v2

Threat Hunting Professional

Threat Hunting Terminology

Section 01 | Module 02

Table of Contents

MODULE 02 | THREAT HUNTING TERMINOLOGY

2.1 Threat Hunting Terms

2.2 Threat Hunter Mindset: Threat Intelligence

2.3 Threat Hunter Mindset: Digital Forensics

2.4 Threat Hunting Simulations

Learning Objectives

By the end of this module, you should have a better understanding of:

- ✓ Common Threat Hunting Terms
- ✓ The Cyber Kill Chain
- ✓ Different Threat Hunting Mindsets
- ✓ Importance of continuous training

2.1

Threat Hunting Terms

2.1.1 Advanced Persistent Threat

The first term we'll discuss in this module is **APT,** which stands for **Advanced Persistent Threat**; this is a word you are probably familiar with, even if you're just getting into threat hunting.

APTs are ***groups*** or ***nation-states*** that have a significant amount of resources and infrastructure to conduct their malicious activities. Their targets are in various industries, such as governments, health care systems, and defense systems.

2.1.1 Advanced Persistent Threat

Despite what you might read or see in the headlines, not all APT groups attack US-based networks. An example of this would be ***Stuxnet***.

Stuxnet was a cyberweapon – malicious software targeting ***Iran's*** nuclear program. It was designed to target Siemens Step7 software on computers controlling a PLC (programmable logic controller).

2.1.1 Advanced Persistent Threat

There are a couple of things you should be aware of regarding the word **APT**. Just because a group or nation-state is labeled as an APT group, it does not mean that their techniques are advanced, but they are still considered a persistent threat.

How can they still be considered a persistent threat even though they're not advanced? One answer. **Resources**.

2.1.1 Advanced Persistent Threat

As mentioned earlier, they could have a significant amount of money, manpower, etc., which will allow them to continually attempt to infiltrate a network for weeks, months, or even years.

Another important point is that APT groups are identified in various ways. One common naming convention is the word **APT** followed by a **number**, like **APT 1**.

2.1.1 Advanced Persistent Threat

Below is a small chart displaying some of the different names this particular group, APT 1, might be called.

APT 1	Comment Panda	PLA Unit 61398	TG-8223	Comment Crew

2.1.1 Advanced Persistent Threat

The way that the APT group will be referenced will depend on which vendor-specific APT report you're reading.

For example, **Mandiant** will refer to ***Comment Crew*** as ***APT 1,*** whereas **CrowdStrike** will refer to them as ***Comment Panda***.

2.1.1 Advanced Persistent Threat

APT 1 is a Chinese-based cyber espionage group, a ***nation-state***. It has been discovered that APT 1 is **the 2nd Bureau of the People's Liberation Army General Staff Department's 3rd Department**. You might see this particular military unit referred to as **The People's Liberation Army (PLA)** or, more specifically, as **PLA Unit 61398**.

You can read more about this APT group in a report published by Mandiant in 2013 titled "APT1 – Exposing One of China's Cyber Espionage Units".

https://www.fireeye.com/content/dam/fireeye-www/services/pdfs/mandiant-apt1-report.pdf

2.1.1 Advanced Persistent Threat

Florian Roth (@cyb3rops) put together a spreadsheet that lists APT groups and operations.

You can find the spreadsheet here.

2.1.2 Tactics, Techniques & Procedures

The next word we'll look at is **TTP**, which stands for ***Tactics, Techniques, and Procedures***. You might see references to TTPs as ***Tools, Techniques, and Procedures***.

This term, like many terms you'll see in cybersecurity, was taken from the military world. In short, **TTPs** represent the methods or signature of the adversary.

2.1.2 Tactics, Techniques & Procedures

Tactics are the employment and ordered arrangement of forces in relation to each other, which defines the adversary's tactical objective. It is the "why" behind the reason for performing an action.

Techniques are non-prescriptive ways or methods used to perform missions, functions, or tasks; this defines "how" the adversary achieves a tactical objective by performing an action.

2.1.2 Tactics, Techniques & Procedures

Procedures are standard, detailed steps that prescribe how to perform specific tasks. It is the actual implementation of each Technique.

Thus, TTPs tell us the methods the adversary uses to enter the network and how they pivot throughout the network to achieve their goals. TTPs will help us identify the adversary in future attacks by creating **Indicators of Compromise** (IOCs).

Click HERE to go back to Slide 43

2.1.2.1 TTPs - IOCs

IOCs are artifacts that were gathered from an active intrusion or previous intrusion that are used to identify a particular adversary. These artifacts include MD5 hashes, IP addresses, names of EXEs used, etc.

For example, we will look at APT 1 and list certain IOCs for APT 1.

2.1.2.1 TTPs - IOCs

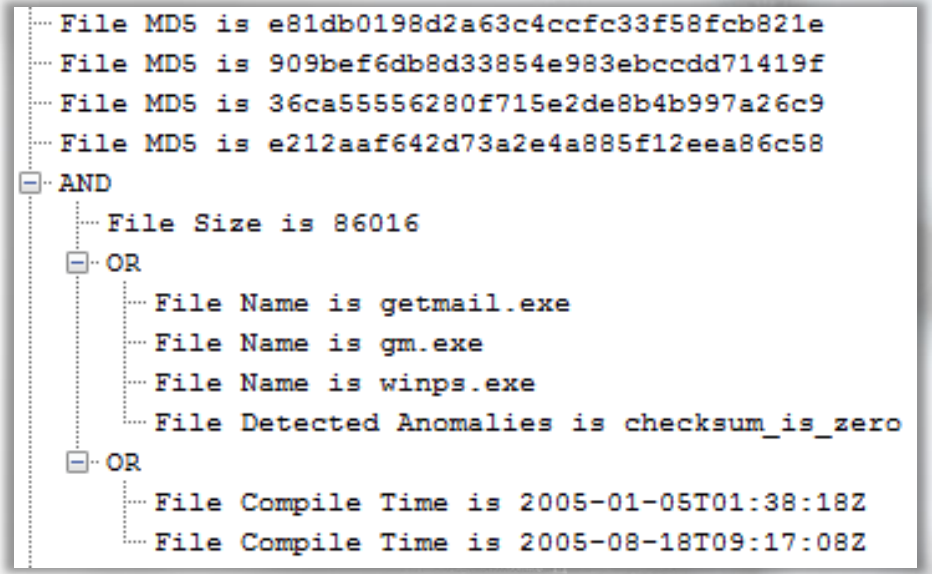
APT 1 uses two custom utilities to steal emails from their victims:

- **GETMAIL**: malware used to extract email messages and attachments from Outlook PST files.

- **MAPIGET**: malware used to extract email messages and attachments from an Exchange server.

2.1.2.1 TTPs - IOCs

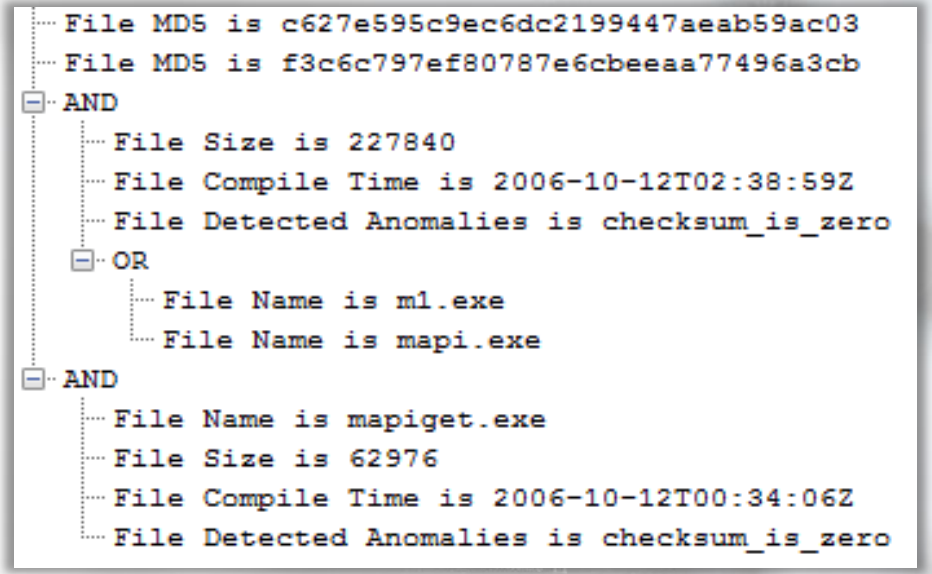
Here is a snippet of the IOC for GETMAIL.



2.1.2.1 TTPs - IOCs

This is a snippet of the IOC for MAPIGET.

We'll discuss IOCs and various IOC-based tools in later modules.



2.1.3 Pyramid of Pain

Now, let's look at the **Pyramid of Pain,** which is a visual that will layer the potential usefulness of indicators that will aid you in detecting an adversary.

It also measures how difficult it will be to obtain that particular indicator or indicators, as well as the impact on obtaining the intelligence on them.

2.1.3 Pyramid of Pain

The Pyramid of Pain was created by David Bianco (FireEye), and he discusses the Pyramid of Pain in a presentation titled ***Intel-Driven Detection and Response to Increase Your Adversary's Cost of Operations***.

The next few slides will outline the Pyramid of Pain and detail each layer of it.

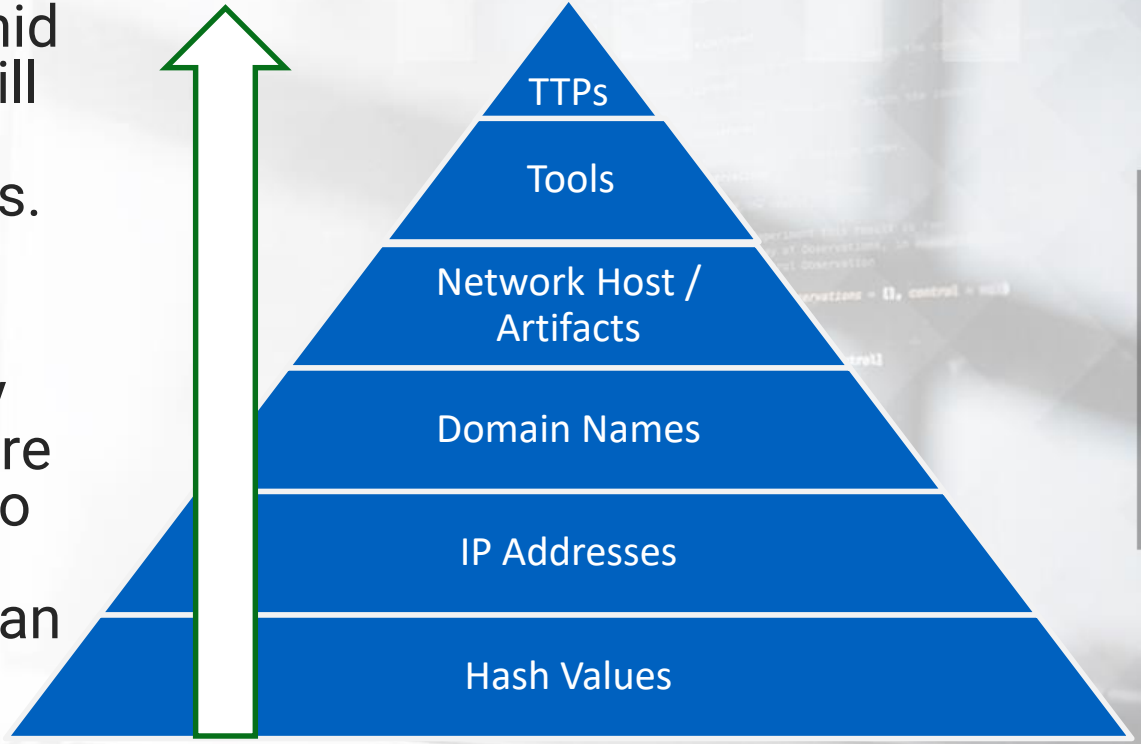
*Click HERE to go back to Slide 30
*Click HERE to go back to Slide 33

https://www.youtube.com/watch?v=zlAWbdSlhaQ

2.1.3 Pyramid of Pain

As we go up the Pyramid of Pain, the harder it will be to obtain the adversary specific IOCs.

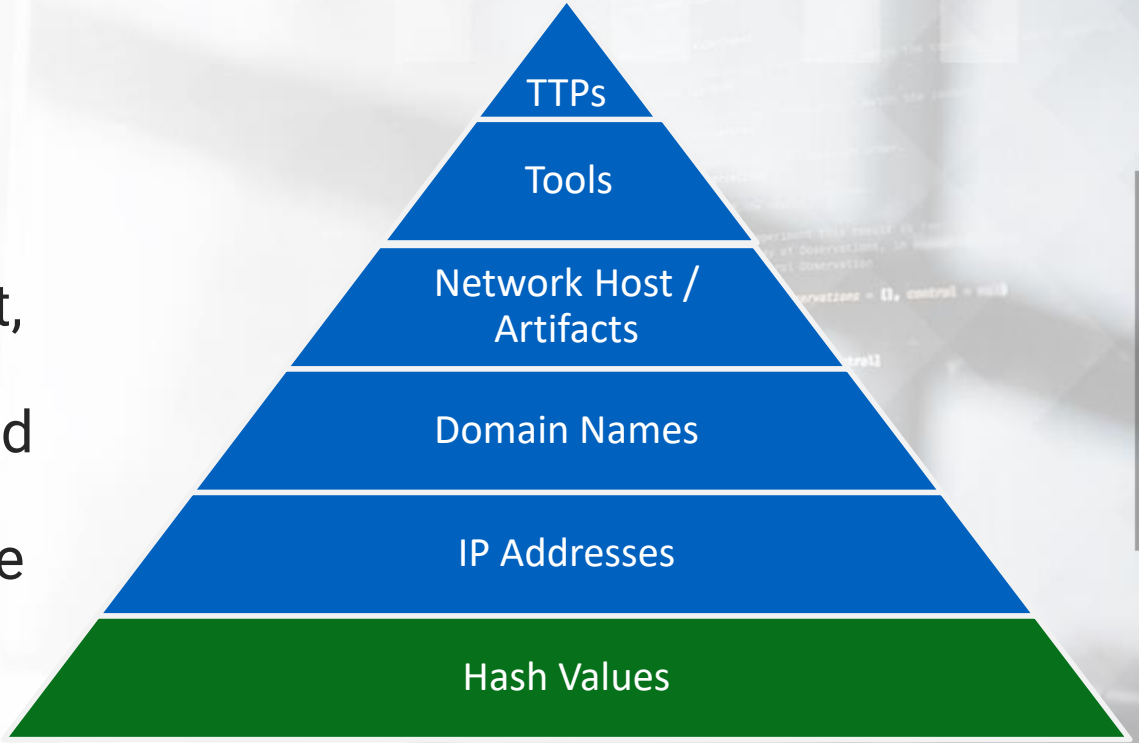
On the flip side, if we obtain those adversary specific IOCs, then we're forcing the adversary to change their attack methods, which is not an easy task for them.



2.1.3.1 Pyramid of Pain – Hash Values

Let's look at Hash Values. So, what is a **hash value**?

According to Microsoft, "a hash value is a numeric value of a fixed length that uniquely identifies data". We use these alphanumeric values as signatures.



2.1.3.1 Pyramid of Pain – Hash Values

You might have seen this before when you download a binary (EXE). The developer may display the hash value of the binary.

You use the hash value of the binary that was downloaded and compare it to the value on the developer's site; this will confirm the authenticity of the binary you downloaded and verify that it has not been tampered with.

2.1.3.1 Pyramid of Pain – Hash Values

The following snippet is from the Putty download page, where we can see the hash values of files for download to confirm what was downloaded is what the website is hosting.

Checksum files

Cryptographic checksums for all the above files

MD5:	md5sums	(or by FTP)	(signature)
SHA-1:	sha1sums	(or by FTP)	(signature)
SHA-256:	sha256sums	(or by FTP)	(signature)
SHA-512:	sha512sums	(or by FTP)	(signature)

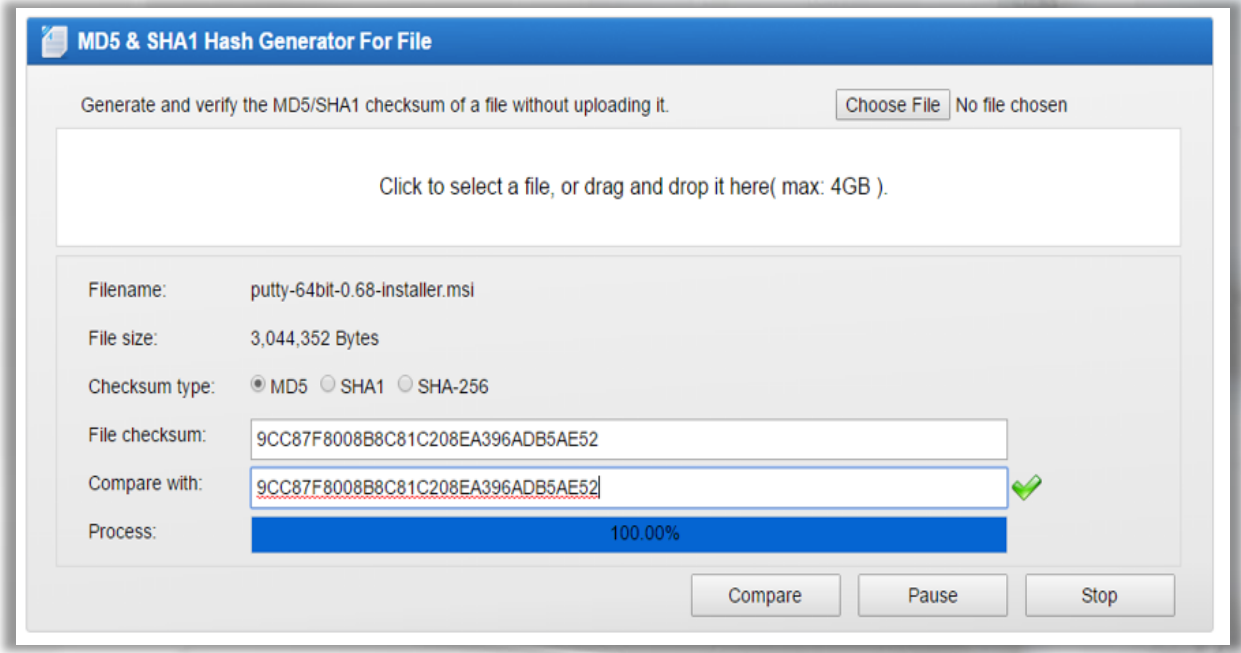
2.1.3.1 Pyramid of Pain – Hash Values

Here is a list of MD5 values, from the md5sum file we saw on the previous slide.

```
1c31b9d59c33124cf19aafe5ca4d8d77  w64/plink.exe
9206dae8b89a9e366b88f57a117068ea  w64/pageant.exe
be183d872773a130efb8bf1f1c60b6db  w64/puttytel.exe
caba0287018a2f1c0f4e7ba357f9072d  w64/puttygen.exe
5ca0a9e56499c658d2790be7113930f1  w64/putty.zip
8ca5e64d33ff45f0278de27aa4994434  w64/pscp.exe
9cc87f8008b8c81c208ea396adb5ae52  w64/putty-64bit-0.68-installer.msi
a04e72503528dfc132c48e95fa3160ad  w64/putty.exe
fc10492df39f9be3d8c139e2828a59da  w64/psftp.exe
```

2.1.3.1 Pyramid of Pain – Hash Values

This screenshot verifies the MSI that was downloaded is authentic based on the checksum (MD5) listed on the download page.



2.1.3.1 Pyramid of Pain – Hash Values

Why are MD5 hashes unreliable?

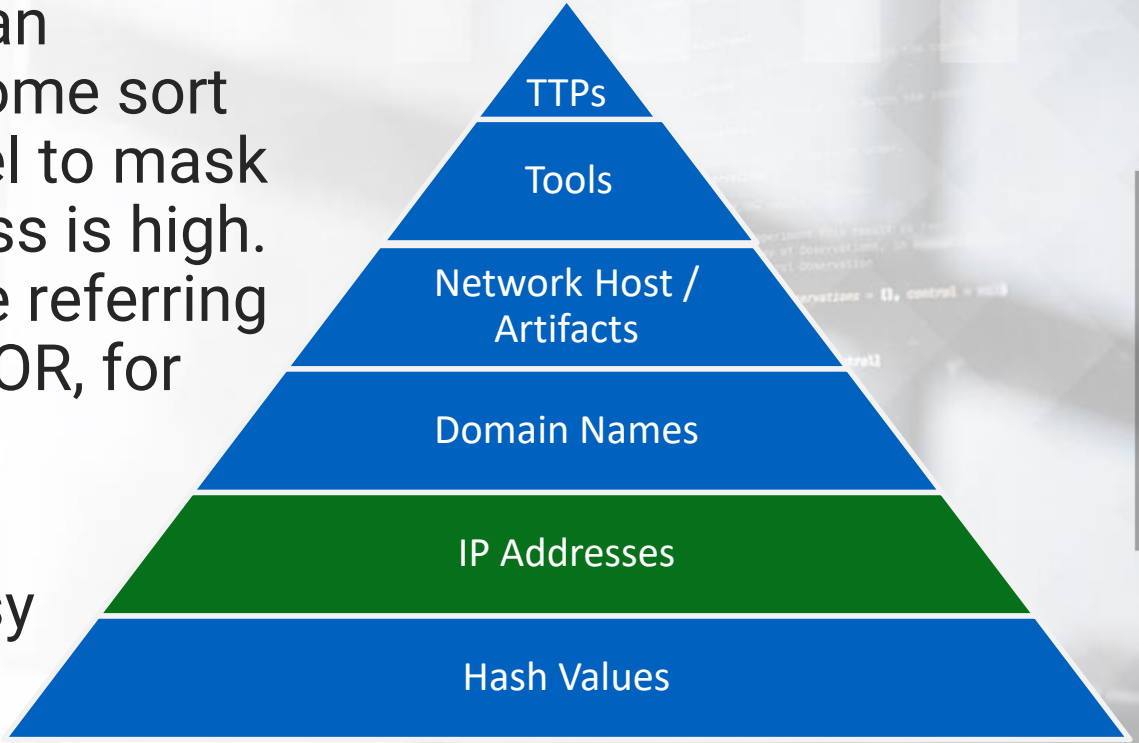
If you use it as the sole identifier for a binary, with no other IOCs, that MD5 value can change by a slight modification to the source code or by recompiling the source code with a different compiler.

In this case, your IOC is nearly useless as it's easy to change and has, therefore, no real impact on the adversary. So, Hash Values are good, but the least reliable compared to other indicators, because they're easy to change.

2.1.3.2 Pyramid of Pain – IP Addresses

The probability that an adversary is using some sort of anonymity channel to mask their actual IP address is high. By anonymity, we are referring to a proxy, VPN, or TOR, for example.

IP addresses are easy to change.



2.1.3.2 Pyramid of Pain – IP Addresses

In the snippet below, you will see examples of the representation of IP addresses (referenced in the presentation in slide 21).

Dotted Decimal 192.168.1.1 **Dotted Hex** 0xC0.0xA8.0x01.0x01 **Dotted Octal** 0300.0250.0001.0001	**Decimal** 3232235777 **Hex** 0xC0A80101 **Octal** 030052000401

2.1.3.2 Pyramid of Pain – IP Addresses

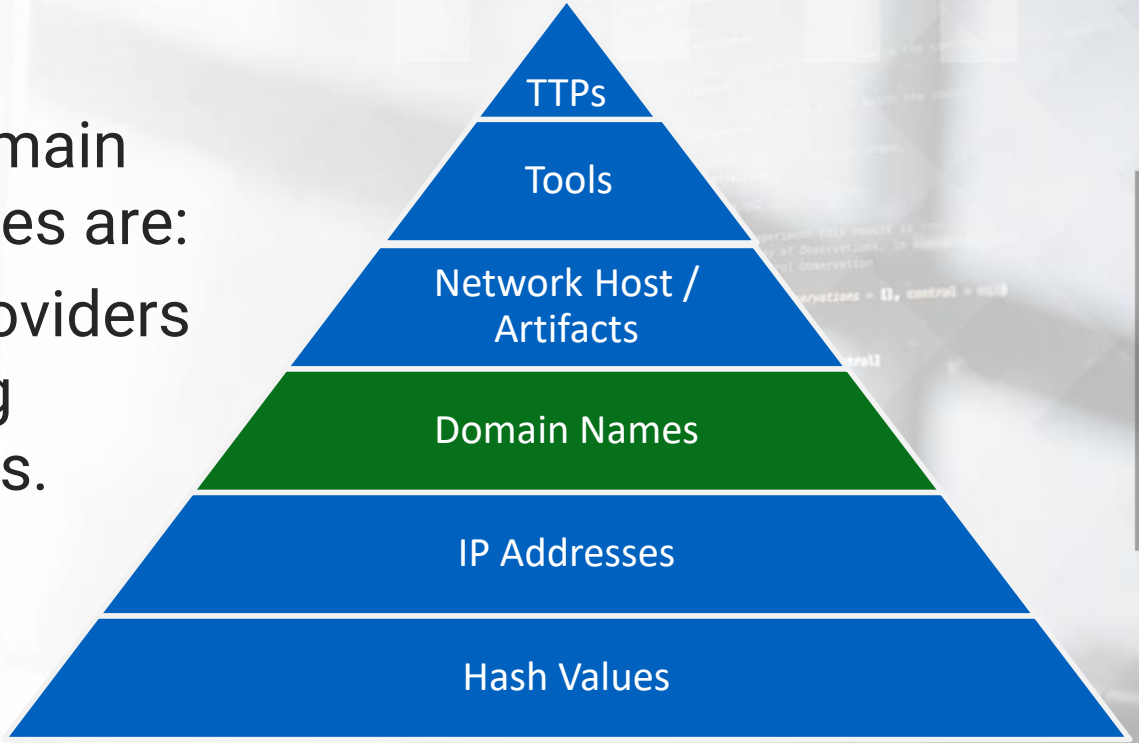
If the IP addresses are hardcoded, then these IPs can be blacklisted and prevented from making outbound communications; this will make it more difficult for the adversary because now the tools and scripts will have to point to a new IP addresses.

Again, this is the case if IP addresses are **hardcoded**.

2.1.3.3 Pyramid of Pain – Domain Names

Let's now look at Domain names. Domain names are:

- Dynamic DNS providers help the updating process with APIs.
- Easy to change.



2.1.3.3 Pyramid of Pain – Domain Names

Below is a snippet from the presentation, mentioned on slide 21, that has examples of domain names.

Unicode 邪悪なドメイン.com **Punycode** Xn—q9j5f9d1dzdq306auhtd.com	**Legitimate Domain** rvasec.com **Malicious Homograph** rvasec.com

*Click HERE to go back to Slide 36

2.1.3.3 Pyramid of Pain – Domain Names

In the chart illustrated on the previous slide, we can see that a domain name can be displayed or accessed in various fashions.

We will not discuss every type of format a domain name can be displayed as or accessed by. Instead, we'll discuss the lesser-known techniques to display a domain name.

2.1.3.3 Pyramid of Pain – Domain Names

What is Punycode?

From punycoder.com, **Punycode** is a special encoding used to convert Unicode characters to ASCII. Punycode is used to encode ***IDN***s (Internationalized Domain Names).

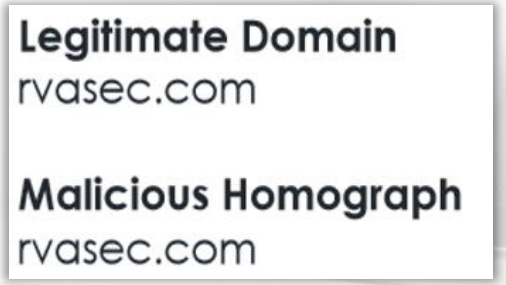
Below is an example of text in Unicode that is converted to Punycode.

Text	Punycode
Example: 點看	Example: xn--c1yn36f

2.1.3.3 Pyramid of Pain – Domain Names

Next, we'll look at **IDN Homograph Attacks**. In the snippet seen on slide 33, it's called a Malicious Homograph.

On the same slide, the domain listed under Legitimate Domain and Malicious Homograph look identical, but they are, in fact, different.



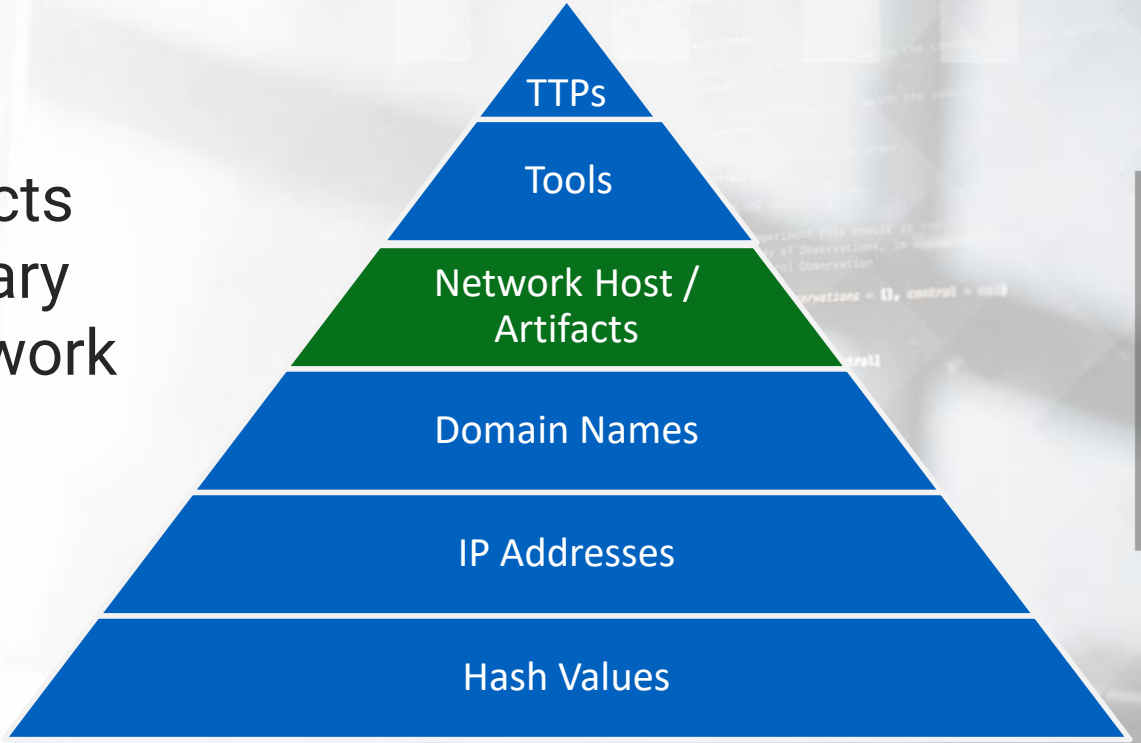
2.1.3.3 Pyramid of Pain – Domain Names

In an IDN Homograph Attack, malicious threat actors will exploit the fact that many different characters look alike; this is similar to another concept known as **typosquatting**.

Please reference the following Black Hat presentation, titled "Unraveling Unicode: A Bag of Tricks for Bug Hunting", for more information and additional references on the subject.

2.1.3.4 Pyramid of Pain – Network/Host Artifacts

Network/Host Artifacts are clues the adversary left for us within network packets and on the endpoint systems.



2.1.3.4 Pyramid of Pain – Network/Host Artifacts

Below is an example of a Network Artifact and a Host Artifact:

Network Artifacts	Host Artifacts
Rare User-Agent strings	Specific Registry key
Traffic on non-traditional ports (i.e. 6667)	Process connected on port 80 that is not a browser

2.1.3.4 Pyramid of Pain – Network/Host Artifacts

Here we see an example of a network artifact, a fake user-agent.

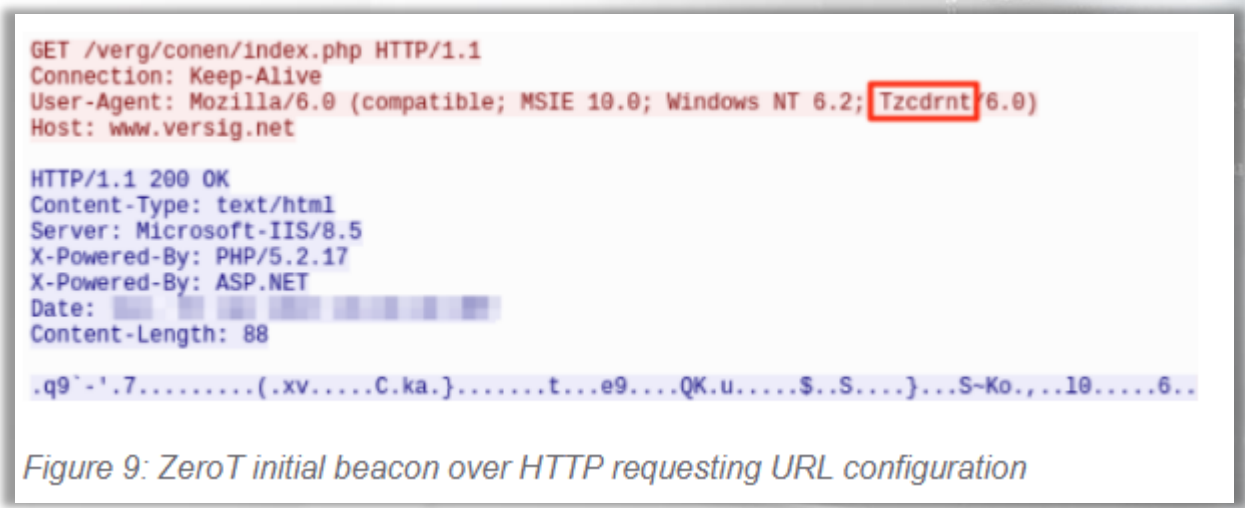
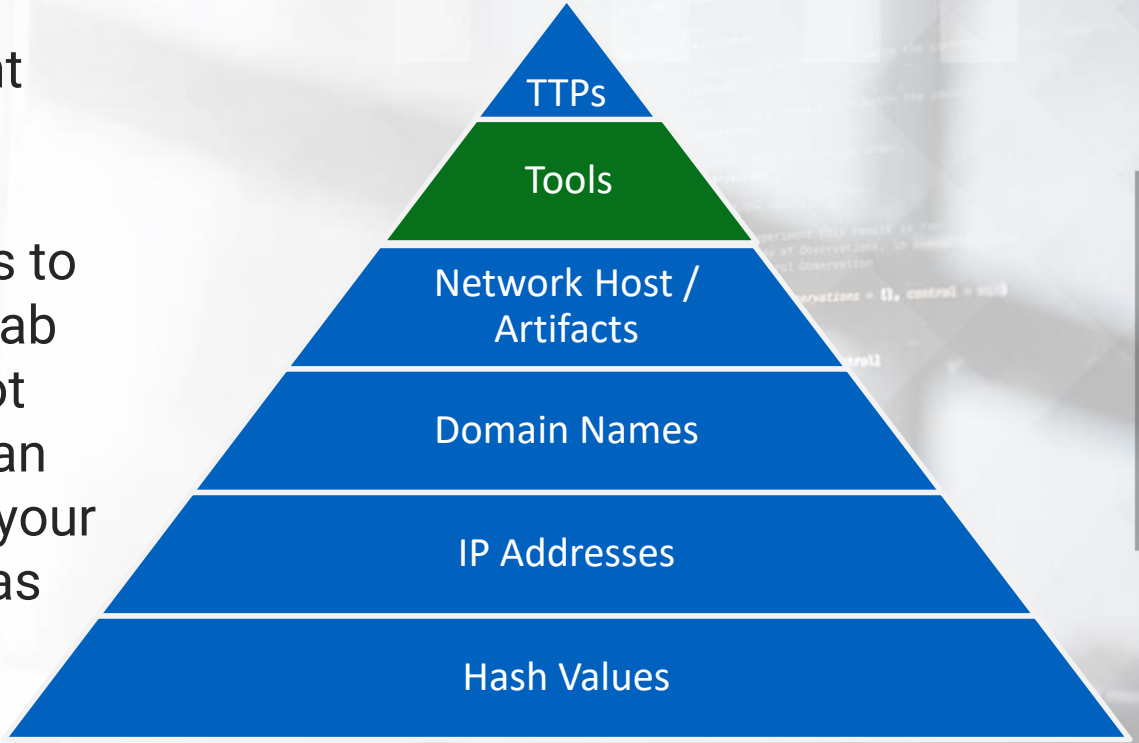


Figure 9: ZeroT initial beacon over HTTP requesting URL configuration

Source: https://www.proofpoint.com/us/threat-insight/post/APT-targets-russia-belarus-zerot-plugx

2.1.3.5 Pyramid of Pain – Tools

An APT group will most likely stick to a consistent set of tools. If you're an experienced penetration tester, then you know this to be true. You won't just grab a tool you normally do not use if you're conducting an SQL attack. You will use your tool of preference, such as SQLMap, or something similar.



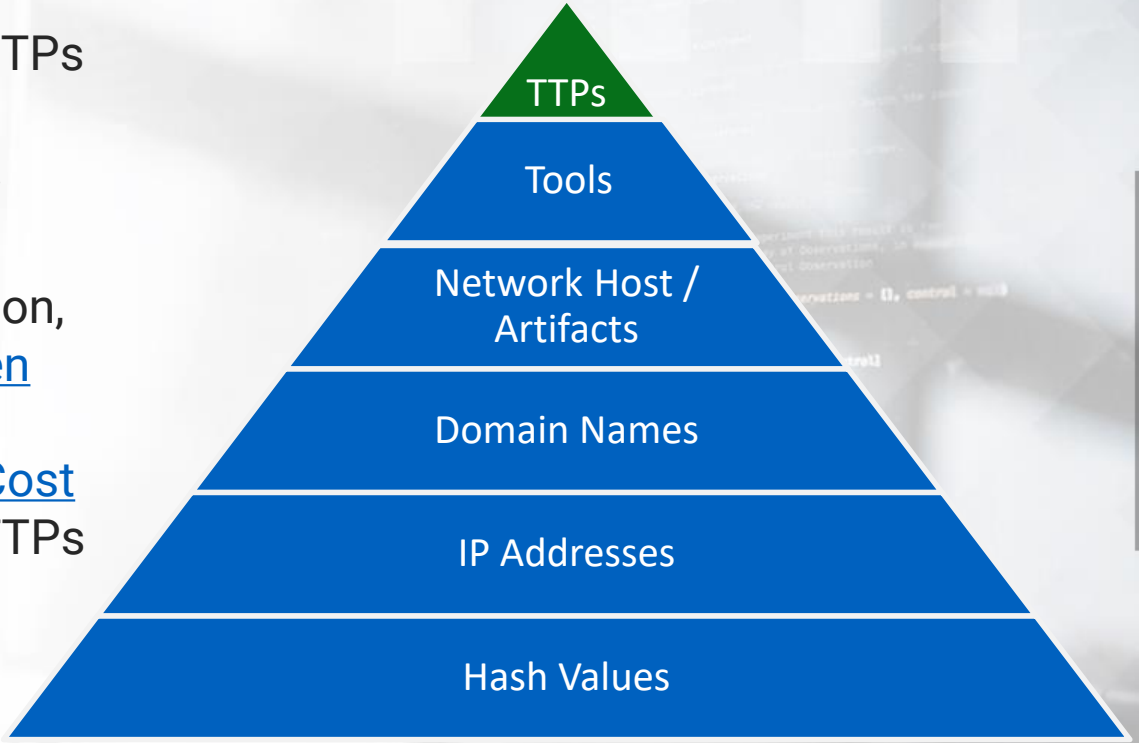
2.1.3.5 Pyramid of Pain – Tools

If you get good at detecting a particular tool, this will force the adversary to use a new tool. This is because the tool they currently use won't work against your detection capabilities anymore, which will lead to more work on behalf of the adversary.

2.1.3.6 Pyramid of Pain – TTPs

If you recall from slide 15, TTPs represent the methods or signatures of the adversary.

In David Bianco's presentation, "Pyramid of Pain: Intel-Driven Detection and Response to Increase Your Adversary's Cost of Operations", he defines TTPs as the **expression of the attacker's training**.



https://www.youtube.com/watch?v=zlAWbdSlhaQ

2.1.3.6 Pyramid of Pain – TTPs

Retraining is hard and expensive. Imagine doing so for 1,000 operators so that the current TTPs and IOCs gathered on them no longer prove to be fruitful. That task is easier said than done, but if they have the funding, it is not impossible.

2.1.4 Cyber Kill Chain Model

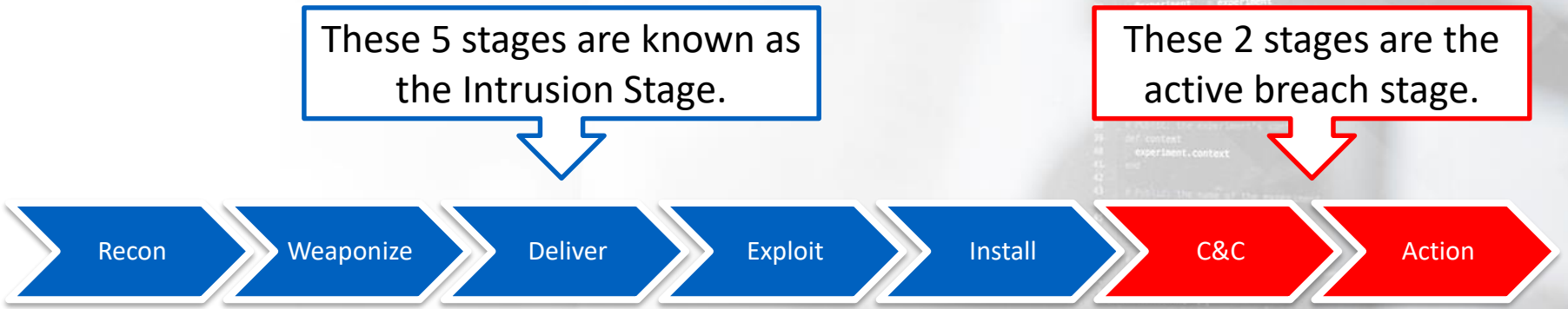
Let's now turn our attention to the **Cyber Kill Chain**; this term also stems from the military. The military term is kill chain. Kill chain, in both cases, refers to the different stages of an attack.

Lockheed Martin is credited for applying this term to information security.

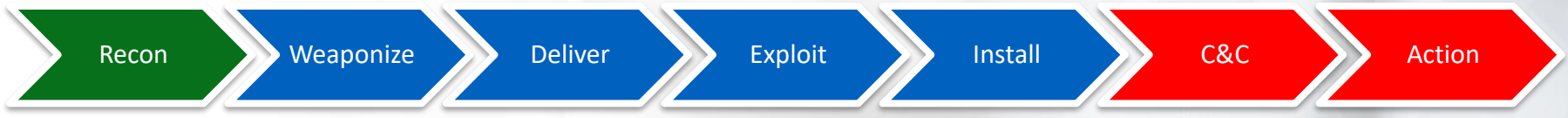
https://www.lockheedmartin.com/us/what-we-do/aerospace-defense/cyber/cyber-kill-chain.html

2.1.4 Cyber Kill Chain Model

Let’s see an example of the Cyber Kill Chain Model through a sample attack scenario.



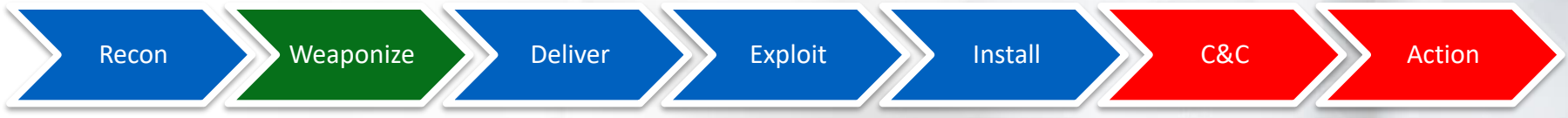
2.1.4 Cyber Kill Chain Model



The **Recon** step involves passive scanning plus Open-Source Intelligence, also known as OSINT (i.e., social media, search engines, etc.).

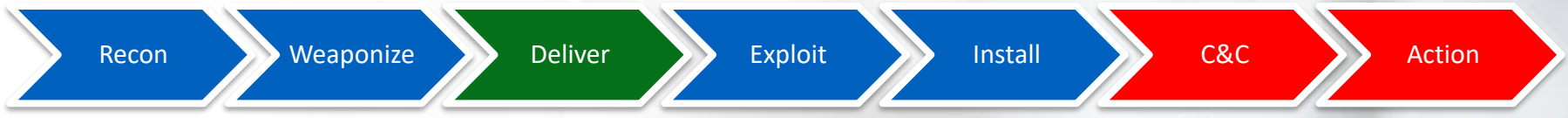
It can also involve active scanning of public-facing IPs.

2.1.4 Cyber Kill Chain Model



Weaponize: This is where the RAT (Remote Access Tool) is added to the exploit. The exploit can reside on a web page or a malicious macro-based document attached to an email. In this stage, the adversary also considers the method of delivery.

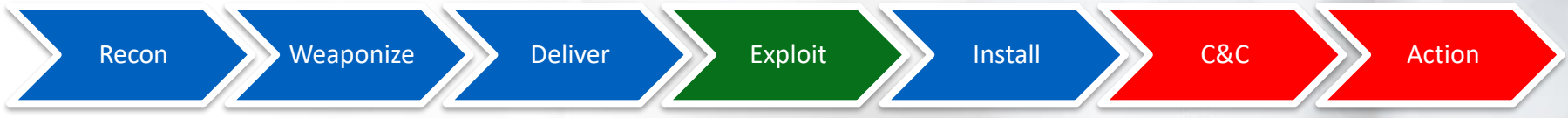
2.1.4 Cyber Kill Chain Model



The **Deliver** phase covers the delivery of the weaponized tool.

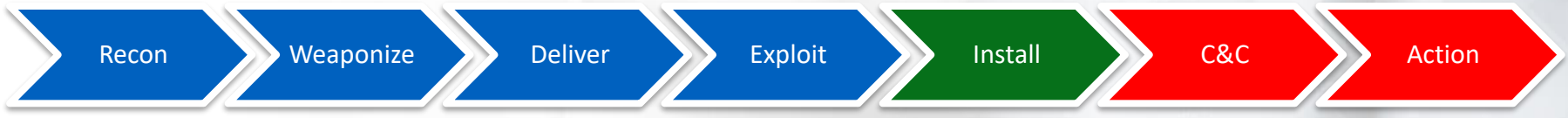
There are a few methods for delivery, including via email, social media, or a watering hole attack.

2.1.4 Cyber Kill Chain Model



The **Exploit** phase is the actual exploitation, and this is when a user opens the document attached to an email, clicks a link, etc.; this can be a 2-step process where a loader is used to download the actual RAT. The loader will typically be small in size and reside only in memory.

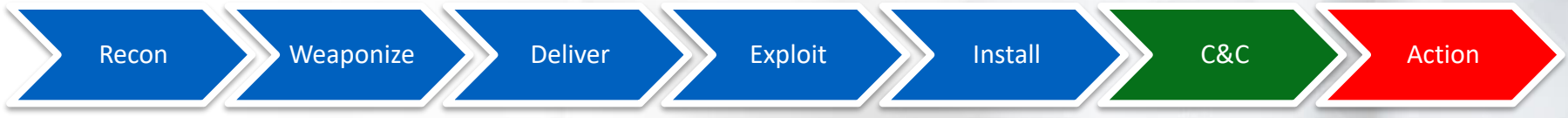
2.1.4 Cyber Kill Chain Model



Install: At this point, in most cases, additional tools are installed via the RAT.

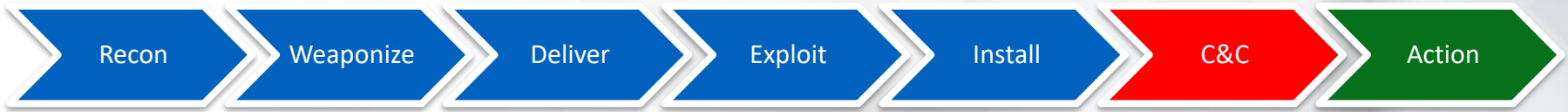
Other tools can be a network scanner, a keylogger, etc.

2.1.4 Cyber Kill Chain Model



C&C is the **command & control (C2) phase**. This is when the victim's machine will call out to an IP or domain and provide the adversary command-line remote access to the compromised machine.

2.1.4 Cyber Kill Chain Model



GAME OVER!

Action: This is where the goal is achieved. The goal can be exfiltration. This is when:

- The adversary scans the network, looks for/reviews data, and grabs what they are looking for.
- What you're trying to protect leaves the network.

2.1.4 Cyber Kill Chain Model

2 things to remember about the Cyber Kill Chain is that it is a cyclical process, not linear. What that means is that once an adversary gets a foothold on a box (machine), they will not stay there.

- They will begin from the start of the kill chain.
- They will perform internal reconnaissance and look for other machines to exploit.
- They will also look to cover their tracks.

Most likely, the box they'll establish the C2 channel with will not be the initial box they exploited.

2.1.4 Cyber Kill Chain Model

Our goal as defenders is to stop the adversary from progressing up the kill chain. Doing this in one of the ***early stages*** of the chain is always preferred.

Our goal as hunters is to detect the adversary before their objective is achieved, not just at code execution.

2.1.4 Cyber Kill Chain Model

Companies such as FireEye and Mitre have developed their own model for the Cyber Kill Chain. You will see references to these models as “**Attack Lifecycle**”.

Learn more about their attack models by doing a quick search online.

2.1.5 The Diamond Model

The last model we'll look at is called the **Diamond Model**. The paper describing the Diamond Model was released in 2013 by ***The Center for Cyber Intelligence Analysis and Threat Research***.

The link to the official paper is here.

*Click HERE to return to slide 60

2.1.5 The Diamond Model

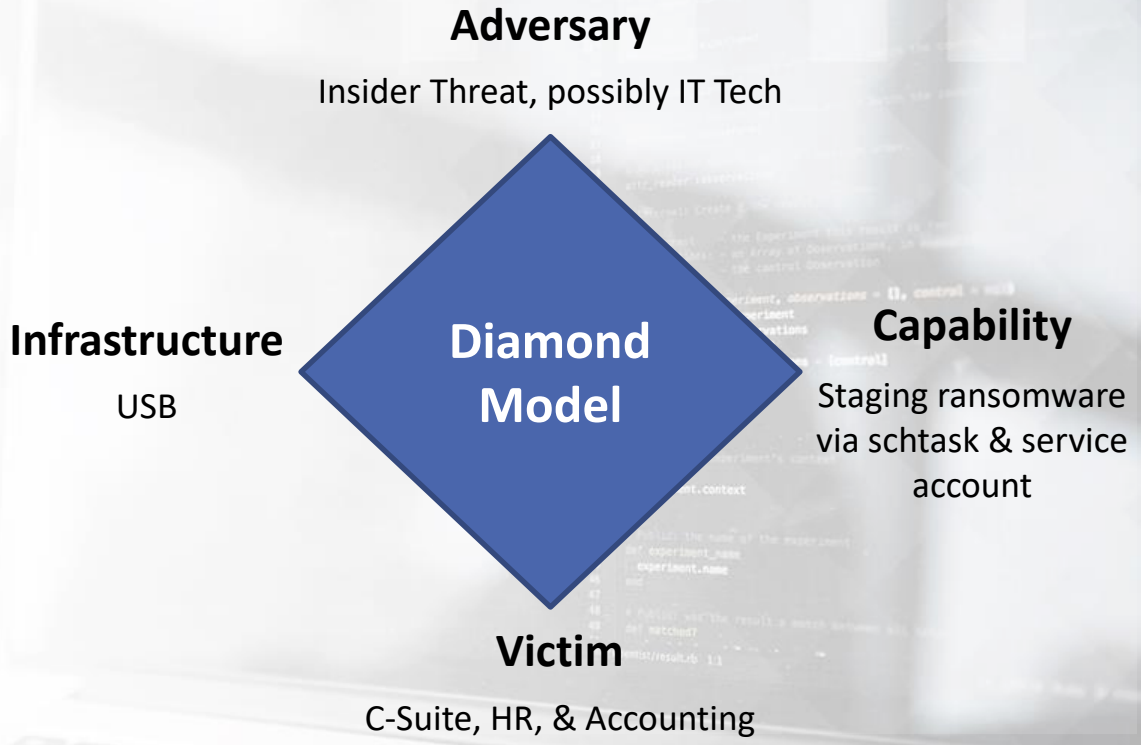
What is the Diamond Model?

"In its simplest form, the model describes that an ***adversary*** deploys a ***capability*** over some ***infrastructure*** against a ***victim***."*

https://apps.dtic.mil/docs/citations/ADA586960

2.1.5 The Diamond Model

Here is a visual depiction of the Diamond Model.



2.1.5 The Diamond Model

In the same paper referenced on Slide 57 under Diamond Event - Axiom 1, it states: “for every intrusion event there exists an adversary taking a step towards an intended goal by using a capability over infrastructure against a victim to produce a result.”

The Diamond Model can be used in conjunction with the Cyber Kill Chain model. Remember, the goal is to prevent the adversary from reaching his/her goal.

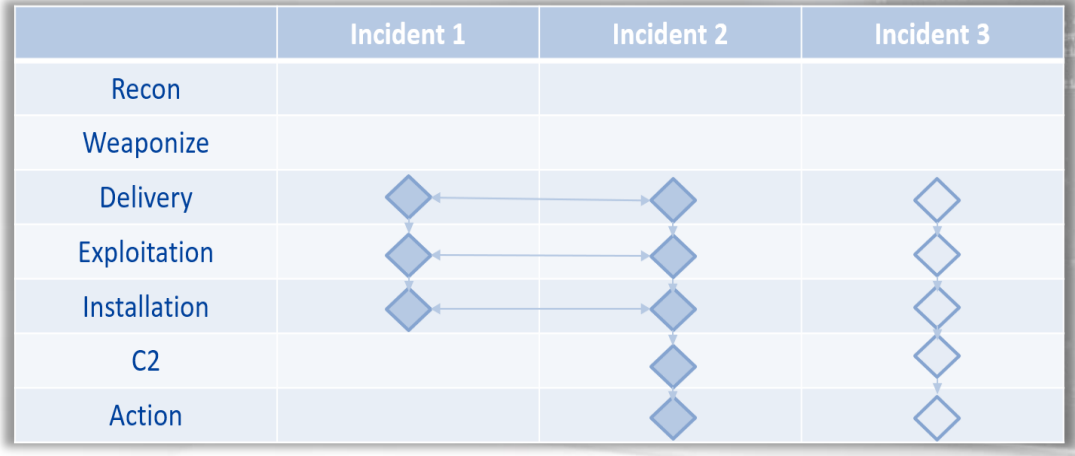
https://apps.dtic.mil/docs/citations/ADA586960

2.1.5 The Diamond Model

Axiom 4 states that “every malicious activity contains two or more phases which must be successfully executed in succession to achieve the desired result.”

2.1.5 The Diamond Model

This image shows the conjunction between the Cyber Kill Chain and the Diamond Model, and also illustrates the information gathered from 3 incidents.



2.1.5 The Diamond Model

In each stage of the Kill Chain, the Diamond Model is used to collect data on the attack. In Incident 2, based on similarities of Incident 1, the hypothesis would be that it's the same adversary. Incident 3 shows no correlation between Incident 1 or 2, so it leads us to believe that it's a different adversary.

Should these two adversaries strike again, you have information, as well as indicators that have been created to assist you in detecting and stopping them.

2.1.5 The Diamond Model

Remember to find a methodology and model that works for you.

Everything within cybersecurity should follow some methodology.

Threat Hunting Mindset: Threat Intelligence

2.2 Threat Hunting Mindset: Threat Intelligence

In most cases, a threat hunter has one of two mindsets. One hunter will rely mostly on indicator-based detection through **threat intelligence,** while the other will rely mostly on technique or anomaly-based detection through **digital forensics**.

Let’s look at threat intelligence first.

2.2 Threat Hunting Mindset: Threat Intelligence

What is threat intelligence?

A simple definition of **Threat Intelligence** is that it is data on threats. The information will come in various forms and could be obtained through multiple channels, such as open-source, social media, vendor reports, etc.

2.2 Threat Hunting Mindset: Threat Intelligence

The data can be IP addresses, netblocks, domains, MD5 hashes, etc. The threats can be APTs, cybercrime groups, hacktivists, etc. Here, we focus on alerting based on the specific use of those identified bad tools or resources.

Data is exactly that, just data. For the information to become intelligence, it has to be analyzed. Once it's analyzed and it becomes actionable, then it's categorized as intelligence, because there is context around the information. Some data might not apply to your organization.

2.2.1 The 3 Types of Threat Intelligence

An appliance will be used to sift through all of the data so you can focus on what you need to.

Threat Intelligence can be divided into 3 types:

1. Strategic: Who, Why, and Where
2. Tactical: What and When
3. Operational: How

2.2.1.1 The 3 Types of Threat Intelligence - Strategic

1. Strategic: Who, Why, & Where

Strategic Intelligence is designed to assist senior management in making informed decisions about the security budget and security strategies (such as risk management).

With specific intelligence, senior management ***might*** obtain answers to the following questions.

2.2.1.1 The 3 Types of Threat Intelligence - Strategic

Who is the adversary?

Why are they targeting you?

Where have they attacked prior to attacking you?

Note: Sometimes, it's not easy to provide answers to those questions.

2.2.1.2 The 3 Types of Threat Intelligence - Tactical

2. Tactical: What & When

Tactical Intelligence, which merges into Operational Intelligence, deals with the adversary's TTPs; this is where the Cyber Kill Chain and Diamond Models are used to attempt to identify the adversary's pattern of attacks, also known as their signature.

2.2.1.2 The 3 Types of Threat Intelligence - Tactical

As stated earlier, Tactical Intelligence addresses the what and when.

- **What** is the adversary's toolset?
- **When** are these attacks orchestrated?

2.2.1.3 The 3 Types of Threat Intelligence - Operational

3. Operational: How

Operational Intelligence deals with the actual indicators, the IOCs, and it addresses the how.

How is the adversary conducting their attack?

2.2.1.3 The 3 Types of Threat Intelligence - Operational

Remember that Operational Intelligence can merge into Tactical Intelligence. In most cases, you will see it identified as Operational Intelligence.

ISACs are one of several avenues to assist with obtaining this subset of intelligence. **ISACs** are ***Information Sharing and Analysis Centers***. We will look into this further in the next module.

2.2 Threat Hunting Mindset: Threat Intelligence

As hunters, we are more focused on tactical and operational intelligence, how the adversary does what they do, so we can detect it and prevent further escalation through the attack chain.

In summary, this type of hunter will be focused on ***known bad*** information, data that will assist him/her in the hunt.

Threat Hunting Mindset: Digital Forensics

2.3 Threat Hunting Mindset: Digital Forensics

Now we'll look at the other type of hunter; this hunter will focus primarily on the host, network, and memory forensics in his/her hunt when hunting for the ***unknown***.

The data sources may be:

- Network, VPN, and Firewall logs
- Disk/Share Access
- Disk Forensic artifacts and advanced system logging
- Memory Forensic artifacts
- Reputation based intelligence
- Passive DNS

2.3 Threat Hunting Mindset: Digital Forensics

They will still use threat intelligence, it would be foolish not to, but this type of hunter will not solely rely on that. This hunter will take it a step further and analyze digital artifacts to see if there is any indication of a threat.

Here, we don’t wait for an alert from one of the appliances regarding a potential threat. We are proactively hunting!

This is human-based detection.

2.3 Threat Hunting Mindset: Digital Forensics

While the goal of hunting is to transform successful hunts into automated detection, the outcome of it may be an initial observation of a threat, which starts a forensic investigation.

2.3 Threat Hunting Mindset: Digital Forensics

Key components that define a good hunter are:

- Knowledge of the available data sources/logs
- Understanding a broad variety of attacks
- Understanding what attacks can be detected in the different data sources/logs
- Ability to locate reliable sources with details about new attack techniques

A great understanding of how attacks work is key to successful hunts.

2.3 Threat Hunting Mindset: Digital Forensics

You should also strive to identify variations of a specific attack, not just that one example defined in sources on attack techniques.

Think about process masquerading – if a certain attack appears to be running as svchost.exe (except that it is from an odd location), your hunt should aim to expand the detection on other processes that may be victims of this type of attack.

2.3 Threat Hunting Mindset: Digital Forensics

With the available data sources, we can choose to **perform hunts in 2 distinctive ways**:

1. Attack based hunting
2. Analytics-based hunting

2.3.1 Attack Based Hunting

In attack based hunting, we search for evidence whether or not a specific attack has occurred in the environment. We are defining it by asking:

“Did …….. happen in my network”?

2.3.1 Attack Based Hunting

Examples:

- Did pass the hash happen in my network?
- Did anyone create a local account on a domain machine?
- Did credential theft happen in my network?

2.3.2 Analytics Based Hunting

In analytics-based hunting, we look at a set of data and try to see if anything stands out. It is, therefore, crucial to know what is normal. We are defining it by asking:

“Does anything in … data look malicious”?

2.3.2 Analytics Based Hunting

Examples:

- Unexpected encryption detected in network traffic
- A receptionist attempting to access HR data
- A workstation connecting to 10 others, using the same user account
- HR user performing LDAP queries or running “net” command
- Multiple wininit.exe running on a host

2.3.3 Hunting Periods

The data utilized in either of the hunting methods can be split into **3 distinctive hunting periods**:

1. Point in Time
2. Real-Time
3. Historic

Let’s look at these periods.

2.3.3.1 Point in Time

Point in Time only detects what is happening on a system at a point in time. It does not identify the activity that occurred before or after that point in time.

It is easy to perform, as no additional tool installation is required; however, there is a high likelihood of missing short-lived volatile data.

2.3.3.2 Real Time

Real Time detects activity that is occurring in real time. The data collection agent is required to be installed, and the collected data is sent to SIEM. A custom configuration for the collected data is recommended.

2.3.3.3 Historic

Historic utilizes logs to identify activities that occurred in the past.

Logging must be configured ahead of time.

2.3.4 Reverse Engineering Binaries

Finally, the hunter might also reverse engineer binaries, to see if the binary is legitimate or malicious.

Not all hunters have this ability, but in smaller organizations where a hunter is expected to wear more than one hat, this might be the case.

Threat Hunting Simulations

2.4 Threat Hunting Simulations

Threat Hunting is a broad topic that requires many skills that you can acquire in separate courses. The goal of this course is to provide you with mindset, methodologies, and practical skills to perform a hunt, and help you take it a step further to hunt for the unknown.

One point we felt that shouldn't be overlooked is **Threat Hunting Simulations**. The concept behind this is for the hunter to always practice and train, so that they can hunt effectively.

2.4 Threat Hunting Simulations

Think about soldiers.

Once they pass boot camp and are trained, they don’t end training forever. They are constantly training to ensure their skills don’t get rusty and that they don’t forget their training.

2.4 Threat Hunting Simulations

Penetration Testers exercise this practice as well. It is often achieved through Capture the Flag activities. These platforms are used as competitions, but some are used to train and enhance one's skill.

In summary, the threat landscape is constantly evolving. As hunters, we have to stay current and not get rusty at hunting.

Conclusion

This concludes this module on Threat Hunting Terminology. We have covered:

- ✓ Various terms associated with Threat Hunting
- ✓ Various attack models and methodologies
- ✓ The two threat hunter mindsets: intel and forensics
- ✓ Attack and Analytics based hunting
- ✓ The importance of continual training

References

References

[FireEye Cyber Threat Map]

https://www.fireeye.com/cyber-map/threat-map.html

[APT1 – Exposing One of China's Cyber Espionage Units]

https://www.fireeye.com/content/dam/fireeye-www/services/pdfs/mandiant-apt1-report.pdf

[APT Groups and Operations]

https://goo.gl/QEayyo

[RVAs3c: David Bianco: Pyramid of Pain: Intel-Driven Detection/Response to Increase Adversary's Cost]

https://www.youtube.com/watch?v=zlAWbdSlhaQ

References

Unraveling Unicode: A Bag of Tricks for Bug Hunting

http://www.blackhat.com/presentations/bh-usa-09/WEBER/BHUSA09-Weber-UnicodeSecurityPreview-SLIDES.pdf

The Cyber Kill Chain

http://www.lockheedmartin.com/us/what-we-do/aerospace-defense/cyber/cyber-kill-chain.html

Diamond Model

https://apps.dtic.mil/docs/citations/ADA586960

Oops, they did it again: APT Targets Russia and Belarus with ZeroT and PlugX

https://www.proofpoint.com/us/threat-insight/post/APT-targets-russia-belarus-zerot-plugx