of interest and follow them through pointer dereferencing to reconstruct the full data structure
- Fingerprint/pattern search: search for relevant patterns in memory

2.3.4 Identifying Malicious Processes and Network connexions

Upon capturing a RAM dump, it is possible to list all the active processes and network connexions, making it possible to indentify and extract malicious code.

2.4 Storage And Volume Analysis

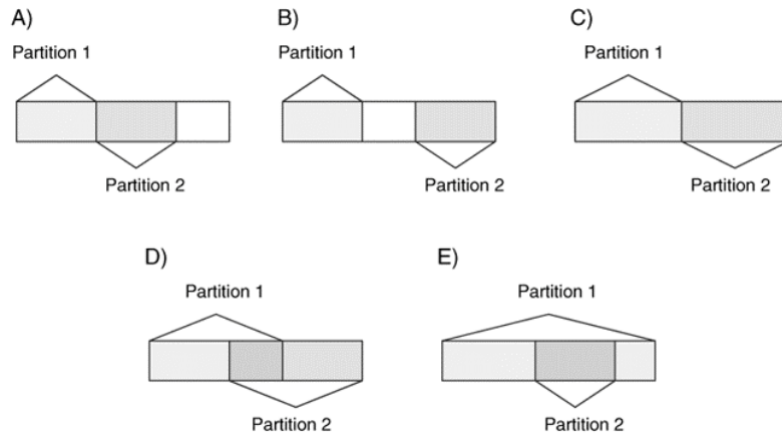
A volume is a collection of addressable sectors that an OS or application can use for data storage. A partition is a fraction of consecutive sectors in a volume. By definition, a partition is also a volume.

2.4.1 Layout

To identify the volume layout, the partition tables must be analyzed, however if the partition system becomes corrupt or erased, automated tools may not work.

Consistency

It's necessary to check each partition relative to the other partitions.



2.4.2 HDD And SSD

There are some considerations of note when working with different drive types. HDDs can have passwords, be self-encrypting or self-wiping. It is possible to hide data in the Host Protected Area (HPA), added in ATA-4 spec. There are also two types of formatting for these drives:

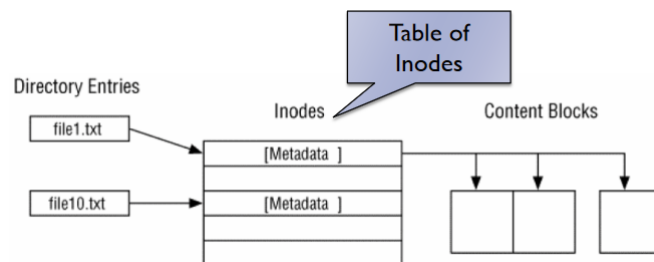
- Low-level formatting: Physically defines tracks and sectors on disk and erases data
- High-level formatting: Performed when initializing a file system on a partition, does not destroy data, only FS metadata

SSDs are not as well understood as HDDs, the physical location of data is hard to predict, deleted data is hard to recover and it is difficult to read data off chips directly.

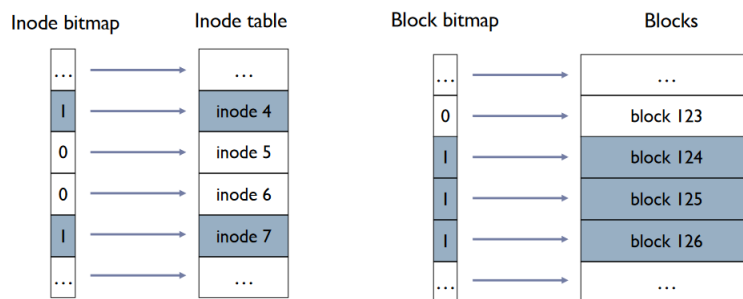
2.4.3 Evidence From File Systems

Each partition of a disk can have its own file system. In many Linux distributions, ExtX is the default file system. In ExtX:

- File contents are stored inside blocks
- The blocks allocated to a file are kept by a record called inode
- Directory entries associate the file name with the file's inode



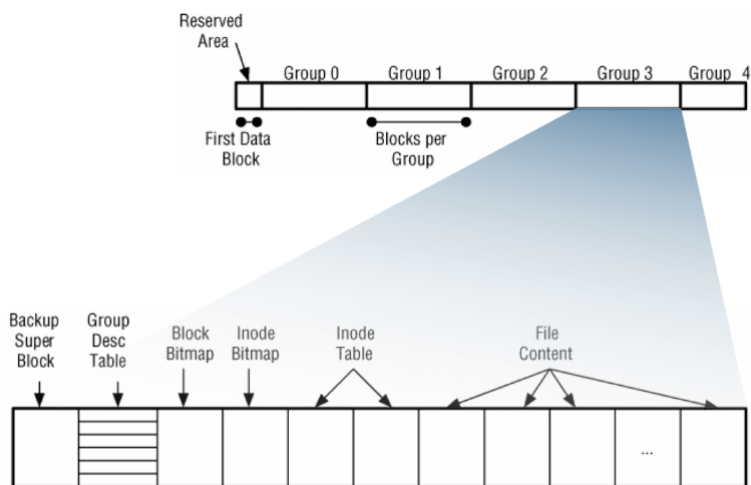
Each inode is the root of an unbalanced tree of blocks that belong to a given file. To keep track of inode and block allocation bitmaps (bit arrays) are used. The inode bitmap tells which inodes are allocated to files and the block bitmap tells which data blocks are allocated to files.



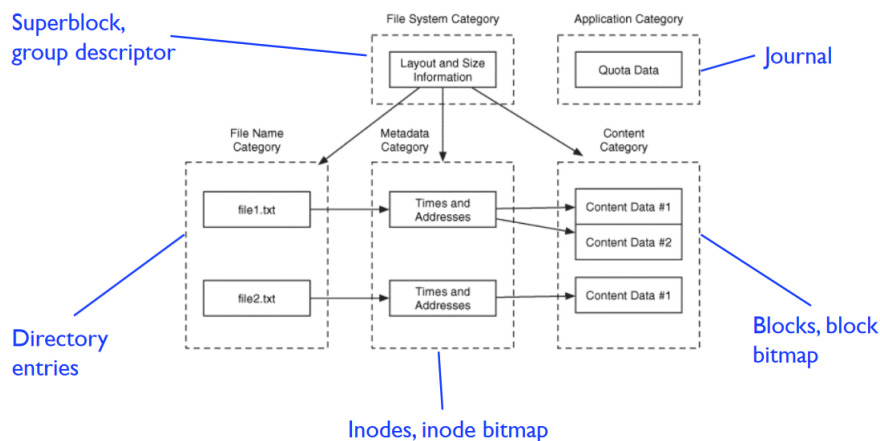
In order to create a new file it's necessary to allocate a new inode, the required blocks and an entry in the directory, as well as update data blocks, inode, and directory entry. When deleting a file, the inode and block bitmaps are updated and the directory entry unallocated, most contents of inode, data blocks, and directory entry remain intact. Deleting a file involves updating the record length of previous file entry in the directory.

In Ext3, the recent history of updates to the system is maintained in a journal. Journals are implemented as a circular buffer and are append-only. If the computer crashes while journaling, the file system may enter an inconsistent state.

The ExtX file system is organized as sequence of logical blocks in disk (block size is defined upon disk formatting). Blocks are grouped into larger units called block groups and all block groups have equal length possibly except the last one. The first data block aka boot block is not used by the FS, it has a fixed 1024 byte length and may contain bootstrap code.



Data evidence categories of the ExtX file system family:



File System Analysis Techniques

In the content category:

- Data unit viewing
- Logical file system-level searching
- Unallocated data unit searching
- Consistency analysis
- Data carving
- Anti-forensic techniques (e.g., data wiping)

In the metadata category:

- Metadata lookup

- Logical file viewing
- Logical file searching
- Unallocated metadata analysis
- Metadata attribute searching and sorting
- Consistency checking

In the file name category:

- File name listing
- File name searching
- Consistency checking

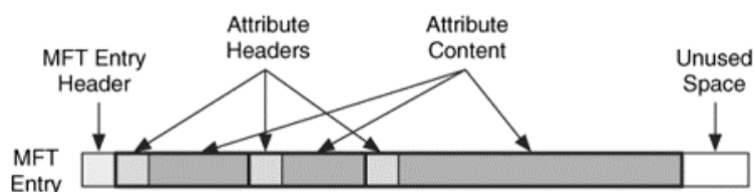
NTFS

In NTFS, files are chunked into clusters (akin to ExtX's blocks), with the central paradigm that everything is a file.

NTFS Boot Sector	Master File Table	File System Data	Master File Table Copy
------------------------	-------------------------	------------------------	---------------------------------

Partition formatted to an NTFS file system

In NTFS, information about all files and directories is contained in the Master File Table (MFT). Every file and directory has at least one entry in MFT (1kb size).



MFT entries are a precious source of forensic data and some small files can fit entirely within the MFT entry, bigger files require allocation of extents. NTFS stores metadata across several metadata files. Steps for file creation:

- Find an unallocated MTF entry for the new file (304)
- Initialize the MFT entry 304 with basic attributes
- Allocate two clusters in the \$BITMAP file (692, 693)
- To add a file name, first look up the root directory

- Then, locate the dir1 index in the directory's B-tree
- Follow the entry address 200, and add index entry for file1.txt
- In previous steps, add entries to journal

And to delete a file:

- Find the dir1 directory by processing the MFT entry 5, the root directory
- Search for the file1.dat entry, whose address is 304
- Remove the entry from the index of dir1
- Unallocate MFT entry 304 by cleaning the in-use flag
- Clusters of entry 304 unallocated in the \$Bitmap file
- In previous steps, add entries to journal

In NTFS, every file has a \$DATA attribute, which contains the file content. However, a file can have more than one \$DATA attribute, named Alternate Data Stream (ADS). ADSs are great for hiding files inside files, without creating a new MFT entry.

2.4.4 File Carving

File carving is a powerful technique because it can identify and recover files of interest from raw, deleted or damaged file systems, memory, or swap space data. There are different techniques, such as:

- Structure-based carving (e.g. magic numbers)
- Content-based carving

Parallel Unique Path (PUP)

The key insight behind the PUP algorithm is to grow all files simultaneously and append best match at each step.

- Initially, assume that all file clusters are randomized and identifies headers using keywords/signatures
- For each header, find best match (using matching metric)
- Find best match for recently added node
- Repeat until all files are built or no more nodes can be chosen

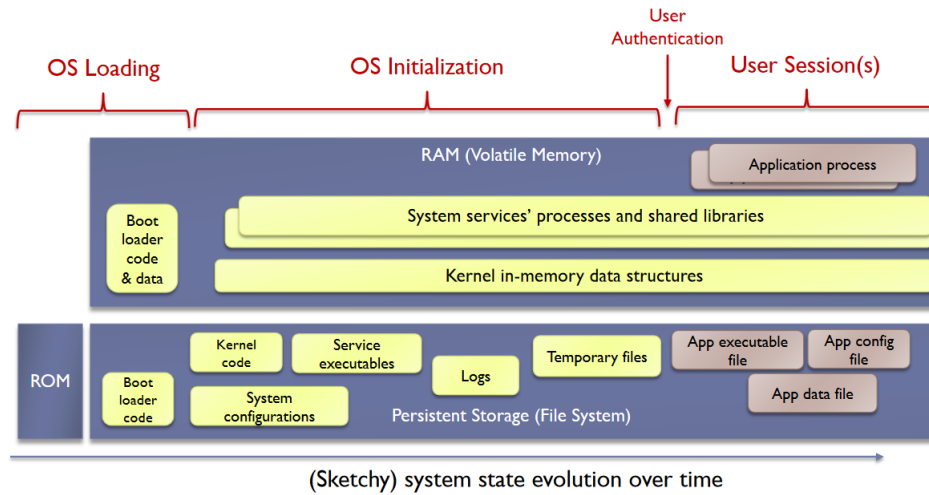
Bifragment Gap Carving (BGC)

BGC leverages an observation that bifragmentation (two fragments only) is the most common fragmentation type. Its goal is to match both fragments of each file.

It uses magic number to locate the header and footer and, to verify if they are properly sequenced, checks if the file obeys the structured rules of its file type.

2.5 Evidence In Operating Systems

OS artifacts are evidentiary data pertaining to data/code maintained/executed by the OS. Such artifacts can help track past/live user activity



Some typical user activities to investigate include:

- File download
- Program execution
- File opening/creation
- Deleted file or file knowledge
- Physical device location
- USB or drive usage
- Account usage
- Network usage

2.5.1 Location of Artifacts

The location of artifacts is OS-dependent.

► Linux

► Configuration files

- /etc/passwd
- /etc/sudoers
- /etc/inittab
- /var/spool/cron/*
- ...

► Log files

- /var/log/boot.log
- /var/log/auth.log
- ...

► Windows

► Dedicated data structures

- Windows Registry

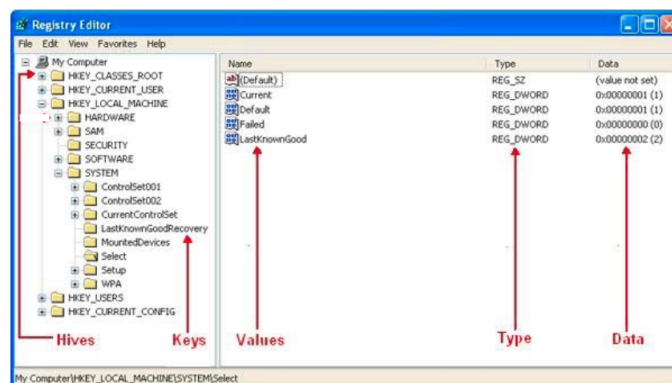
► System logging service

- Windows Event Log

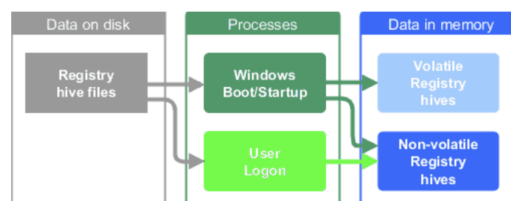
► Special files

Windows Registry

Virtually everything done in Windows refers to or is recorded into the registry. The Registry can be seen as a unified file system, its specific structure is divided into keys and values. Main root keys (named hives) represent the root directory, sub-keys represent the sub folders, and values represent the files.

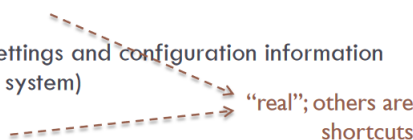


The registry is commonly described as a hierarchical database, however, its database is only ever complete when loaded into your computer's memory, and it is the sum of two parts, the data and the processes that create it and provide access to it.



Each root key shown in the registry editor is actually a file in the filesystem called registry hive. A hive contains a logical group of keys, subkeys, and values in the registry that has a set of supporting files containing backups of its data.

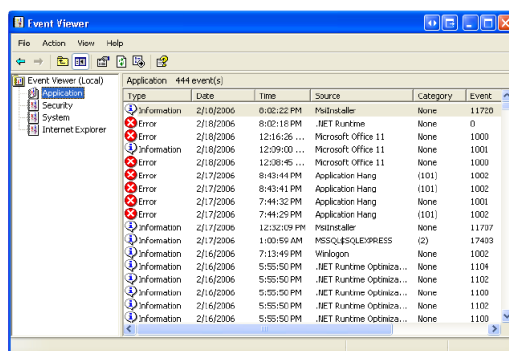
- ▶ **HKEY_LOCAL_MACHINE (HKLM)**
 - ▶ Contains system-wide hardware settings and configuration information (e.g., list of drives mounted on the system)
- ▶ **HKEY_USERS (HKU)**
 - ▶ Contains the root of all user profiles that exist on the system
- ▶ **HKEY_CLASSES_ROOT (HKCR)**
 - ▶ Ensures the correct program opens when executed in Windows Explorer
- ▶ **HKEY_CURRENT_USER (HKCU)**
 - ▶ Contains the profile (settings) of the user who is currently logged in
- ▶ **HKEY_CURRENT_CONFIG (HCU)**
 - ▶ Information about the HW profile used by the computer during start up



Some registry hives are stored on disk even when Windows is not running. All the registry hive structures only exist in memory. This includes a set of volatile hives that only exist when Windows is running. Access is done through the Registry Configuration Manager.

Windows Event Log

Whenever an event, such as a user logging on or off, occurs, the operating system logs the event. Windows has a centralized log service to allow apps and OS to report events that have taken place.



Some other interesting artifacts include:

- MRU lists: MRU ('most recently used') lists contain entries about specific actions done by the user
- OpenSave MRU: tracks files that were opened / saved within a Windows shell dialog box

- Last Visited MRU: Tracks the specific executable used by an application to open the files documented in the OpenSaveMRU key
- UserAssist: contains information about the exe files and links that you open frequently
- Shell Bags: track user window viewing preferences to Windows Explorer
- Prefetch files: a prefetch file is created when an application is run from a particular location for the very first time; used to speed up the loading of applications
- Thumbnails: (on Win XP) hidden file thumbs.db in directory where pictures exist; stores thumbnail even if pictures deleted
- Autorun locations: registry keys that launch programs or apps during boot

2.6 Time Tracking

Computers have several different methods of timekeeping:

- Real Time Clock (RTC)
- System clock
- Network Time Protocol (NTP)
- Network Identity and Time Zone (NITZ)
- Global Positioning System (GPS)

Timestamps may not always be reliable, due to bugs, synchronization issues, bad configurations or interpretation, etc.

2.6.1 Time Tracking In File Systems

MACtimes are three time attributes attached to any file or directory in UNIX, Windows, and other systems:

- atime: last time the file or directory was accessed
- mtime: changes when a file's contents are modified
- ctime: keeps track of when the contents or meta-data about the file has changed: owner, group, file permissions, etc.

Sometimes this information is enriched with creation time.

2.6.2 Timestomping

Timestamps are not meant to be manipulated by the end user, however, a resourceful user (and often, malware) can modify these timestamps using various methods: this is called timestomping. Some strategies for timestomping detection include:

- Anomalies in timestamp format
- Inconsistencies between MFT attributes
- Inconsistencies with other timing sources

2.6.3 Digital Stratigraphy

Digital stratigraphy is a new sub-field emerging in digital forensics that studies file system traces and writing patterns to infer time-related facts. This is usually:

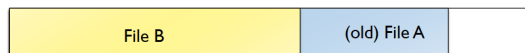
- Based on how data was overwritten:

- ▶ Suppose file A was first created, fits into one block



- ▶ Then, A was deleted and that block was unallocated

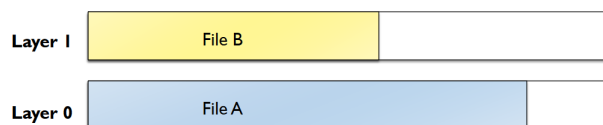
- ▶ Later file B was created, the OS reallocated the same block, but B is a smaller file than A; this is what the block will look like



- ▶ This is what the forensic analyst will be able to retrieve



- ▶ We can conceptualize this into two different strata:



- ▶ So, even if we do not have timestamp info about B, but we have about A, then we can still say that B is likely more recent than A

- Based on data positioning:

- ▶ Suppose now that A uses 3 blocks when created

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

- ▶ Now, file B is created in block #1, and it keeps increasing until it requires 5 blocks:

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

- ▶ The OS will typically try to find contiguous blocks but in this case the above will occur: fragmentation

- ▶ This is what the forensic analyst will be able to retrieve

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

- ▶ We can conceptualize this into two different strata:

Layer 1	1	2	3	4	5	6	7	8	9	10
Layer 0	1	2	3	4	5	6	7	8	9	10

- ▶ So, even if we do not have timestamp info about neither of files, we may likely infer that B is likely more recent than A

2.7 Network Traffic Analysis

The analysis of network traffic allows us to collect information from network traces and detect potential attacks and inspect the contents of communications.

2.7.1 Packet Analysis Techniques

Packet traces are usually collected through packet sniffers. Some of the main packet analysis techniques include:

- Parsing protocol fields
- Packet filtering
- Pattern matching

2.7.2 Flow Analysis Techniques

A flow is a sequence of packets sent from a particular source to a particular unicast, anycast, or multicast destination that the source desires to label as a flow. Flow analysis consists in examination of sequences of related packets. The usual techniques include:

- List flows
- Export a flow
- File and data carving

NetFlow

NetFlow is a technology built into supported devices that collects and categorizes IP traffic, but does not collect the entire payload of the packets. It logs flows, not TCP connections, and for each TCP connection two flows are recorded (since flows are uni-directional).

2.7.3 Protocol Analysis Techniques

Protocol analysis aims to understand how a particular communications protocol works, what it's used for, how to identify it, how to dissect it. Many protocols are standardized, but many others are deliberately kept secret. Some techniques include:

- Search for common binary/hexadecimal/ASCII values that are typically associated with a specific protocol
- Leverage information in the encapsulating protocol
- Leverage the TCP/UDP port number, many of which are associated with standard default services
- Analyze the function of the source or destination server (specified by IP address or hostname)
- Test for the presence of recognizable protocol structures

2.8 Pitfalls in Network Addressing

Network addressing includes MAC and IP addresses, as well as port numbers, which may not always be reliable.

2.8.1 MAC addresses

A MAC address is a unique identifier assigned to network interfaces for communications at the data link layer of a network segment. It can be used to track traffic sources within a network, but it can be modified, so one must be careful about using it as a unique identifier.

2.8.2 IP addresses

IP addresses are scope-defined, and thus, private IP addresses can't be routed through the internet, and are only meaningful on its private network.

One of the main sources of unreliability of IP addresses is the risk of spoofing of the source IP, which might be unreliable in unidirectional communication.

IP addresses may also be masked by middleboxes, such as proxies, VPN servers, NAT, load balancers or firewalls. Some protocols, such as IPsec, also implement tunneling to mask the IP until reaching its final destination.

2.8.3 Ports

Certain protocols have defined ports for communication. Internet Assigned Numbers Authority (IANA) is responsible for IP addresses and port number assignments for well-known ports.

A technique called port knocking aims at hiding ports from view. To gain access to the service you want, you need to go through what amounts to a network authentication. A method of externally opening ports on a firewall is by generating a connection attempt on a set of prespecified closed ports

2.9 Network Scanning

Network scanning is concerned with determining the topology of a given network.

2.9.1 Traceroute

Traceroute is a very useful tool for network scanning. It works by making use of the time to live (TTL) IP header field. It works by:

- Sending a message out to a destination with increasing TTL values
- The first packet being sent has a TTL of 1
- When the very first router (the default gateway) receives the message, it decrements the TTL to 0 and responds with the ICMP error message
- Once the sending system receives the message, it has the IP of the first router

2.9.2 Nmap

Nmap is useful in order to get open TCP ports. It works by performing a SYN scan. A system that has an open port will respond with the SYN/ACK or with a RST message. Nmap responds with a RST after the SYN/ACK.

Nmap can also perform other scan types, such as connect, FIN and UDP scans.

2.9.3 Other Techniques

Banner Grabbing

It's the process of connecting to a service and probing it to get the service banner. Different protocols support different ways of conveying information about themselves. One can use the netcat (nc) tool for banner grabbing.

Ping Sweeps

A ping sweep is a way of identifying all of the hosts that are discoverable on a network. We can use what nmap calls host discovery.

Vulnerability Scanners

A vulnerability scanner works with a database of signatures, and this database needs to be updated regularly.

Network Intrusion Detection Systems

Such systems may provide alerts or logs that include details regarding illicit connexions, and can be configured to begin detecting events it wasn't previously recording.

2.9.4 Correlating Events Across the Network

Individual pieces of digital data might not be useful on their own, but patterns may emerge when combined. To develop a clearer picture of the crime, three forms of reconstruction should be performed when analyzing evidence:

- Relational (who, what, where)
- Temporal (when)
- Functional (how)

Chapter 3

Specialized Techniques and Tools

3.1 Wireless Network Forensics

Wireless networks are important in forensic analysis since, through them, it's possible to identify and recover devices, assets or individuals.

3.1.1 Man-In-The-Middle Attacks

Free wi-fi networks are susceptible to this type of attack.

An attacker might set up a hotspot spoofing the wi-fi network using the same SSID and password and have victims mistakenly connect to it. This is then used to listen to their traffic and possibly steal sensitive information.

3.1.2 Other Typical Wi-Fi Attacks

Some other common wi-fi attacks include:

- Decryption attacks: WEP is easily broken; WPA2 can be broken under weak passwords
- Wardriving: Attackers drive around a neighborhood and use a laptop with a GPS device, antenna to identify and record the location of unprotected wireless networks
- Packet sniffing: Use packet sniffers to intercept traffic on unencrypted WiFi networks or on breakable WEP/WEP2 networks
- Rogue access points: Unauthorized wireless APs that extend the local network bypassing network-enforced security measures

3.1.3 Cellular Networks and Location Tracking

Cellular networks have the same types of forensic data as other TCP/IP networks, along with SMS and telephone calls.

Identifiers

In this type of network, two new device identifiers are present:

- IMEI: a unique number associated with a particular device (searchable on a database)
- IMSI: an identifier of a SIM card. Allows to track all activities relating to a particular subscriber.

Tracking Location

There are several positioning techniques:

- Proximity
- Trilateration, Triangulation
- Fingerprinting
- Hybrid
- GPS

	Cellular Networks	GPS	Wi-Fi Positioning	Bluetooth LE Positioning
Environment	Outdoor	Outdoor	Indoor & outdoor	Indoor mainly
Accuracy	1.2 km	10 – 100 m	5 – 10 m	1 – 2 m
Availability	Widespread in most civilized regions but does not offer full global coverage; allows tracking GSM/GPRS devices (every single mobile phone out there); location information accessible from the network provider	Truly ubiquitous, covering the entire planet; the tracked devices equipped with GPS receiver hardware; GPS information can be automatically transmitted to the broadband mobile provider	Widely available infrastructure, including residential, businesses, and commercial spaces	Increasingly gaining traction, e.g., for staff and asset monitoring in warehouse operations, transportation, healthcare facilities, sports industry, etc.

Attacks

Some specific cyberattacks involving cellular networks include GPS spoofing, smishing and SIM swapping attacks.

3.2 Email Forensics

Ease, speed and relative anonymity of email makes it lucrative option for committing crimes for the criminals. Some common email attacks include:

- Email spamming: can be defined as sending unsolicited emails
- Mail Bombing: the primary intention of mail bombing is to cause denial-of-service attack to the victim. It is achieved by sending huge volumes of emails to the victim's mailbox/server to crash down

- Phishing: defined act of sending an unsolicited and illegitimate email falsely claiming to be from legitimate site/company to win the victim's trust and acquire their personal/account information
- Email spoofing: the act of forging the email header so that the message appears to originate from source other than the actual source

3.2.1 Email Investigations

In email investigations, one looks for evidence of email abuse/incriminating content. The way such investigations are conducted is dependent on the protocols used:

Post Office Service	Protocol	Characteristics
Stores only incoming messages	POP	Investigation must be at the workstation.
Stores all messages	IMAP MS' MAPI Lotus Notes	Copies of incoming and outgoing messages might be stored on the workstation or on the server or on both.
Web-based send and receive	HTTP	Incoming and outgoing messages are stored on the server, but there might be archived or copied messages on the workstation

Some email-related sources of evidence include the header (email itself) and the logs left behind. The header includes the sender, the network path it traversed, timestamp details, encoding information, etc.

Some general advice is to verify IP addresses, look for breaks/discrepancies in the "Received" lines and make a timeline of events.

3.2.2 Antiforensics

Open Relays

Open relays are SMTP server configured in such a way that it allows anyone on the Internet to send e-mail through it, not just mail destined to or originating from known users.

Spoofers use open relays to attempt to hide the person and IP of the system that sent the email. In these cases, the header and logs will contain the originating address.

False Received From Header

Such a header leads the investigator to the wrong server by adding a seemingly valid received from header. To avoid detection, the spoofer's real address will be recorded somewhere in the Received from headers, but the investigator will

not know which one.

In these cases, email received from headers will contain the actual IP address of the originating system, you just won't know which header is correct.

3.3 Web Forensics

An attack scenario describes the ways an attacker might exploit the vulnerabilities of a Web app. These apps are often distributed and business critical, which prove to be challenges for web investigations.

3.3.1 Code Injection Attacks

These attacks are carried out via entering malicious code into the input control of web form or address bar of web browser. Some common types are:

- Cross Site Scripting (XSS): XSS attacks allows an attacker to run arbitrary JavaScript in the context of a vulnerable website. The goal is to steal the client cookies or other sensitive info which can identify the client with the web site
- SQL injection: an attacker injects malicious text string, most often a database query, into an available web form that is eventually executed by the database
- PHP injection: allows an attacker to supply code to the server side scripting engine. This vulnerability allows an attacker to run arbitrary, system level code on the vulnerable server and retrieve any desired information contained therein.

3.4 Deep Web And Anonymity

The surface web is that portion of the World Wide Web that is readily available to the general public and searchable with standard web search engines.

The deep web is not necessarily bad, it's just that the content is not directly indexed. The part of the deep web where criminal activity is carried out is named the dark web.

3.4.1 Google Hacking

Google provides keywords for advanced searching, which can be used to search for file types, finding servers or even unprotected webcams.

3.4.2 Anonymity

It's hard to be anonymous on the internet:

- IP addresses can be linked directly to individuals
- Browsers can be tracked

- Activities can identify people
- Internet access points can be wiretapped

Anonymity Systems

The aim is to conceal the identity of communicating parties. There are several types of anonymity, all with the desired properties of unlinkability and unobservability:

- Sender anonymity
- Receiver anonymity
- Sender-receiver anonymity

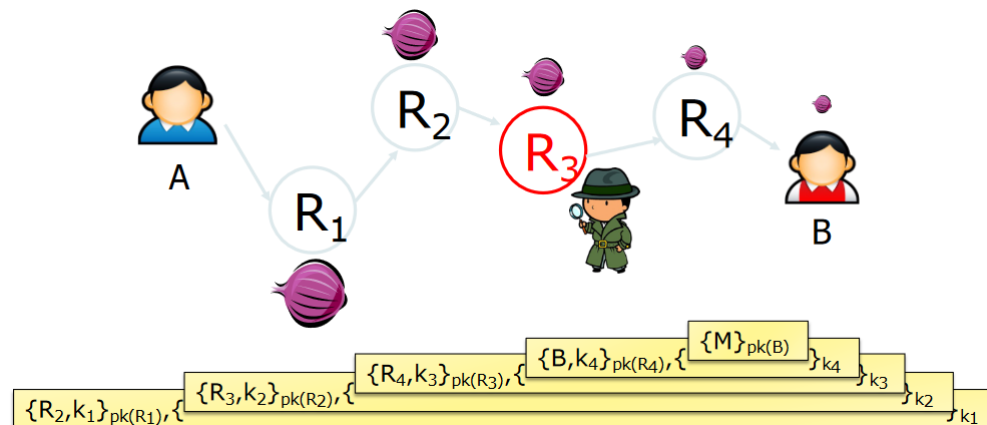
Anonymity can be quantified through anonymity sets, the number of suspects. The larger the set, the stronger the anonymity.

3.4.3 Anonymity Networks

The aim is to overcome limitations of public proxy servers. An anonymity network forwards traffic through a chain of network nodes: relays.

Onion Routing

Onion routing uses public-key cryptography to establish a "circuit" with pairwise symmetric keys between hops on the circuit. The sender chooses a sequence of routers and then symmetric cryptography is used to move data.



Routing info for each link encrypted with router's public key. Each router learns only the identity of the next router.

Tor

Tor is the 2nd generation onion router, a volunteer-based low-latency anonymity network. When Bob wants to connect to Alice through Tor, he downloads the Consensus (list of Tor relays) from the Tor authorities. Each Tor relay

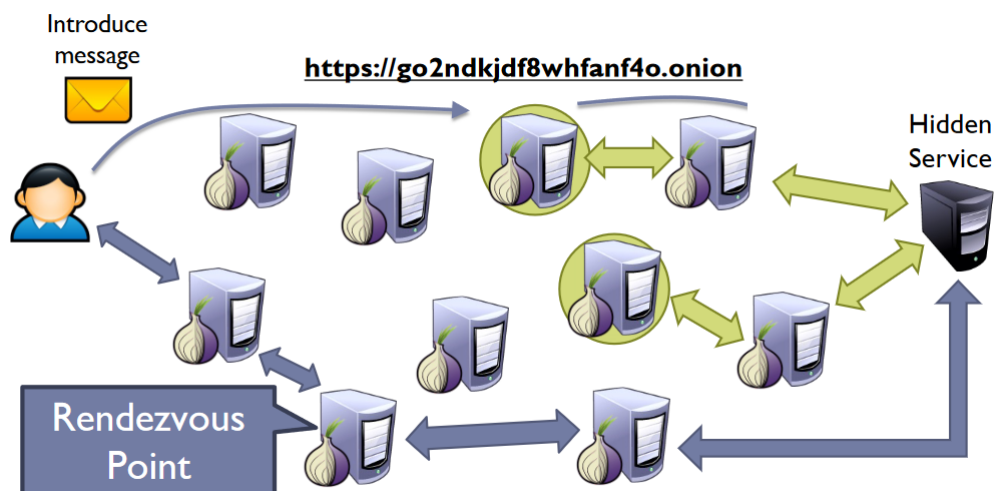
is uniquely identified by an RSA public key. Then three "random" relays are chosen, and the following circuit establishment happens:

- Connecting to first relay:
 - Bob sets up a TLS connection with the Entry node
 - Negotiates a Diffie-Hellman (DH) key with first node in path
- Connecting to second relay:
 - Then complete a DiffieHellman key exchange handshake with the second node by using the first node as a proxy
- Connecting to third relay:
 - Repeat the same procedure with the third node but use the chain of entry and middle nodes as proxies
- Data transmission:
 - To connect to a server, the client packs the messages in fixed 512 byte cells, encrypted with the three relay keys.
 - Each relay strips off one layer of encryption, breaking the linkability between the sender and the destination
 - Normally, messages are also encrypted using end-to-end protocol

3.4.4 Hidden Services And The Dark Web

Tor supports hidden services: allows for running a server and have people connect without disclosing the IP or DNS name. This is done through a rendezvous point: a meeting place for the user and service.

Each hidden service has introduction points (yellow), that can be found through Onion URLs, since they are hashes. Before sending an introduction message, the client selects a rendezvous point, and then informs the service:

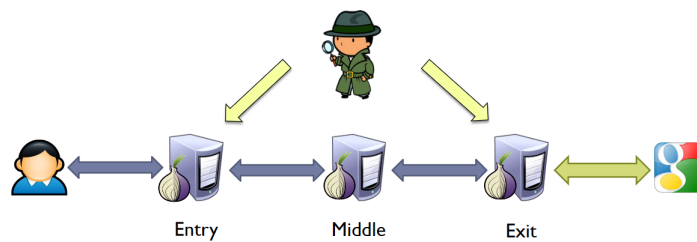


3.4.5 Investigation of Tor Traffic

Onion Routing gets its security from the fact that it is difficult for an adversary to position itself on networks such that it is able to view all the nodes in the route, but if someone is able to learn the entire path, it loses its security. By probing the entry and exit relays, we can correlate the traffic (e.g., by volume or timing).

Predecessor Attack

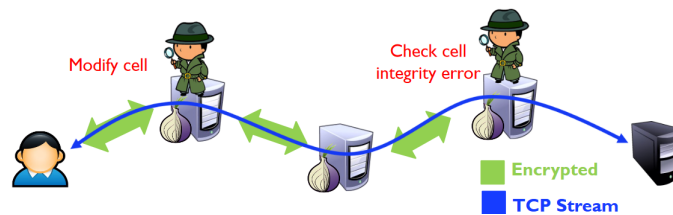
The investigator controls two relays of the Tor network. Probability of being in the right positions (entry and exit) increases over time.



Guard relays help prevent attackers from becoming the first relay. If a user chooses an evil guard, there is a M/N chance of full compromise, where N is the total number of nodes and M is the number of nodes controlled by the attacker.

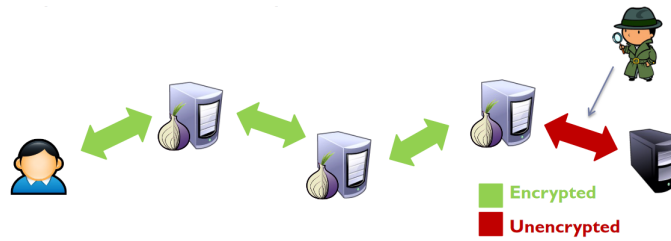
Protocol Level Attacks

Can confirm anonymous communication relationships quickly and accurately by manipulating one single cell. Detection is possible because Tor uses the counter mode AES (AES-CTR) for encrypting cells.



Eavesdropping On The Exit Node

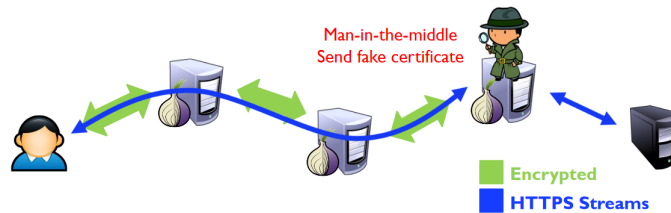
Tor does not provide end-to-end encryption. If users don't use end-to-end encryption, e.g., https, it is possible to eavesdrop on the circuit's exit node.



HTTPS traffic sniffing at exit node

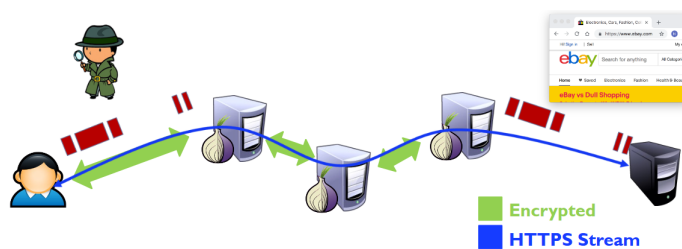
Tor exit relays tamper with HTTPS connections, e.g., by:

- Impersonating the destination website with a rogue certificate
- SSL stripping: downgrade the connection from HTTPS to HTTP



Tor Traffic Fingerprinting

A local, passive observer can monitor a web-browsing client's encrypted traffic to determine its web activity. This also works for hidden services. One can control the Tor entry point for the computer hosting the hidden service and fingerprint it.



3.5 Botnets

A botnet is a collection of software "robots" that run on host computers autonomously and automatically, controlled remotely by an attacker or attackers.

Botmasters are not necessarily (only) botnet's end-beneficiaries. They may be paid by advertising companies, or companies that send spam, viruses, or other malware.

During a bot's installation, the malware typically installs what is known as a backdoor, or a program that allows the bot master to communicate, control and install software onto the infected computer.

3.5.1 Infecting Botnet Nodes

The most popular methods of spreading a botnet are:

- Email
- Pirated software

On the other hand, if one wishes to recruit a botnet, the most popular method is by drive-by download, while visiting a malicious site.

3.5.2 Trail Obfuscation Techniques

Some techniques to obfuscate a botnet's trail are:

- Piggyback on existing protocols and systems
- Fast Flux and Domain Generation Algorithms (DGA)
- P2P botnet architectures
- Encryption
- Rootkits

Internet Relay Chat (IRC) is often used to facilitate communication in the form of text. An attacker opens private IRC channel on ordinary IRC server and waits for bots to subscribe his own private IRC channel before giving commands.

Another option is to use HTTP, where bots query dedicated web sites regularly for getting new commands.

Centralized botnets are easy to implement and have low latency, but are easier to detect and disrupt.

3.5.3 Investigating Botnets

Determining the source of a botnet-based attack is challenging. The classic approach involves an anti-virus scanner, but requires installation on every machine.

Vertical Detection

Vertical detection of single bot infections without packet inspection relies on the fact that the botmaster establishes C&C connections frequently to disseminate orders. The idea is to use these statistical properties of C&C communication (periodic behavior) in a machine learning model.

3.6 Rootkits

Exploits are malicious programs that take advantage of application software or OS vulnerabilities. Exploit kits are more comprehensive tools that contain a collection of exploits.

The behavior of the operating system can be affected by the presence of rootkits. The goals of a rootkit are:

- Enable future access to system by attacker
- Remove evidence of original attack and activity that led to rootkit installation
- Hide future attacker activity (files, net connections, processes) and prevent it from being logged
- Install tools to widen scope of penetration
- Secure system so other attackers can't take control of system from original attacker

3.6.1 Rootkit Tools

Some tools commonly used in rootkits include:

- Backdoors
- Packet sniffers
- Log-wiping utilities
- DDOS programs
- IRC programs
- Password crackers
- Hiding utilities

Some stealth techniques are:

- File masquerading
- Hooking
- Virtualization
- Direct Kernel Object Manipulation (DKOM)

3.7 Malware Analysis

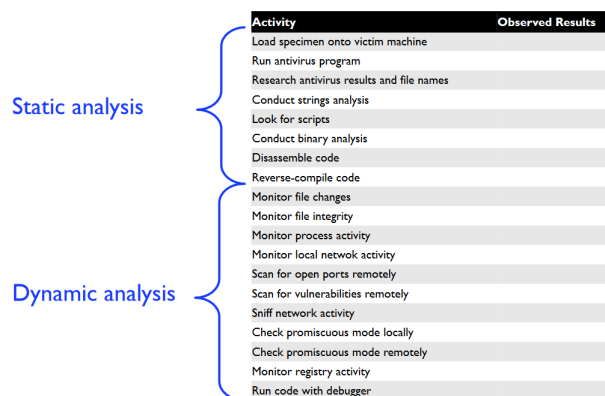
Malware may be analysed for various reasons, including the assessment of damage, identification of vulnerabilities or to aid in finding the perpetrator.

3.7.1 Analysis Techniques

Some static analysis techniques are:

- Hashing the file
- Disassembly
- Decompilation

While static analysis will reveal some immediate information, a dynamic analysis is conducted by observing and manipulating malware as it runs and may reveal more information.



Activity	Observed Results
Load specimen onto victim machine	
Run antivirus program	
Research antivirus results and file names	
Conduct strings analysis	
Look for scripts	
Conduct binary analysis	
Disassemble code	
Reverse-compile code	
Monitor file changes	
Monitor file integrity	
Monitor process activity	
Monitor local network activity	
Scan for open ports remotely	
Scan for vulnerabilities remotely	
Sniff network activity	
Check promiscuous mode locally	
Check promiscuous mode remotely	
Monitor registry activity	
Run code with debugger	

3.7.2 Anti-Analysis Techniques

Some anti-static analysis techniques include opcode obfuscation, imported function obfuscation and targeted attacks on analysis tools.

On the other hand, some anti-dynamic analysis techniques are the detection of virtualization, instrumentation and debuggers.

3.7.3 Creating a Safe Environment

When performing malware analysis it is common practice to use a virtual machine. It is important, however, to set up the machine with no network or host-only network and to take into consideration that no virtualization software is perfect. Malicious code may be able to detect it is running on a virtual environment.

3.8 Cryptocurrency Investigations

Virtual currencies are often used for online transactions.

3.8.1 Bitcoin

In the dark web, bitcoin is frequently used as currency, since it is fully decentralized and more private.

In this system, clients are identified by public keys of their choice. Peers of the Bitcoin network connect to each other over an unencrypted TCP channel. There is no authentication in the network, so each node just keeps a list of IP addresses associated with its connections.

Transaction Graph

In the transaction graph, the current transaction references previous transactions, called inputs. Other nodes verifying this transaction will check those inputs and through referenced input links, ownership of bitcoins is passed along.

To validate a transaction, one must check the entire chain all the way back to the first tx ever made: Transaction history is organized as a chain of blocks, a blockchain.

Mining

Mining is the process by which new bitcoin is added to the money supply. Miners are responsible for generating and appending new blocks to the head of the blockchain. Miners provide processing power to the bitcoin network in exchange for the opportunity to be rewarded bitcoin.

Miners compete to solve a difficult math problem, based on the cryptographic hash algorithm. The solution to the problem (Proof-of-Work) is included in the block and acts as a proof that the miner solved the problem.

3.8.2 Bitcoin Investigations

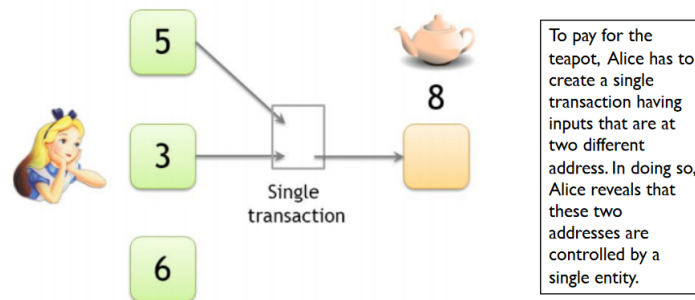
Some trail obfuscation techniques present in bitcoin include:

- Bitcoin address are not mapped to the real user identity
- Bitcoin transactions don't contain personal information
- IP address of client not included in new transactions
- User can generate as many Bitcoin addresses as needed

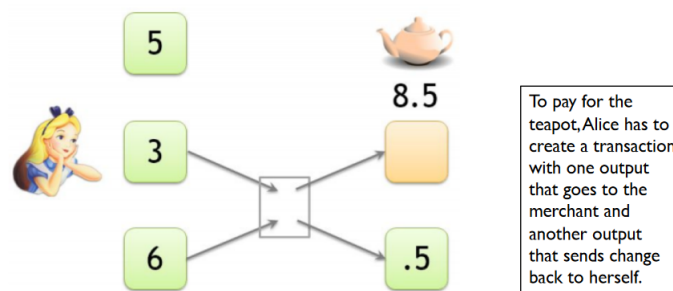
Although names are not associated with transactions, their metadata is recorded and publicly available. Some other weaknesses include:

- Authentication mechanism in Bitcoin service providers may link user IPs to Bitcoin addresses
- The chain of transactions is transparent and traceable
- Bitcoin address exposed on the Internet reveal all transactions related to its owner
- Gathering some or all inputs when sending Bitcoins to others may expose other addresses of the sender

Often times, heuristics are also applied, for example, the shared spending or the fresh-change-address heuristics. Shared spending is evidence of joint control of the different input addresses. Two inputs to the transaction are most likely under the control of the same user:



Wallet software typically generate a fresh address whenever a change address is required. Change addresses generally have never before appeared:



In summary, when investigating bitcoin, one should analyse the transaction graph and the bitcoin protocol and network.

3.9 Mobile Forensics

3.9.1 Evidence From Android Devices

By having the Linux kernel at the heart of its platform, Android tries to ensure security at the OS level. Android devices have two primary types of memory:

- NAND flash
- RAM

And the sequence of steps on the boot process is as follows:

- Boot ROM code execution
- The boot loader
- The Linux kernel
- The init process

- Zygote and Dalvik
- The system server

Data Storage

Android apps primarily store data in two locations: internal and external storage (SD card).

The internal data of all apps is saved in `/data/data/<apppkg>`. There are five methods of data storage on a device

- Shared preferences: this location provides a framework to store key-value pairs of primitive data types in the .xml format
- Files on internal storage: data stored here is private and cannot be accessed by other applications
- Files on external storage: files can be stored by the apps in external storage, which does not have strict security enforcement
- SQLite database: Android SDK provides dedicated APIs that allow developers to use SQLite DBs in their applications
- Network: The final data storage mechanism available to developers is the network (e.g. Dropbox)

3.9.2 Evidence Extraction

ADB Pull

The ADB pull command can be used to pull single files or directories directly from the device. Google implemented ADB backup functionality which allows users (and forensic examiners) to backup app data to local computer over ADB.

Screen Capture

Sometimes, taking a picture is the only way to take data off of a phone

Software-Based Physical Techniques

It is possible to extract data using an imaging tool (e.g. dd).

Hardware-Based Physical Techniques

JTAG interface: used during the device production process to communicate with the processor for testing

Chip-off: involves heating the device's circuit board until the solder holding the components to the board melts, and then removing the flash memory chip

3.9.3 Android App Reverse Engineering

The reverse engineering of apps enables repackaging attacks. The easy decompilation of DEX bytecode enables source code exposure and the simple distribution of repackaged apps make this a serious issue.

3.10 Cloud Forensics

There are three main models: IaaS, PaaS, and SaaS. Some challenges when investigating cloud networks are:

- Storage system is no longer local
- Each cloud server contains files from multiple users
- Even if data belonged to a particular subject is identified, separating it from different users is difficult
- Other than cloud service providers (CSPs), there is usually no evidence that links a given data file to a particular suspect

3.10.1 Virtual Machines

Cloud services often rely on the power of virtual machines. This brings some difficulty, since when we turn off a virtual machine, all the data will be lost if we do not have the image of the instance.

Virtual Machine Introspection (VMI) is the process of externally monitoring the runtime state of VM from either the Virtual Machine Monitor (VMM), or from some virtual machine other than the one being examined.