Topics of Ch-2

- Current and Next-Generation Malicious Software: Viruses, Worms, Trojans, Botnets
- Polymorphic and Metamorphic Malware
- Advanced Persistent Threats
- Intro to Defensive Strategies Against Malware
- Worm Fingerprinting / Signature Generation
- Behavioral Approaches to Detection of Malware

Types of Malware

- **Virus**
- Viruses are a subgroup of malware. A virus is malicious software attached to a document or file that supports macros to execute its code and spread from host to host. Once downloaded, the virus will lie dormant until the file is opened and in use.
- A virus usually comes as an attachment in an email that holds a virus payload, or the part of the malware that performs the malicious action. Once the victim opens the file, the device is infected.
- **Worms**
- A worm is a type of malicious software that rapidly replicates and spreads to any device within the network. Unlike viruses, worms do not need host programs to disseminate. A worm infects a device through a downloaded file or a network connection before it multiplies and disperses at an exponential rate. Like viruses, worms can severely disrupt the operations of a device and cause data loss.
- Worms have the ability to copy themselves from machine to machine, usually by exploiting some sort of security weakness in a software or operating system and don’t require user interaction to function. Worms often attack a computer’s memory or hard drive

Types of Malware

- **Ransomware**
- One of the most profitable, and therefore one of the most popular, types of malware amongst cybercriminals is ransomware.
- Ransomware is malicious software that gains access to sensitive information within a system, encrypts that information so that the user cannot access it, and then demands a financial payout for the data to be released. Ransomware is commonly part of a phishing scam. By clicking a disguised link, the user downloads the ransomware.
- **Trojan**
- Trojan viruses are disguised as helpful software programs. But once the user downloads it, the Trojan virus can gain access to sensitive data and then modify, block, or delete the data. This can be extremely harmful to the performance of the device. Unlike normal viruses and worms, Trojan viruses are not designed to self-replicate.
- Trojans masquerade as harmless applications, tricking users into downloading and using them. Once up and running, they then can steal personal data, crash a device, spy on activities or even launch an attack.

Types of Malware

- **Spyware**
- Spyware is malicious software that runs secretly on a computer and reports back to a remote user. Rather than simply disrupting a device's operations, spyware targets sensitive information and can grant remote access to predators. Spyware is often used to steal financial or personal information. A specific type of spyware is a keylogger, which records your keystrokes to reveal passwords and personal information.
- **Adware**
- Adware is malicious software used to collect data on your computer usage and provide appropriate advertisements to you. While adware is not always dangerous, in some cases adware can cause issues for your system. Adware can redirect your browser to unsafe sites, and it can even contain Trojan horses and spyware.
- Adware programs push unwanted advertisements at users and typically display blinking advertisements or pop-up windows when you perform a certain action. Adware programs are often installed in exchange for another service, such as the right to use a program without paying for it.

Types of Malware

- **Fileless malware**
- Fileless malware is a type of memory-resident malware. As the term suggests, it is malware that operates from a victim's computer's memory, not from files on the hard drive. Because there are no files to scan, it is harder to detect than traditional malware. It also makes forensics more difficult because the malware disappears when the victim computer is rebooted.
- It uses legitimate programs to infect a computer. Fileless malware registry attacks leave no malware files to scan and no malicious processes to detect **OR** is a kind of attack that uses vulnerabilities in legitimate software programs like web browsers and word processors to inject malicious code directly into a computer's memory.
- Many fileless malware attacks use PowerShell, a command line interface and scripting tool built into Microsoft Windows operating systems. Hackers can execute PowerShell scripts to change configurations, steal passwords, or do other damage.
- **Ex:** Malicious macros are another common vector for fileless attacks. Apps like Microsoft Word and Excel allow users to define macros, sets of commands that automate simple tasks like formatting text or performing calculations. Hackers can store malicious scripts in these macros; when a user opens the file, those scripts automatically execute

Other Types of Malwares

- **Bots/Botnets**
- A bot is a software application that performs automated tasks on command. They're used for legitimate purposes, such as indexing search engines.
- A botnet is a network of internet-connected, malware-infected devices under a hacker's control. Botnets can include PCs, mobile devices, Internet of Things (IoT) devices, and more.
- **EX:** Hackers often use botnets to launch DDoS attacks, which bombard a target network with so much traffic that it slows to a crawl or shuts down completely.
- **Rootkit/ Remote access malware**
- Rootkits were not originally designed as malware, but they have become a common attack vector for threat actors.
- A rootkit is software that gives malicious actors remote control of a victim's computer with full administrative privileges. Hackers can then use these elevated permissions to do virtually anything they want, like adding and removing users or reconfiguring apps. Hackers often use rootkits to hide malicious processes or disable security software that might catch them.

Other Types of Malwares

- **Scareware**
- Cybercriminals scare us into thinking that our computers or smartphones have become infected to convince victims to purchase a fake application. In a typical scareware scam, you might see an alarming message while browsing the Web that says “Warning: Your computer is infected!” or “You have a virus!” Cybercriminals use these programs and unethical advertising practices to frighten users into purchasing rogue applications.
- **Cryptojackers**
- A cryptojacker is malware that takes control of a device and uses it to mine cryptocurrency, like bitcoin, without the owner's knowledge. Essentially, cryptojackers create cryptomining botnets. Mining cryptocurrency is an extremely compute-intensive and expensive task. Cybercriminals profit while users of infected computers experience performance slowdowns and crashes.
- **Ex:** Cryptojackers often target enterprise cloud infrastructure, allowing them to marshal more resources for cryptomining than targeting individual computers.

Other Types of Malwares

- **Mobile Malware**
- As the name suggests, mobile malware is designed specifically to target mobile devices. This kind of malware has become more common not just with the proliferation of smart phones, but with the increase of mobile and tablet use by organizations and employees.
- Mobile malware can employ several tactics, including spying and recording texts and phone calls (a form of spyware), impersonating common apps, stealing credentials, or accessing data on the device. Mobile malware often spreads through **smishing, also known as SMS phishing**, which is a form of phishing that comes through text messages.
- Other forms of mobile malware include remote access tools, bank Trojans, and crypto mining malware. As phones become a more valuable tool in the workplace, often including the device used in MFA applications, they become a larger target for threat actors.
- **Wiper Malware**
- A wiper is a type of malware with a single purpose: to erase user data and ensure it can't be recovered. Wipers are used to take down computer networks in public or private companies across various sectors. Threat actors also use wipers to cover up traces left after an intrusion, weakening their victim's ability to respond.

Polymorphic and Metamorphic Malware

- **What is obfuscation?**
- Obfuscation means to make something difficult to understand.
- Programming code is often obfuscated to protect intellectual property or trade secrets, and to prevent an attacker from reverse engineering a proprietary software program.
- Encrypting some or all of a program's code is one obfuscation method.
- Other approaches include stripping out potentially revealing metadata, replacing class and variable names with meaningless labels and adding unused or meaningless code to an application script.
- **A tool called an obfuscator** will automatically convert straightforward source code into a program that works the same way, but is more difficult to read and understand.
- Unfortunately, malicious code writers also use these methods to prevent their attack mechanisms from being detected by antimalware tools.
- **Ex.** The 2020 SolarWinds attack is an example of hackers using obfuscation to evade defenses.

Obfuscation

- Ex: https://obfuscator.io/#code
- **How does obfuscation work?**
- Obfuscation in computer code uses complex roundabout phrases and redundant logic to make the code difficult for the reader to understand. The goal is to distract the reader with the complicated syntax of what they are reading and make it difficult for them to determine the true content of the message.
- With computer code, the reader may be a person, a computing device or another program. Obfuscation is also used to fool antivirus tools and other programs that rely heavily on digital signatures to interpret code.
- Decompilers are available for languages such as Java, operating systems such as Android and iOS, and development platforms like .NET. They can automatically reverse engineer source code; obfuscation aims to make it difficult for these programs to do their decompiling as well.
- Code obfuscation is not about changing the content of a program's original code, but rather about making the delivery method and presentation of that code more confusing. Obfuscation does not alter how the program works or its end output.

Obfuscation techniques

- Obfuscation involves several different methods. Often, multiple techniques are used to create a layered effect.
- Some common obfuscation techniques include the following:
- **Renaming:** The obfuscator alters the methods and names of variables. The new names may include unprintable or invisible characters.
- **Packing:** This compresses the entire program to make the code unreadable.
- **Control flow.** The decompiled code is made to look like spaghetti logic, which is unstructured and hard to maintain code where the line of thought is obscured. Results from this code are not clear, and it's hard to tell what the point of the code is by looking at it.
- **Instruction pattern transformation.** This approach takes common instructions created by the compiler and swaps them for more complex, less common instructions that effectively do the same thing.
- **Dummy code insertion.** Dummy code can be added to a program to make it harder to read and reverse engineer, but it does not affect the program's logic or outcome.

Obfuscation techniques

- **Metadata or unused code removal:** Unused code and metadata give the reader extra information about the program, much like annotations on a Word document, that can help them read and debug it. Removing metadata and unused code leaves the reader with less information about the program and its code.
- **Opaque predicate insertion:** A predicate in code is a logical expression that is either true or false. Opaque predicates are conditional branches -- or if-then statements -- where the results cannot easily be determined with statistical analysis. Inserting an opaque predicate introduces unnecessary code that is never executed but is puzzling to the reader trying to understand the decompiled output
- **Anti-tamper:** These tools detect code that has been tampered with, and if it has been modified, it stops the program.
- **String encryption:** This method uses encryption to hide the strings in the executable and only restores the values when they are needed to run the program. This makes it difficult to go through a program and search for particular strings.
- **Code transposition:** This is the reordering of routines and branches in the code without having a visible effect on its behavior.

Polymorphic and Metamorphic Malware

- Metamorphic and polymorphic malware are two types of malicious software (malware) that can change their code as they propagate through a system.
- The main difference between them is that polymorphic malware can morph itself to change its code using a variable encryption key, whereas metamorphic malware rewrites its code without an encryption key.
- Polymorphic malware is more common with most malware executables falling under this category.
- In contrast, metamorphic malware is more complex and hugely transformative, which enables it to evade traditional detection methods.

Polymorphic Malware

- **A polymorphic virus** is a complex virus that is **encrypted with a variable key** so that each copy of this virus differs from the other. The purpose of this virus is to hide from anti-malware or scanners.
- Any pest can be detected through anti-malware or scanner, but this virus is smart, it has learned to pick up different encryption keys. For example, the user downloaded a file on the website, then the second user entered the same site and downloaded the same file.
- However, the two downloaded files don’t look the same for security programs.
- Polymorphic malware uses an encryption key to change its shape and signature. It combines a mutation engine with self-propagating code to change its appearance continuously and rapidly morph its code.

Polymorphic Malware

- In the normal course of action, a scanner or anti-malware could detect the virus through two identical keys in different files, but a polymorphic virus uses different encryption keys on different files, making the task more difficult than it seems.
- Therefore, there are two methods by which it is possible to detect polymorphic viruses.
- This is a **general description of the technology** and an **algorithm of the input point**. The general description technology allows the file to be run on a protected virtual computer. The login algorithm provides machine code verification at the point of each file, so it uses software virus detection.

Polymorphic Malware

- This type of malware exists in multiple forms, such as the following:
- Viruses, bots, Trojans, keyloggers & Worms

- Polymorphic malware consists of two parts, namely:
- **Encrypted virus body.** Code that changes its shape.
- **Virus decryption routine.** Code that doesn't change its shape and decrypts and encrypts the other part.
- Since only one part changes its shape, while the other remains the same, it's easier to detect polymorphic malware than metamorphic malware.

Polymorphic Malware

- Bad actors can use one of the following obfuscation techniques to create polymorphic malware:
  - **Dead-code insertion**: which randomly injects dead code throughout a program.
  - **Subroutine reordering**: in which the ordering of the code's subroutines is changed in a randomized way so that it is harder for antivirus programs to detect it.
  - **Register reassignment**: which changes registers to newer generations, while retaining the program code and behavior.
  - **Instruction substitution:** which changes code by replacing some instructions with equivalent ones.
  - **Code transposition**: in which routines and branches in the code are reordered without having a visible effect on its behavior.
  - **Code integration**: in which the malware integrates itself in the target program and produces a new version of the target program.

Polymorphic Malware

- **Polymorphic malware risk factors**
- Bad actors use polymorphic malware to take advantage of poor cybersecurity hygiene among employees, as well as undetected zero-day vulnerabilities.
- When careless or clueless employees click on a malicious attachment in a phishing email or enter sensitive information into a phishing (fake) website, they leave the enterprise network and data vulnerable to polymorphic malware attacks.
- **WannaCry**, a worm that spreads by exploiting Windows OS vulnerabilities.
- **CryptoLocker**, a virus that changes virtual servers into encrypted data blocks.
- **Virlock**, a type of ransomware that weaponizes files it infects and spreads through the cloud like a virus.
- **CryptXXX**, a Windows ransomware that is distributed using the Angler exploit kit.

Metamorphic Malware

- A metamorphic virus can be transformed due to its ability to edit, rewrite and translate its own code.
- The purpose of the virus is to damage the computer but to make it so that it is unnoticed by anti-malware. Metamorphic virus does not use encryption keys to change its copies.
- The virus converts its existing instructions into functionally equivalent instructions when creating its copy. This is why the virus cannot return to its original form. This is the moment that complicates the work of anti-malware programs.
- There are two methods to detect metamorphic viruses:
- **using emulators** for tracking and **geometric detection**.

Metamorphic Malware

- In other words **It will reprogram itself**. What could it mean? The virus tries to outmaneuver the antivirus and transmits its own code and at the same time creates a temporary representation. After it has bypassed the security, it is written back into the normal code. Copies of this virus are always differrent, making it difficult for anti-malware to detect these copies.
- Metamorphic malware is rewritten with each iteration without using an encryption key. After each iteration, the new version becomes more sophisticated, although it functions the same way as before.
- This malware is body-polymorphic, meaning a new instance (body) of the malware is created instead of generating a new decryptor.

Metamorphic Malware

- As with polymorphic malware, obfuscation techniques are used to create new instances of metamorphic malware. Often, malware authors use **multiple transformation techniques**
- Metamorphic malware reprograms itself by translating its own code and then rewriting it to ensure that subsequent copies appear different with each iteration. No part of the malware remains constant, nor does the malware ever return to its original form.
- That's why this malware is more difficult to detect and identify using signature-based antivirus software or other cybersecurity tools.

DIFFERENCE

- Polymorphic virus involves changing each copy of its code to bypass anti-malware protection, while Metamorphic Virus with each iteration rewrites its own code.
- The polymorphic virus uses the encryption key to change its code, while Metamorphic Virus itself rewrites its code.
- Writing Metamorphic Virus is much more difficult for a programmer than creating a Polymorphic one, because you need to use several methods of conversion.
- Methods for detecting these two viruses are different. In the case of polymorphic viruses, we need such methods: general description technology and input point algorithms. And in the case of Metamorphic Virus, you need to use the following methods: the use of emulators for tracking and geometric detection.

Defensive Strategies Against Malware

Malware response plan

Keep these six basic steps in mind when creating your malware response plan

1. **Identify:** Identify which endpoints have been impacted by the attack.
2. **Communicate:** Once the impact of the attack and the point of entry have been identified, communicate your findings to necessary parties ASAP.
3. **Block:** If possible, block any further access from the origin of the malware, such as the originating website, email or IP address.
4. **Restore:** Put affected data back in a known-good state where there is no chance of malware remaining. This can be done with reimaging, rebuilding or a combination of the two.
5. **Recover:** Recover as much affected data as you can using available backups. This is particularly applicable to ransomware attacks.
6. **Re-examine:** Sit back and take a hard look at your current security strategy and what allowed the malware to get through in the first place. By analyzing and sealing these gaps, you protect your organization from a similar attack in the future.

Defensive Strategies Against Malware

- organizations should patch all known vulnerabilities to minimize the threat of successful polymorphic and metamorphic malware attacks. They should also step up their efforts to find and patch zero-day vulnerabilities.
- Employee training is also critical to ensure that they know how to recognize and resist phishing scams that leave the door open to polymorphic and metamorphic malware attacks.
- Upgrading all software, including operating systems, and ensuring that vendors provide all necessary security updates are crucial.

Defensive Strategies Against Malware

- Implementing multifactor authentication to minimize password-related attacks.
- Deploying behavior-based detection tools to identify suspicious behaviors and take immediate action.
- Using endpoint detection and response tools to narrow down threats in real time.
- Strengthening the security posture using standard frameworks, like Mitre ATT&CK.
- Employing heuristic analyses to scan for shared components of threats.
- Tightening access controls, especially to sensitive data and business-critical systems.
- Employing advanced antispam and antiphishing software to identify, quarantine and remove suspicious emails.

Advanced Persistent Threats (APT)

- **Threat actors** are individuals or groups that attack digital devices, networks or computer systems.
- Threat actors, also known as cyber threat actors or malicious actors, are individuals or groups that intentionally cause harm to digital devices or systems. Threat actors exploit vulnerabilities in computer systems, networks, and software to perpetuate a variety of cyberattacks, including phishing, ransomware, and malware attacks.
- An advanced persistent threat (APT) is a prolonged and targeted cyber attack in which an intruder gains access to a network and remains undetected for an extended period.
- APT attacks are initiated to steal highly sensitive data rather than cause damage to the target organization's network. The goal of most APT attacks is to achieve and maintain ongoing access to the targeted network rather than to get in and out as quickly as possible.

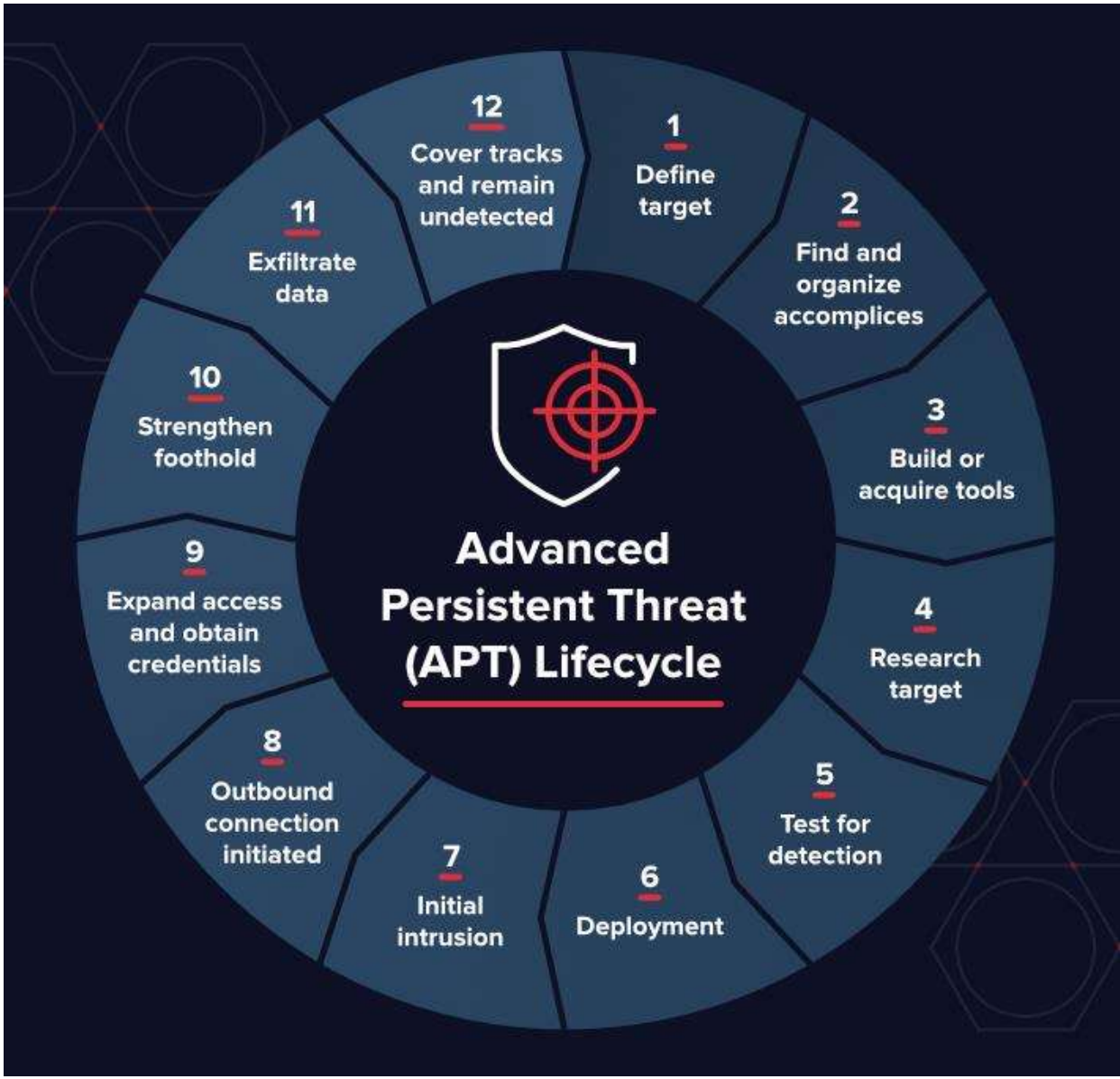
APT

- To gain access, APT groups often use a variety of advanced attack methods, including social engineering techniques.
- To maintain access to the targeted network without being discovered, threat actors continuously rewrite malicious code to avoid detection and other sophisticated evasion techniques.
- Common techniques used during APT attacks include the following:
- **Spear phishing.** APT actors commonly use highly targeted spear phishing emails to fool people into divulging personal information or clicking on harmful links that can execute malicious code into their systems. These emails are skillfully written to appear authentic and tailored to the recipient.
- **Zero-day exploits.** APT actors often take advantage of zero-day vulnerabilities in software or hardware that have recently been discovered but not yet patched. By exploiting the vulnerabilities before they've been addressed, threat actors can easily gain unauthorized access to target systems.

APT

- **Watering hole attacks.** APT actors use the watering hole attack to breach websites often accessed by their specific targets. By injecting malicious code into these websites, they can infect the systems of unsuspecting visitors.
- **Supply chain attacks.** Supply chain attacks target a specific organization's supply chain, compromising software or hardware before it reaches the intended receiver. This lets APT actors gain access to the victim's network.
- **Credential theft:** APT actors use methods such as keylogging, password cracking and credential phishing to obtain login credentials. Once they have legitimate credentials, they can navigate the network laterally and gain access to sensitive information.
- **Command-and-control (C&C) servers:** Using C&C servers, APTs create communication routes between hacked systems and their network. This lets the attacker maintain control over the compromised network and exfiltrate data.
- **Evasion strategies:** To avoid being discovered by security systems, APT attackers often hide their operations using legitimate tools and processes, code obfuscation and anti-analysis measures.

APT Cycle


Advanced Persistent Threat (APT) Lifecycle
1 Define target
2 Find and organize accomplices
3 Build or acquire tools
4 Research target
5 Test for detection
6 Deployment
7 Initial intrusion
8 Outbound connection initiated
9 Expand access and obtain credentials
10 Strengthen foothold
11 Exfiltrate data
12 Cover tracks and remain undetected

APT Example

- **Gelsemium** targeted a Southeast Asian government for six months between 2022 and 2023. The cyber espionage group responsible has been operational since 2014. They initially exploited their target by installing web shells to perform basic reconnaissance.
- **APT41** targeted the proprietary information of technology and manufacturing companies via malware, including digitally signed kernel-level rootkits. The Chinese state-linked group, also known as Winnti, targeted companies in East Asia, Western Europe and North America from at least 2019 to 2021.
- **The Stuxnet worm** used to attack Iran's nuclear program was detected by cybersecurity researchers in 2010. Although Stuxnet isn't considered a cybersecurity threat today, it's still considered to be one of the most sophisticated pieces of malware ever detected. The malware targeted SCADA (supervisory control and data acquisition) systems and was spread with infected USB devices. The U.S. and Israel have both been linked to the development of Stuxnet. While neither nation has officially acknowledged its role in developing it, there have been unofficial confirmations that they were responsible for Stuxnet.

Malware Detection

- Signature-based detection uses a unique signature, or digital footprint, from software programs running on a protected system. Antivirus programs scan software, identify the signature then compare it to signatures of known malware.
- Antivirus programs have a vast library of known malware signatures which is updated on a regular basis. When an antivirus program pinpoints software that matches a known signature, it will either delete or quarantine it.

Malware File Signature

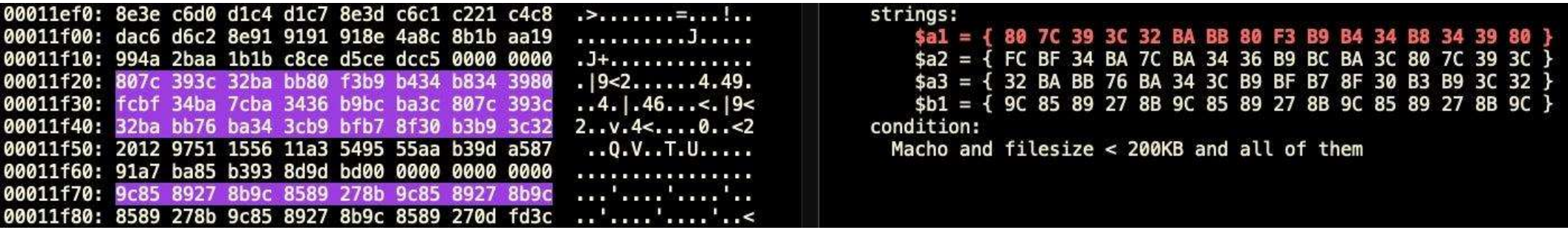
- In order to create a signature for a particular malware file or family of files, a security analyst needs one or more (the more the better) samples of the file to work from.
- Such samples may be gathered 'in the wild' from infected computers, sourced from the darknet and other places malware authors trade their work, or from shared malware repositories where security researchers (and in some cases the public) can share known malware files.
- Some popular malware repositories available to security professionals include **VirusTotal, Malpedia** and **MalShare.**

Malware File Signature

- Once a vendor has a set or ‘corpus’ of files to work with, they begin to examine the files for common characteristics.
- These characteristics can involve factors such as file size, imported or exported functions, data bytes at certain positions (‘offsets’), sectional or whole-file hashes, printable strings and more.
- The process of generating **signatures** can be automated, but it is often initially done manually by specialist malware analysts and reverse engineers, particularly when an entirely new family of malware is found.
- there are many different formats for creating signatures, one of the most popular formats widely in use today is YARA, which allows malware analysts to create signatures based on textual and binary patterns

Example

- For example, the following image shows a slice of code from a well-known malware family distributed by APT threat actor OceanLotus on the left, and a YARA signature to detect it on the right.



Example

- The signature condition, which states that the file must be of type ‘Macho’ (Mach-O), and have a file size of less than 200KB, while also containing all the strings defined in the rule.
- In the YARA format, the strings may occur as regular human-readable characters set between quotation marks, or
- As in the example – as hexademical-encoded bytes set between curly brackets. Some signature writers exclusively use the latter, even when the string to be matched is a string of human readable characters.
- Thus, ‘hello, world’ might be encoded in the signature as { 68 65 6c 6c 6f 2c 20 77 6f 72 6c 64 }.

Malware File Signature

- Vendors like SentinelOne realized from the outset that signature-based detection was insufficient to protect endpoints not only from commodity malware but also from targeted attacks.
- Rather than relying on file characteristics to detect malware, SentinelOne developed machine learning algorithms and behavioral AI that examine what a file does or will do upon execution.
- Such an approach solves the most serious drawbacks associated with signature detection. To begin with, harnessing the power of computer processors and machine learning algorithms takes the burden off analysts having to write individual signatures for new malware families.

Benefits

- Signature-based malware detection uses signatures and the best way to describe them is like the ‘fingerprint’ of a virus which is unique to that specific virus. This makes signature-based malware detection accurate in identifying known threats, as it matches the threat with its known code.
- Signature-based malware detection is a very effective technique used against known and frequent attacks, such as phishing, malware, or denial-of-service. It is also very easy to install and maintain, as it relies on regular updates of the signature database from security experts or vendors.