InStep Week 8 Presentation

Mitchell Burcheri

Timeline

- Week 1-4: Web pentesting.
- Week 5: Learning the fundamentals of threat hunting and malware analysis, obtained 4 Lex certificates, setup malware analysis environment, worked through the phase documents and started watching some malware analysis videos.
- Week 6: Start initial presentation. Learning practical static and dynamic malware analysis. Caught Covid.
- Week 7: Writing reports for malware samples. Lots of samples wouldn’t allow me to do dynamic analysis. Continued added to the presentation.
- Week 8: Worked on malicious document analysis, reverse engineering, improving my malware analysis reports, finalised and presenting my presentation.

Threat Hunting

Threats

Physical Threats

- **Internal**: short circuit, fire, unstable power supply
- **External**: natural disasters like floods and earthquakes
- **Human**: intentional or unintentional human behaviour such as destroying hardware, theft, disruption, unintentional errors

Non-Physical Threats

- Malware
- Denial of Service (DoS)
- Phishing
- **Any digital threat that could lead to disruption of a system or computers**. Any form of digital hacking.

Attack Methods

- Malware
- Phishing
- Smishing
- Man-In-The-Middle
- Distributed Denial of Service (DDoS)
- SQL Injection
- Drive-by-attack (attacker injects code on a site which infects users who view the page.)
- Zero-day exploit
- Crypto-Jacking
- DNS Tunneling
- IoT-based Attacks
- Cross Site Scripting (XSS)
- Social Engineering

Cyber Threat Landscape

As technology grows, the attack surface also grows with an increasing number of threats. **Implementation can lead to vulnerabilities**.

The threat landscape includes vulnerabilities, malware, threat actors, techniques and tactics, etc.

Attackers may target organisations for valuable information, monetary benefits, and geopolitical factors, and attackers will target organisations, sectors, or individuals when performing attacks.

Threat Hunting

Threat hunting is a **proactive defense approach** where you are actively searching for indicators of a threat of a breach. You must be using an **offensive strategy** where you must think like an attacker and must always assume there might be a breach.

Threat hunting is **important for protecting against sophisticated attacks where the traditional approach of security does not stop a skilled hacker from breaking in**. A company's' defence is only as good as its threat intelligence. The **average time taken to detect a break is 56 days**. The security operations center (SOC) and **automated security defenses may stop 70% - 80%** of attacks but **the remaining 20% - 30% of attacks are from more sophisticated threats**.

To be able to hunt threats you must have a clear understanding of how an adversary behaves, knowledge of the Cyber Kill Chain, the tools and techniques used by attackers, how to prevent an attack from progressing along the kill chain, use threat intelligence and digital forensics.

Detect the intruder, prevent intruder from further access, and remove the intruder from the network.

Every industry is at risk of attack.

The importance of Threat Hunting

Adversaries leave their tracks everywhere. Email logs, system logs, HTTP proxy logs, registry files, database logs, security logs, and network logs.

Threat detection is a passive approach (not proactive) and **can only detect known threats**. It is the process of detecting and isolating threats from the network. Firewall, Intrusion Prevention System (IPS), Intrusion Detection System (IDS), etc.

Threat hunting is a proactive approach and **searches for undetected threats**. Threat hunters search through networks, endpoints and systems for malicious activity.

Threat Hunter’s Responsibilities and Skills

Data Analytics: Monitor and analyse data, and recognise patterns in data for malicious activity so threats can be detected.

Forensics: Investigate the root cause of attacks and to reverse engineer the malicious software to understand how the malware affects the system and to identify the adversaries.

Network Knowledge: Deep understanding how networks work and normal and abnormal network behaviour.

A Threat Hunter's Maturity Model

- **Initial** (Level 0): Relies on automated alerting and there is little or no routine data collection
- **Minimal** (Level 1): Incorporates threat intelligence indicator searches and has a moderate or high level of routine data collections
- **Procedural** (Level 2): Follows data analysis procedures created by others and has a high or very high level of routine data collection.
- **Innovative** (Level 3): Creates new data analysis procedures and has a high or very high level of routine data collection.
- **Leading** (Level 4): Automates majority of successful data analysis procedures and has a high level of routine data collection.

Factors that influence a threat hunters hunting capabilities:

- Quality and quantity of the data that an organisation provides
- The tools for analysing and visualising the data
- Automated analytics provided by the organisation

Cyber Kill Chain

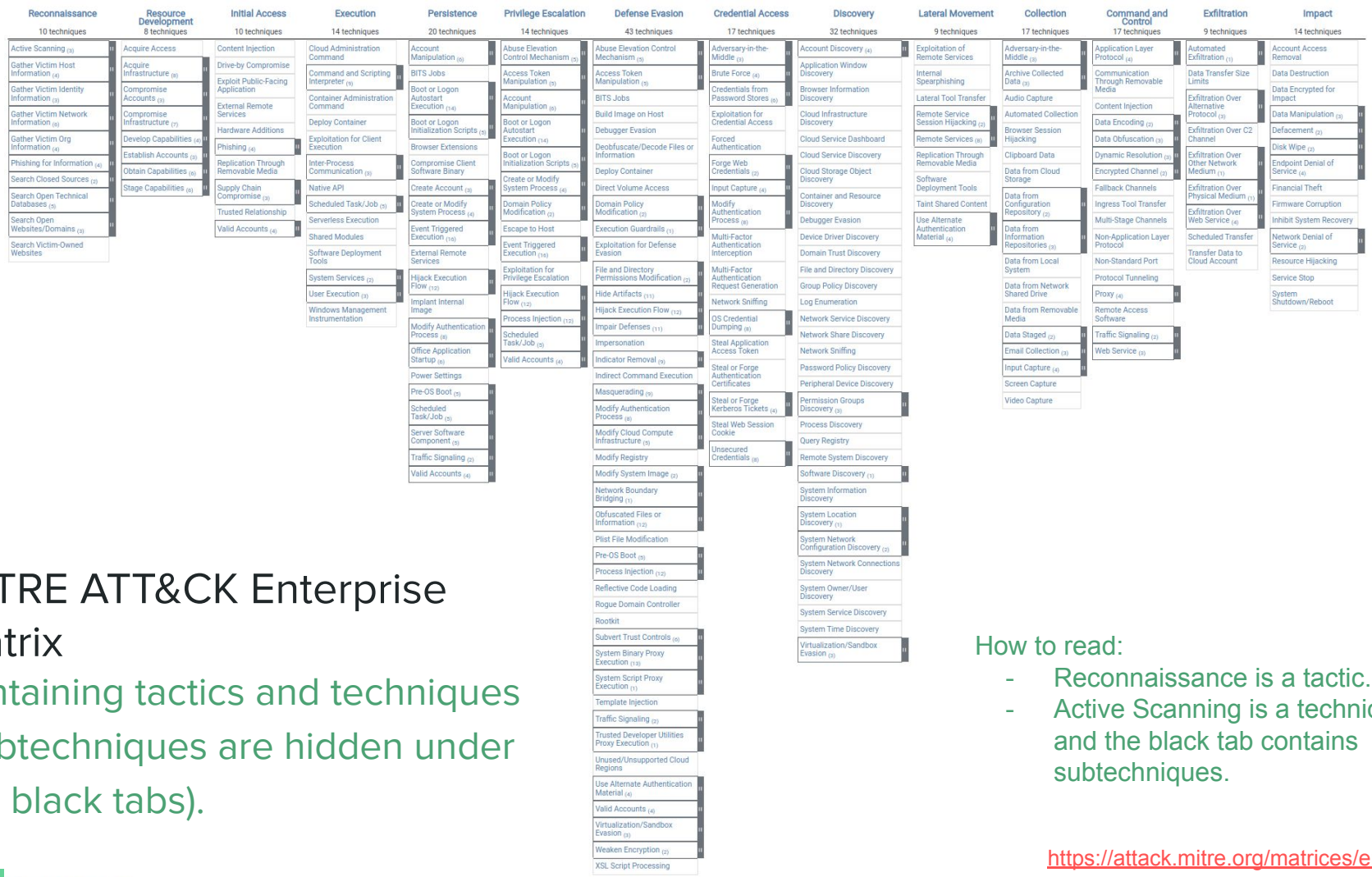
1. **Reconnaissance** - Gathers information about the target including personal information, vulnerabilities and weaknesses.
2. **Weaponization** - Prepare to gain access to the environment based on the recon data and build a payload to achieve access.
3. **Delivery** - Sends the payload/weapon to the target via email, malicious USB, vulnerable web components, social engineering.
4. **Exploitation** - Malicious content is executed on the target leveraging the vulnerability and compromises the network.
5. **Installation** - Once the exploit is executed, a backdoor is installed to access the network and additional tools are downloaded required for pivoting.
6. **Command and Control (C2)** - Once the backdoor is installed, the malware tries to establish a connection to the attackers domain or IP address so the attacker will have “hands on keyboard” access to the network.
7. **Action on objectives** - Attacks take actions to achieve their goal which may be by exfiltrating data, destroying data, encrypting data for ransom, etc.

MITRE | ATT&CK

MITRE ATT&CK is a large knowledgebase of adversarial techniques showing us how an adversary interacts with systems. It organises techniques into tactics which is relevant to Red and Blue (attacking and defending) teams. Contains tactics, techniques, mitigations, groups, and software.

Tactics, Techniques, and Procedures (TTP)

- Tactics: Enter a network
- Techniques: Achieve goals, pivoting
- Procedures: Indicators of Compromise (IOC)



MITRE ATT&CK Enterprise Matrix

containing tactics and techniques (subtechniques are hidden under the black tabs).

How to read:

- Reconnaissance is a tactic.
- Active Scanning is a technique and the black tab contains subtechniques.

https://attack.mitre.org/matrices/enterprise/

Common Threat Hunting Terms

- Advanced Persistent Threat (APT)
- Indicators of Compromise (IOC)
- Pyramid of Pain
- Diamond Model

Advanced Persistent Threat (APT)

An Advanced Persistent Threat (APT) is a sophisticated cyber attack where the attacker achieves undetected access to a network to steal sensitive information. Their aim is to maintain persistence which allows them to be a system for a long time undetected. These attacks require a lot of skill and research to identify vulnerabilities and to evade all security access controls. APTs groups are generally well funded and the members are knowledgeable and experienced.

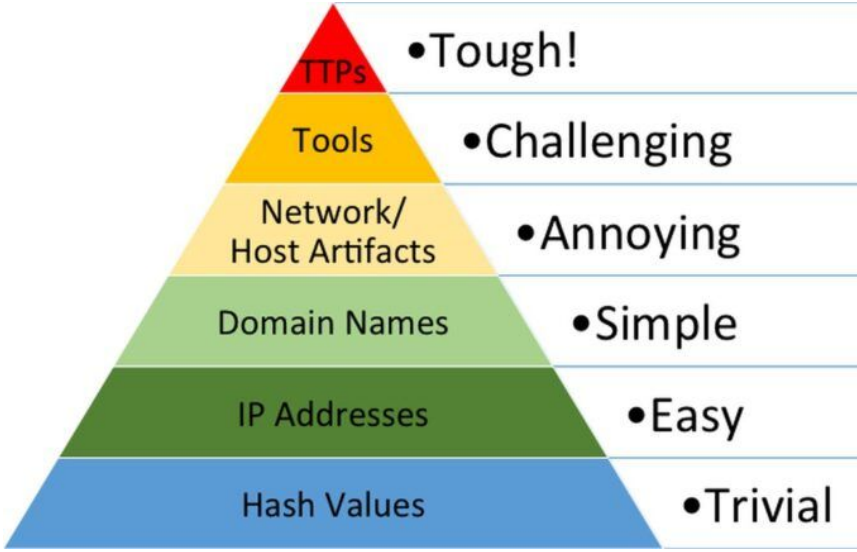
The goal of an APT could be cyber espionage, financial gain, hacktivism, and destruction, and their targets can be governments, industries, organisations, defense systems, and whoever the want to target.

Indicators of Compromise (IOC)

Indicators of Compromise (IOC) are a piece of forensic evidence, such as data, that indicates a breach and helps detect malicious activity. Security teams monitor for IOC and try to detect them in the early stages to prevent attacks from developing. IOCs contain malware signatures such as hash values, IP addresses, domain names, network/host artifacts, tools, and TTPs.

Pyramid of Pain

The Pyramid of Pain demonstrates that some Indicators of Compromise (IOC) are more troubling for attackers to change.



The levels of difficulty for an adversary to change.

Hash values: These can easily be changed if a single bit is changed in a program.

IP addresses: New IP addresses can be assigned.

Domain names: Domains can be changed.

Network Artifacts: Network traffic protocols, HTTP User-Agent, SMTP Mailer values, etc.

Host Artifacts: Registry keys or values known to be created by specific pieces of malware, files, or directories.

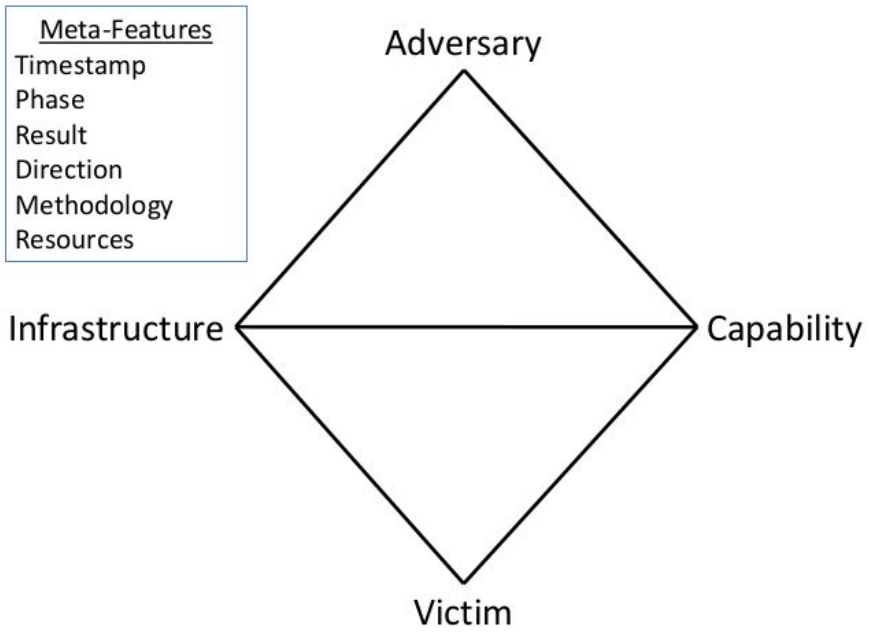
Tools: Software used my the attackers to achieve their mission such as software that create malicious documents for phishing, backdoors to establish command and control, password crackers, and host-based utilities.

TTPs: How an adversary achieves their goals such as every stage of the Cyber Kill Chain.

Diamond Model

An adversary has the capability to attack the victims infrastructure which effects the victim.

The victim discovers the malware which contains information on the command and control server which reveals the infrastructure the adversary is using and reveals the adversary.



Threat Hunting Process

Threat Intelligence

Threat Intelligence (TI) is about the data on threats. Threat intelligence contains processed, analysed and actionable data regarding threats such as Indicators of Compromise (IOC) and Tactics, Techniques, and Procedures (TTPs). Threat Hunting relies on Threat Intelligence databases for information on known types of threats.

Categories of Threat Intelligence:

- **Strategic**: Meant for a non-technical audience and outlines trends of threat actors.
- **Tactical**: Meant for a technical audience and outlines TTPs of threat actors.
- **Operational**: Contains technical details about specific attacks and campaigns.

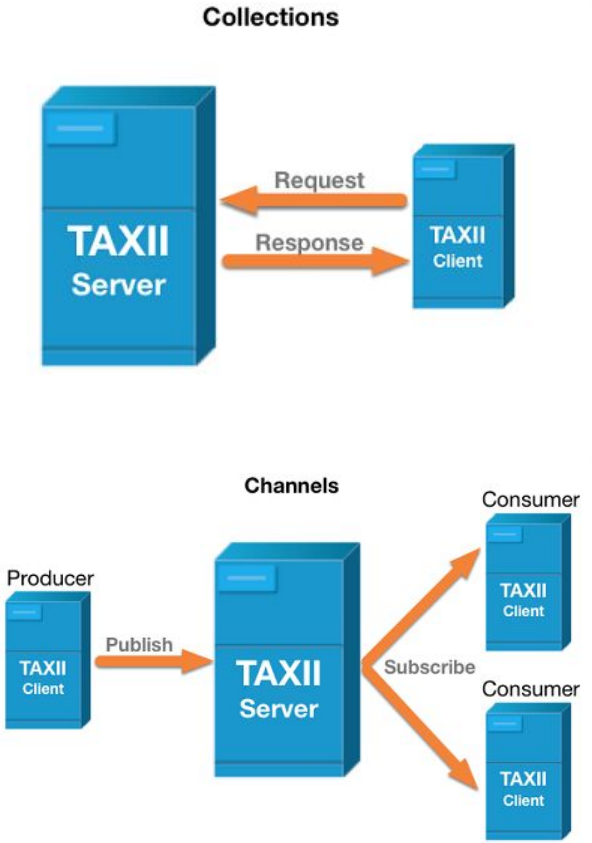
STIX

Structured Threat Information Expresion (STIX) is a standardised language for **describing cyber threat information** and is **used to exchange cyber threat intelligence**. STIX is structured to describe the threats motivations, abilities, capabilities and response.

TAXII

Trusted Automated Exchange of Intelligence Information (TAXII) is a**n application protocol for exchanging cyber threat intelligence (CTI)** over HTTPS. TAXII defines a RESTful API (a set of services and message exchanges) and a set of requirements for TAXII Clients and Servers. Two primary services are used to support common sharing models:

- Collections: An interface that allows a producer to host a set of CTI data.
- Channels: Allows producers to push data to consumers.



Malware Analysis

Malware

Malware stands for malicious software and is a piece of code that is used to disrupt the operations of a system or to take control of a system to steal information or perform large scale attacks.

Types of malware:

- Virus - inserts itself into existing programs and alters the program's behaviour
- Worm - a program that self replicates itself to spread to other hosts
- Spyware - gathers information about the target and sends it back to the attacker
- Adware - software used to spam host with ads
- Ransomware - encrypts data and threatens to leak or delete it unless a ransom is paid
- Trojan Horse - innocent on the outside and malicious on the inside
- Botnet - a network of bots
- Advanced Persistent Threat (APT) - advanced threat that maintains persistence

Malware Analysis

Malware analysis is the process of extracting information from malware through static and dynamic analysis. There are many different tools, techniques and processes used to extract data from malware which include IP addresses, domains, hashes, signatures, threat actors, tactics, techniques and procedures (TTPs), etc.

Malware analysis is required to determine the nature of the malware and it’s interactivity with the file system, machine and network.

Types of Malware Analysis (Static Analysis)

Static Analysis - Extracting information from malware without executing it.

- ID Assignment: MD5, SHA1, are SHA256 are used to verify the integrity of a file.
- Filetype Identification: helps identify the target OS and the type of file from the file headers which can be found using a hex editor.
- String Analysis: can reveal malicious operations and resources.
- Antivirus Detection: VirusTotal and similar tools can help identify the malware.
- Protective Mechanism Identification: obfuscation is used to hide the code from the the analyst to hide how the code works.
- PE (Portable Executable) Structure Verification: A malformed PE header is usually a sign of either a corrupt file or a deliberate attempt to hide malware.
- Reverse Engineering: malware is disassembled so the functionality is uncovered.

Types of Malware Analysis (Dynamic Analysis)

Dynamic Analysis - Extracting information about the malware by executing it and investigating it’s behaviour.

- **Host Behaviour** is about malware component installation, persistency, protective mechanisms, and basic actions.
- **Network Behaviour** is about its access to remote domains and IP addresses, files downloaded, and data sent.

Host Behaviour

- **Component Installation**: is related to the malware being installed via a malware dropper, downloader, or macro. Common folders malware is installed to are the windows folder, system folder, and temporary folder.
- **Persistency**: malware tries to hide itself to achieve persistence. This can be achieve by hijacking the boot sector, infecting system files, adding itself to the startup folder, using task scheduler, and utilising the registry.
- **Protective Mechanisms**: malware authors implement protective mechanisms to avoid detection and analysis. Malware can be polymorphic, metamorphic, obfuscated, encrypted, and can terminate antimalware.
- **Basic Actions**: include monitoring the file system, creating, modifying and deleting files, and registry monitoring.

Network Behaviour

- **Remote domain or IP addresses**
- **Files that are downloaded**
- **Data that is sent**

The process of analysis

1. You believe a file is malicious.
2. You load the file into an virtual machine.
3. You run a static analysis of the file to identify what the file may do and write YARA rules for identifying the file.
4. You isolate your environment to begin running the dynamic analysis where you infect your computer and see what the program does on a new machine to learn amount it’s behaviour and what it installs once it’s on the machine so the right steps can be taken to remove the malware.

Where I found my malware samples for malware analysis

https://github.com/Virus-Samples/Malware-Sample-Sources

theZoo: https://github.com/ytisf/theZoo

Malware Bazaar: https://bazaar.abuse.ch/

Setting up a malware analysis lab

Virtual Machine Notes

Install virtualbox > install windows 10 iso into virtualbox (I allocated 4096 MB of base memory, 3 processors, NAT network settings for installation process, after a failed install go to storage and remove the floppy disk then rerun) > do a normal and legitimate installation. Follow the advanced installation to install without the product key. After installation, take a snapshot.

Go to settings and pause windows updates and disable all the microsoft defender features and disable the firewall. In the windows security/defender features there is a file exclusion feature. Exclude the directory that malware is installed to and before running a dynamic analysis add the C drive to the list of excluded directories. There is a file I use to disable the microsoft defender real-time scan registry. Before running malware, take a snapshot.

After windows has been installed, install flarevm from github. You will need to go to powershell as administrator and run this command [Set-ExecutionPolicy unrestricted] before downloading. After flare vm has been installed take a snapshot. Then install additional software and take a snapshot after all the tools that are needed are installed. Then install malware, and take a snapshot before execution.

Remnux config

On remnux: cd /etc/inetsim; sudo nano inetsim.conf;

Uncomment start_service dns; service_bind_address 0.0.0.0 (we set this address); dns_default_ip <ip of the remnux vm>.

Start dns server with inetsim

On windows set the dns server to the remnux box in your control panel config select ethernet and change the settings in ipv4 the remnux ip. The use ipconfig /flushdns.

Safety precautions (before dynamic analysis)

- Make sure network settings in virtual machine are set to host-only adapter
- No file sharing
- No usb sharing
- No file or text sharing
- Take a snapshot when the above are done.
- I always write a junk file on the home desktop to tell myself that the malware has been activated so I don’t change network settings early without checking that the malware isn’t active. That way when I restore an older snapshot I know there is no malware running if the junk file isn’t present.

Important note, if you don’t know what a tool does, run the environment with no network adapter or host-only adapter. I have accidentally ran ransomware when experimenting with a debugger. I was lucky to have had my virtual machine network settings set no network adapter.

Huge Note!

Run malware as Administrator if nothing happens on normal execution.

Static Analysis tools I used the most for executables

- **CAPA** to understand the behaviour of the malware. Also generates file hashes.
- **Pestudio** analyses the raw data for me, extracting the important strings.
- **HxD** allows me to look at the file headers of a file.
- **YARA** used to use YARA rules to find matching files.
- **VirusTotal** shows me if the file has previously been detected as malicious and details about the malware.
- **Hybrid Analysis** shows me if the file has been detected as malicious and conducts an analysis for a specified machine so I can compare their results with mine.
- **Malware Bazaar** has good metadata details that I used in my report.
- **Ghidra** for reverse engineering code.

Static Analysis Tools for malicious document analysis

ExifTool to get meta information about the sample.

Strings to look interesting strings.

Xorsearch to look for decrypted strings.

olemeta to get meta information about the sample.

oleid to detect if the file may cause any risks.

olevba to view the visual basic code and can be used to deobfuscate a large chunk of the code. It also shows suspicious strings.

Dynamic Analysis Tools

Remnux **inetsim** for DNS simulation / fake-dns on the windows 10 vm seemed to do the same thing and it was better because it showed me traffic that was being sent to fake dns. Remnux has its own version of fake-dns but windows is better.

Fake-NG used as a DNS and shows the DNS queries made.

ProcMon allows me to record and monitor call the processes from the time I start process monitor.

WireShark allows me to view the network traffic from my machine.

RegShot records and compares the before and after shot of the registry to report any changes.

I still need to try TCPview.

Malware Analysis Report Showcase

- Link to my malware analysis reports: https://github.com/Burchonator/InStep/tree/main/week8-reports

Lex Certificates

- SOC - Threat Hunting and Malware analysis
- Purdue ACE Malware Analysis and Threat Hunting
- Offensive Security Certified Professional (OSCP) Certification Guide
- DVWA Mastering Web Application Security

Thank you for selecting me for the internship

Thank you for selecting me for the internship. It has been a excellent experience and opportunity for me to develop my web pentesting skilling and develop new skills with malware analysis.