Practical Cyber Security for Cyber Security Practitioners

Prof. Sandeep Kumar Shukla

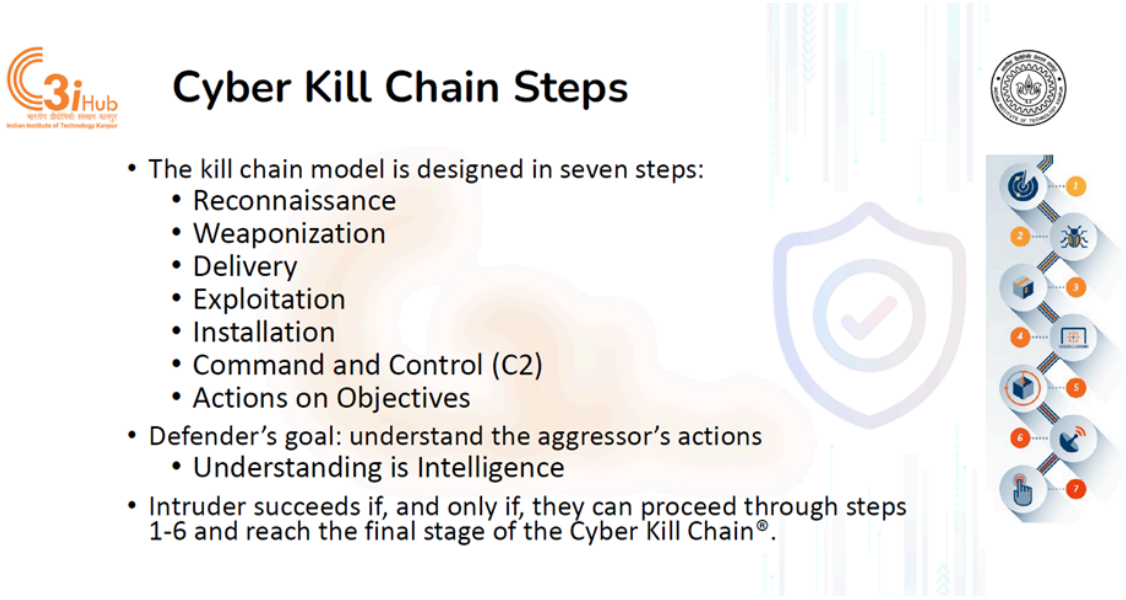
Department of Computer Science and Engineering

Indian Institute of Technology, Kanpur

Lecture 04

Mastering the Cyber Kill Chain: Command & Control and Actions on Objectives

So, today, what we are trying to do in this module, in this lecture, is to complete Module 2, which is the Cyber Kill Chain, the Lockheed Martin Cyber Kill Chain. We have covered the seven stages of the Lockheed Martin Cyber Kill Chain, namely reconnaissance, weaponization, delivery, exploitation, installation, command and control, and, eventually, actions on objectives.



RECONNAISSANCE *Identify the Targets*

- **ADVERSARY**
  - *The adversaries are in the planning phase of their operation.*
  - *They conduct research to understand which targets will enable them to meet their objectives.*
    - Harvest email addresses
    - Identify employees on social media networks
    - Collect press releases, contract awards, conference attendee lists
    - Discover internet-facing servers

- **DEFENDER**
  - *Detecting reconnaissance as it happens can be very difficult, but when defenders discover recon – even well after the fact – it can reveal the intent of the adversaries.*
  - Collect website visitor logs for alerting and historical searching.
  - Collaborate with web administrators to utilize their existing browser analytics.
  - Build detections for browsing behaviours unique to reconnaissance.
  - Prioritize defences around technologies or people based on recon activity.

WEAPONIZATION *Prepare the Operation*

- Adversary
  - Obtain a weaponizer, either in-house or obtain through public or private channels
  - For file-based exploits, select "decoy" document to present to the victim.
  - Select backdoor implant and appropriate command and control infrastructure for operation
  - Designate a specific "mission id" and embed in the malware
  - Compile the backdoor and weaponize the payload

- Defender
  - Conduct full malware analysis – not just what payload it drops, but how it was made.
  - Build detections for weaponizers – find new campaigns and new payloads only because they reused a weaponizer toolkit.
  - Analyze timeline of when malware was created relative to when it was used. Old malware is "malware off the shelf" but new malware might mean active, tailored operations.
  - Collect files and metadata for future analysis.
  - Determine which weaponizer artifacts are common to which APT campaigns. Are they widely shared or closely held?

DELIVERY *Launch the Operation*

- Adversary
  - Adversary controlled delivery:
    - Direct against web servers
  - Adversary released delivery:
    - Malicious email
    - Malware on USB stick
    - Social media interactions
    - "Watering hole" compromised websites
- Defender
  - Analyze delivery medium – understand upstream infrastructure.
  - Understand targeted servers and people, their roles and responsibilities, what information is available.
  - Infer intent of adversary based on targeting.
  - Leverage weaponizer artifacts to detect new malicious payloads at the point of Delivery.
  - Analyze time of day of when operation began.
  - Collect email and web logs for forensic reconstruction. Even if an intrusion is detected late, defenders must be able to determine when how delivery began.

EXPLOITATION *Gain Access to Victim*

- Adversary
  - Software, hardware, or human vulnerability
  - Acquire or develop zero-day exploit
  - Adversary triggered exploits for server-based vulnerabilities
  - Victim triggered exploits
    - Opening attachment of malicious email
    - Clicking malicious link
- Defender
  - User awareness training and email testing for employees.
  - Secure coding training for web developers.
  - Regular vulnerability scanning and penetration testing.
  - Endpoint hardening measures:
    - Restrict admin privileges
    - Use Microsoft Windows Defender Exploit Guard
    - Custom endpoint rules to block shellcode execution
  - Endpoint process auditing to forensically determine origin of exploit.

INSTALLATION *Establish Beachhead at the Victim*

- Adversary
  - Install webshell on web server
  - Install backdoor/implant on client victim
  - Create point of persistence by adding services, AutoRun keys, etc.
  - Some adversaries “time stomp” the file to make malware appear it is part of the standard operating system install.
- Defender
  - HIPS to alert or block on common installation paths, e.g. RECYCLER.
  - Understand if malware requires administrator privileges or only user.
  - Endpoint process auditing to discover abnormal file creations.
  - Extract certificates of any signed executables.
  - Understand compile time of malware to determine if it is old or new.

COMMAND & CONTROL (C2)
Remotely Control the Implants

- Adversary
  - Open two way communications channel to C2 infrastructure
  - Most common C2 channels are over web, DNS, and email protocols
  - C2 infrastructure may be adversary owned or another victim network itself
- Defender
  - Discover C2 infrastructure thorough malware analysis.
  - Harden network:
    - Consolidate number of internet points of presence
    - Require proxies for all types of traffic (HTTP, DNS)
  - Customize blocks of C2 protocols on web proxies.
  - Proxy category blocks, including “none” or “uncategorized” domains.
  - DNS sink holing and name server poisoning.
  - Conduct open source research to discover new adversary C2 infrastructure.

So, we have to wrap up by actually discussing a few important things about defense. We saw that the idea of the Lockheed Martin Cyber Kill Chain is that you can stop any of the stages of the Cyber Kill Chain, and that way, you break the chain that the adversary was trying to create in your organizational network. To do that, you have to have defense in depth. That is, you have to first try to stop the reconnaissance by creating very robust rules and policies with respect to social media presence, data leakage, etc. All internet-facing systems or services should be hardened.

If you fail to stop the reconnaissance, then you try to stop the weaponization. You have to have proper malware security and the ability to check what is happening in the devices inside your organizational network that people are using and connecting to your network. You need to have good antivirus software on people's devices to stop that part of the chain. However, in case you even fail to do that, you also have to try to stop the delivery. For delivery, you can have better email security, spam filtering, and phishing email filtering.You can actually make awareness a high priority in your organization so that people don't get into phishing traps, or you can conduct cyber drills or phishing drills to make people more aware and on their toes. You can also harden your internet-facing services and the operating systems, applications, or middleware that you use to make your presence on the internet more secure.

If you fail in stopping the delivery, then you have to try to stop the execution. To stop the execution, you need to figure out how you can detect when some unwanted binary is being executed and how you can stop it from being executed. You need endpoint security and agents on the endpoints to prevent this. If you fail in execution, then you need to intervene during installation when persistence is being created. To create persistence, adversaries might register it on the registry or put the binary into the startup folder, among various other ways to achieve persistence through installation.

You need endpoint security to stop this as well. If you fail to do that, then you have to stop the C2C (command and control) traffic. You need to observe the traffic and perform network monitoring to see whether any C2C traffic is going to certain IP addresses or URLs. You might need to use AI and ML techniques to detect which URL might be a C2C URL, or you need threat intelligence regarding various IPs that might be part of the C2C infrastructure of the adversary.

If you do not succeed in stopping the C2C traffic, then chances are the malware or payload is already working in your system. It is probably trying to perform privilege escalation, internal reconnaissance, lateral movement through the network, etc. To stop this, you need other techniques. For example, if they are trying to do data exfiltration, you need to implement data loss prevention and detect anomalous, large amounts of data leaving your system. For privilege escalation, you need to monitor privileged accounts and usage through Privileged Identity Management (PIM).

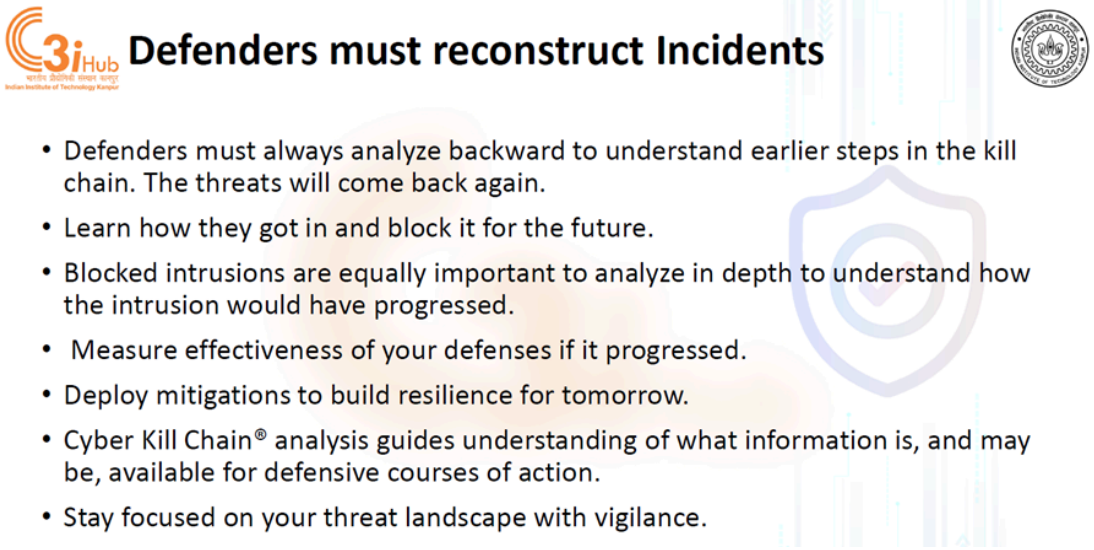
Or you may have to do various network monitoring to see whether there is internal reconnaissance, or you have to also see whether somebody is trying to encrypt your devices. As soon as the encryption starts, if you can detect that through the use of trap files and things like that, you can stop it. The main lesson here is that it's a defense in depth. You have to continue to organize your defense with the hope that you will stop it at one of the seven stages. This whole idea is based on the fact that the defender is vigilant and self-aware, monitoring the network and the activities on all devices within the network. All devices should have their logs and events recorded, and all kinds of information, such as new files being created and new logins happening, must be sent in real-time to a Security Information and Event Management (SIEM) server. From there, it should generate alerts based on rules or AI-ML models. Then, there is a whole remediation team sitting in the Security Operation Center (SOC) that looks at the alerts and takes immediate action. Some actions may be trivial, while others may require quite a bit of work.

To do this, you also have to train the people working in your SOC. They should be able to recognize signs of various kinds of intrusions and be aware of the modus operandi of various Advanced Persistent Threat (APT) groups. This way, if they see alerts and there is no specific AI-ML technique or signature-based technique to detect a specific APT group (such as APT28 or APT3), well-trained SOC engineers can still recognize the threat. Knowing the patterns and the various ways that threats are carried out by different threat actors is crucial. This is what threat intelligence analyst training is about, as seen in various certification programs by organizations like the EC-Council. The basic idea is to get ahead of the threat actor, always be vigilant, and maintain a defensive posture to stop them at any stage.

Now, the issue is how the defender learns. The defender learns by analyzing incidents. This is why you have cyber forensic experts who perform forensics after incidents happen. In SOC parlance, first, there are events. Events happen and may rise to the level of alerts because they look anomalous, or a series of events may look anomalous. Signature-based pattern matching might indicate to the SOC analyst that something needs to be looked at, generating alerts. For certain kinds of alerts, based on the threat

intelligence in your system, you might declare something as an incident. If the incident requires a lot of resources and involves many people and stakeholders, it is declared a cyber crisis.

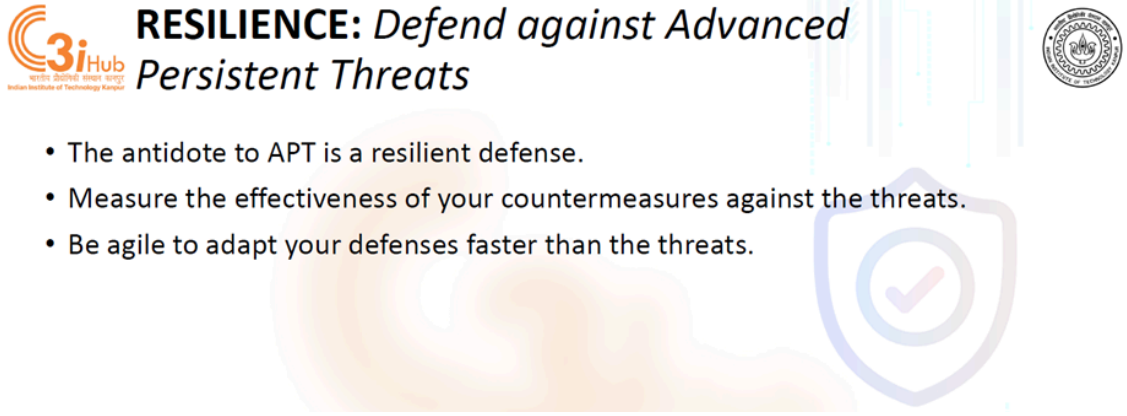
So, the decision regarding when something is declared as an incident, an event, or a crisis is based on the organization's cybersecurity policy. They have to think ahead about when to call something an event, an alert, an incident, or a crisis. When any of these things happen, especially incidents or crises, it most likely means you have been compromised. If you have been compromised, there are various kinds of indicators in different parts of your system. It could be in your network packet trace that you have stored. If you didn't detect it in real-time, then you have the packet stream stored. You also have devices that might have been compromised with malware, so you need to look at their memory for in-memory dumps and examine their disk file system to figure out what happened. By analyzing the network trace, you can reconstruct how the malware came in. Did it come through an email, as a result of someone visiting a website, or through a USB drive? There are various possibilities.



So, you have to look at these things, and once you do, you have to know how to block them, right? You might have failed this time, but you will learn from this how to block it in the future. In case you have an incident that did not lead to a large-scale compromise, data exfiltration, or ransomware encryption, then we can say that the attack has been thwarted. However, even in such cases, you have to perform forensics to analyze how it happened, which part of your defense did not work properly, and why it didn't work. Was it because the patterns were not there in your system to recognize the threat, or did your AI-ML model fail? Were your firewall rules inadequate, or were your email filters, which are supposed to stop malicious attachments, not working? This is how you gather your own threat intelligence.

As I mentioned before, threat intelligence can be collected from various sources. One source is learning from your own incidents. Additionally, you might get threat intelligence from various vendors who collect incidents from around the world and convert them into threat intelligence. You might also receive it from your regulators or your CERT (Cyber Emergency Response Team) collections and feeds. There are various ways to get threat intelligence and feed it into your defensive equipment, such as firewalls, intrusion detection systems, and endpoint detection systems. Continuously upgrading this software, the network monitoring software, intrusion detection software, and firewalls is essential to prepare for the next onslaught. This is the essence of the Defenders' strategy.

Now, the thing we talk about these days is that we can never claim that our organizational system is 100% secure. Nobody can claim that because there will always be certain ways for the perpetrator to get into your system. Like we discussed in the Stuxnet case, it was through USB drives. In the case of the attack on Ukraine, it was through phishing emails sent to high-level executives. So, you can never be 100 percent sure that you will be protected.

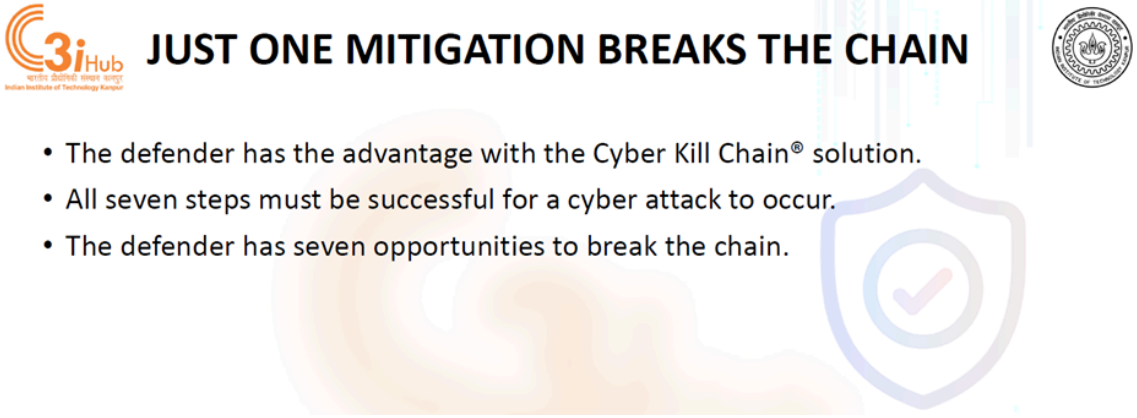


Having protective devices or protective hardware, software, or applications does not necessarily mean that you can rest easy. Nowadays, we say that we do not aim for a cyber secure system; instead, we strive to make the system cyber resilient. Resilience is the term we discussed before. It refers to the ability of your system to continue functioning even in the face of very strong attacks. In other contexts, resilience may not necessarily relate to attacks, but in this case, it is about your ability to function, maybe with reduced performance or effectiveness, but without a complete crash. This is what we call a graceful degradation of quality of service, but you should at least provide some functionality.

In case the compromise does affect functionality, you must recover very quickly. As soon as the attack stops or is withdrawn, you should be able to recover very fast. The time required to recover is a measure of your resilience. If your system gets attacked and is inoperative for three, five, or seven days, it is bad. That means you are not very resilient. If it takes only a couple of hours, you are much more resilient than the other one.

So, that's the idea of resilience. Resilience is the only way we can actually measure how effective our cybersecurity is. We cannot have, as I said, zero risk or zero attack scenarios. As I also mentioned before, the APT (Advanced Persistent Threat) groups are funded by nation-states and are very resourceful and knowledgeable. Therefore, they find various ways to carry out attacks, and you can be assured that they will sometimes penetrate your protections. Hence, you need to know how to function and recover very quickly.

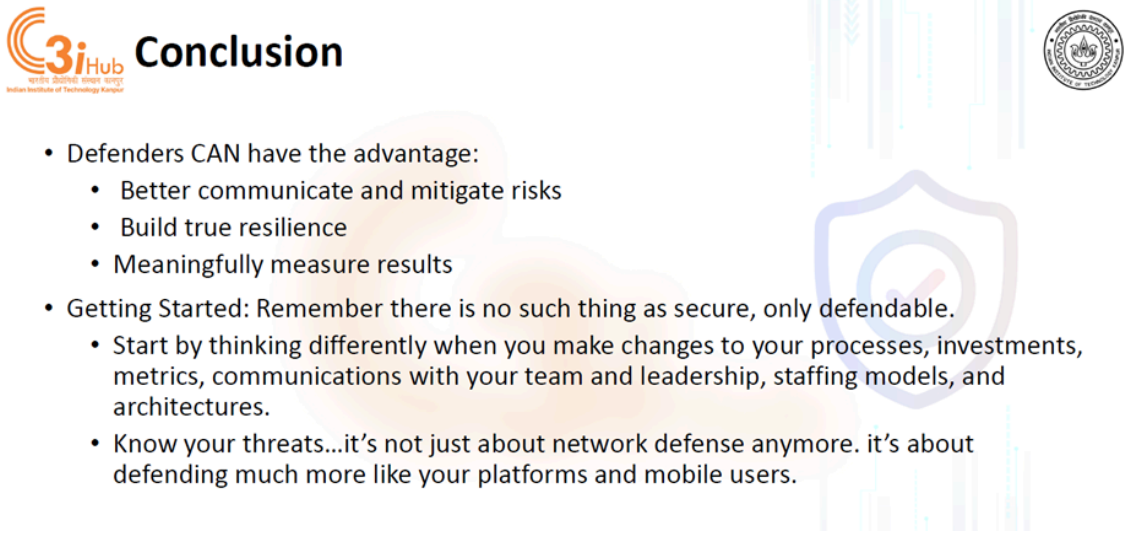
In some cases, you may have to respond rapidly. You have to be agile so that you can defend faster. Later on in this course, we will talk about the measure of resilience or what we call CRR (Cyber Resilience Review). It is a cyber resilience measurement that you will look at, and you will see that there are various levels of resilience. You may be totally non-resilient or resilient to some extent, but the best resilience comes when your system is adaptive. Your system should adapt to the onslaught of DDoS attacks or have enough redundancy to failover to another part of the system to provide functionality to its clients or customers. Adaptiveness or adaptivity is the name of the game if you really want to be resilient.



So, as I said, and you must have read the paper from Lockheed Martin, the idea of the kill chain is that just one mitigation breaks the chain. They think it is an advantage for the defender because the defender has to break the chain at one stage, whereas the adversary has to complete all the stages. But I do not see that as realistic because you cannot be sure. You cannot simply target the installation stage, reconnaissance stage, or command and control (C2) stage to break the chain.

You cannot be sure that if you just focus on, let's say, command and control, you will catch all kinds of C2 traffic to stop them. As I mentioned before, there are algorithmic URL generation techniques and very quick migrations from one IP to another for the command and control server. There are various ways adversaries can fool your tools or abilities to break the chain. Therefore, I think you have to do defense in depth and arrange for defense at every stage.

That way, you do not get a significant advantage. As it says here, the defender has seven opportunities to break the chain, which means it has to have seven types of defenses. You cannot just rely on one stage, thinking that breaking that stage will provide mitigation.



So, in conclusion, we can say that defenders can have the advantage if you think from the perspective that you have to just break one stage while the adversary has to complete all the stages. However, the bigger issue is that the defender has to do better communication and mitigate risks. When I say better communication, I mean that you have to always assess the risk. Later in this course, we will talk about risk assessment and risk mitigation. You also have to ensure that all your employees, who are in charge of various cyber assets in your organization, know about these risks and the various threats they might face. This will help them make better decisions while using these assets.

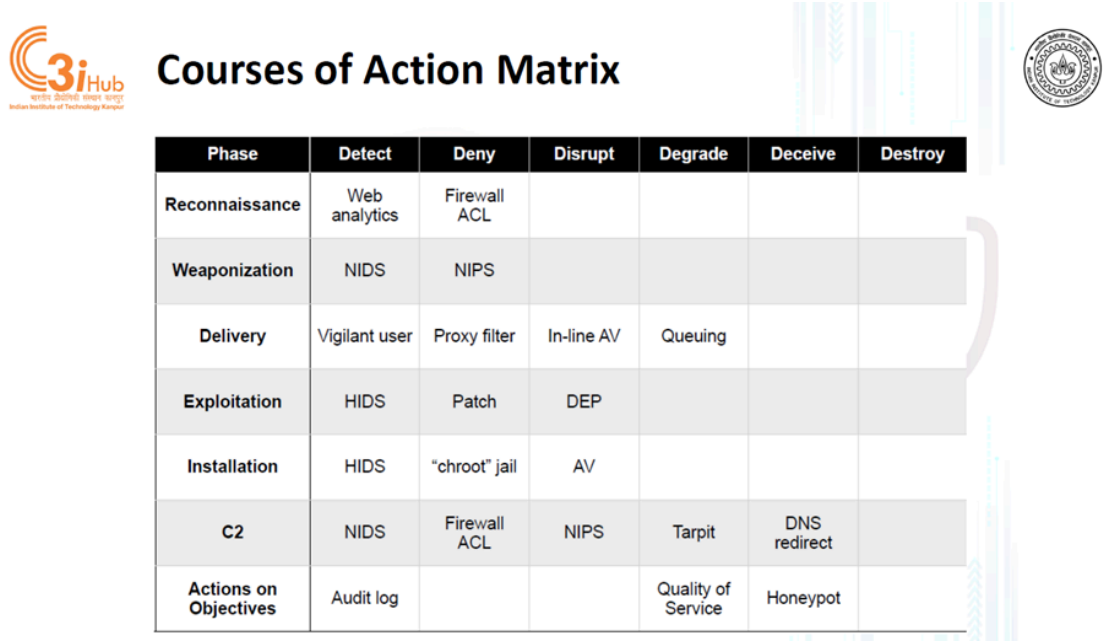
You also need to build a resilient design. Resilience is now the key term in cybersecurity. Ensuring that your system is cyber resilient is more important than ever. You have to measure and continuously improve. This is often misunderstood by many cybersecurity teams. They might do a one-time risk assessment to figure out the possible threats and then implement defenses like perimeter defenses, endpoint defenses, network monitoring, and AI-ML tools for intrusion detection or malware detection. They might also train their people. However, they must continuously measure how effective their defenses are and identify where they went wrong in an attack or where they can improve, even if they did not face a known attack. Attacks are happening all the time.

The question is how aware and vigilant you are, and how effective your monitoring systems are. You may or may not see the attack, but if you want to improve, you have to measure how effective you are. Remember, there is no such thing as a fully secure system. Systems are only defensible, but not necessarily 100 percent secure.

You have to think differently than you might have been doing. Most cybersecurity training focuses on network security, application security, or operating system security. We do not necessarily talk about highly resourceful attackers and how they can string together vulnerabilities in your network, perimeter defenses, endpoints, devices, and people through social engineering. You have to start thinking about that. Once you do, you can identify where you might need to improve your cybersecurity posture. Is it the processes? Is it that you are not investing enough? Are you not communicating the risks

and threats to the team and leadership to get the proper investment in cybersecurity? Is it staffing issues? Or is it that your architecture is not strong enough?

Additionally, you have to know your threats. To do well in cybersecurity, you need to understand what the threats are. Many times, we get into trouble because we did not think about the threats. Consider how someone could use social engineering, exploit an unpatched application, operating system, or service running on the internet. Think about whether unnecessary open ports are being exploited, or if there are logins being attempted several times and eventually succeeding. Ensure that OTPs and other security measures are properly protected. All these factors need to be considered.



Phase	Detect	Deny	Disrupt	Degrade	Deceive	Destroy
Reconnaissance	Web analytics	Firewall ACL				
Weaponization	NIDS	NIPS				
Delivery	Vigilant user	Proxy filter	In-line AV	Queuing		
Exploitation	HIDS	Patch	DEP			
Installation	HIDS	"chroot" jail	AV			
C2	NIDS	Firewall ACL	NIPS	Tarpit	DNS redirect	
Actions on Objectives	Audit log			Quality of Service	Honeypot	

So, the last thing in this context is the courses of action matrix. As I have been telling you, this is just a model. This is one way of organizing your thoughts and your plans to make your system resilient. What this matrix shows, and we will see a similar matrix in another context when we go beyond the Lockheed Martin Cyber Kill Chain and delve into MITRE TTPs or the unified kill chain, is a structured approach to defense.

If you remember this table, it can be very beneficial for you because it shows that at various stages, there are ways to detect whether something is happening, ways to deny something from happening, ways to disrupt that thing from happening, ways to degrade the ability of the adversary, and ways to deceive the adversary into thinking that it is achieving its goal while you are actually observing them. You can also destroy the adversary. In the MITRE DEFEND case, this approach is formulated slightly differently, but it provides good terminology and ontology for organizing your defense and resilience.

For example, here we are saying that if you want to know whether someone is conducting reconnaissance, there are many ways to do reconnaissance. One possible way shown here is visiting your organizational websites to identify people, their contact addresses, and so on, to perform spear phishing or phishing emails in a very targeted manner. These emails are personalized, increasing the likelihood that the recipient will trust the email and click on a link. To know whether strange people are visiting your websites and trying to scrape data, you can look at web analytics, Apache logs, NGINX logs, or whatever web server you have. You can try to detect anomalous activities for detection.

So, if you want to deny reconnaissance, you can use various methods like access control lists or firewalls to restrict access to certain web pages. Some web pages may be behind an authentication wall or a firewall. If they try to automatically scrape your websites, you can use a web application firewall to stop that. This is an example of denial. You might also consider disrupting or deceiving them by putting in names of people who do not belong to your organization and wrong addresses, but usually, that is not done due to potential repercussions regarding the company's reputation.

For weaponization, if someone is trying to exploit your weaknesses, you can use a network intrusion detection system (NIDS) to detect emails with malware attachments or attempts to send payloads through the network to target your web services. If you are more advanced, you can use a network intrusion protection system (NIPS) to provide not only detection but also protection by stripping off those payloads or stopping those packets and sending them for forensics, thereby denying their delivery to the actual target.

In the case of delivery, such as through email phishing or vishing, detection requires vigilant users. However, you can also stop delivery through proxy filters. Web proxies or email proxies can filter content before it reaches the end user. You can also disrupt them by using inline antivirus (AV), which doesn't wait to detect malware after the payload reaches the endpoint. Since the payload has to go through your network to eventually reach the target's email server or the endpoint email client of the user, you can observe network traffic and reconstruct the files being delivered. If you find that a file is malware, you can stop it, preventing it from being delivered to the user.

But, in case it is encrypted traffic, the best way is to use web proxies so that the termination point of the encryption is at the proxy, not all the way to the end user. At the proxy, the traffic will be unencrypted, and you can apply AV to the traffic if it contains malware. You can also degrade performance by slowing down suspicious traffic. Not necessarily stopping it, but creating queuing delays, so that immediate actions are hindered. During the queuing, you can apply additional filtering.

Now, if you focus on the exploitation phase, what happens is that the payload is already on the user's device. It has been delivered and is on the user's device. This is where host intrusion detection systems (HIDS) or endpoint detection systems (EDS) come into play. Exploitation can be stopped if the intrusion detection is effective. Typically, these systems have an agent on the device that reports all activities to a management server. By

the time the management server realizes that the endpoint has been infected and malware has been downloaded, it might be too late to stop it.

However, to deny exploitation, you should ensure that your operating system, middleware, firmware, and applications are fully patched. There should not be any known vulnerabilities that users might exploit if they manage to execute or detonate the payload. This does not guarantee complete safety because there could be zero-day or zero-click vulnerabilities that cannot be fixed even with a patched version.

To disrupt execution, you can use Data Execution Prevention (DEP). DEP is one way to stop buffer overflow attacks by preventing code from being executed in certain regions of memory not intended for code execution.

It does not necessarily mean that DEP (Data Execution Prevention) provides 100% freedom from buffer overflow attacks. There are other ways to execute buffer overflow attacks, such as return-oriented programming and heap overflow. DEP is one way of stopping the execution of binaries from the program stack, but it is not exhaustive or 100% guaranteed. It can help, but it is not foolproof.

During the installation phase, the adversary tries to create persistence by making the downloaded binary write itself into the startup folder or the auto-run registry key. In this case, HIDS (Host Intrusion Detection System) can help in detection. You can configure it so that any binary outside a specified set runs in a limited environment without access to the entire file system. This is known as a chroot jail. However, this approach can be problematic for users, especially developers who need to run various binaries that are not malware. So, placing everything in a chroot jail may be inconvenient for users.

Disruption can occur when the antivirus quarantines a binary or payload, preventing its execution. This disrupts the installation phase.

In the command and control (C2) phase, you can detect C2 traffic with a network intrusion detection system (NIDS). You can deny C2 traffic by configuring the firewall quickly and adaptively or by using access control systems. Micro-segmentation or network segmentation can also help isolate certain network parts, making it difficult to establish C2 communication.

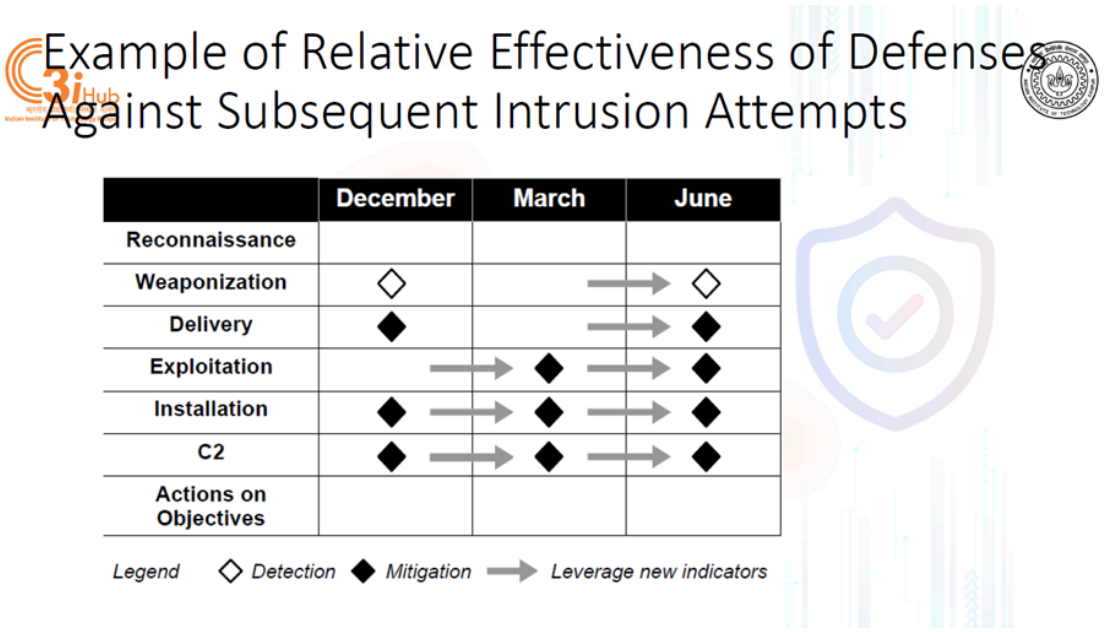
To disrupt C2 traffic, you can use a network intrusion prevention system (NIPS). Once it detects C2 traffic, it not only reports it but also effectively blocks it by configuring rules or firing certain functions.

TARPIT is actually very similar to queuing. It is a degradation method where you slow down the traffic so that the C2 server does not have the kind of traffic rate it needs. This can disrupt the effectiveness of the C2 server, reducing its ability to communicate with the malware on your device. You can also deceive by doing a DNS degrade, where the local DNS resolver translates the URL into a known IP address that you control, preventing the traffic from reaching the adversary's server. However, modern malware is often aware of this trick and may self-destruct when it detects a DNS redirect, preventing you from doing any forensics.

In the actions on objectives phase, the adversary might try to encrypt a device, move from one machine to another through a weak protocol, or perform privilege escalation. Audit logs can help in detecting these activities. Denial is very difficult at this stage because the adversary has successfully completed all previous stages. However, if your applications and operating systems are fully hardened with no vulnerabilities, you can stop privilege escalation and other actions.

This phase is a catch-all in the Lockheed Martin Cyber Kill Chain, as there are many potential actions the payload can take. Depending on the action, the denial method will vary. You can degrade the quality of service to slow down the malware, giving you more time to contain the damage. Honeypots are a way to deceive the attacker. You can use honeypot orchestration to redirect attack traffic to a honeypot service, which mimics the actual service being attacked. The attacker will then believe they are working on your device and may bring in additional payloads or communicate with other C2 servers, allowing you to gather threat intelligence.

So, that's all about this approach. The other important thing is to continuously measure your effectiveness over time to ensure you are improving.



So, here is a kind of made-up example. An organization may want to see whether in December it was able to detect something during the weaponization phase, but did not stop there, and instead could stop the delivery by having mitigation measures in place. Now, in March, it might find that it was able to mitigate exploitation and installation in situ, and then in June, it might say that it was detecting weaponization again, and so on. You can figure out that in December, you learned something and used that knowledge.

This is what they call leveraging the new indicators. You get some indicators, like which things were compromised, what IP addresses were used in the C2, what malware was used, and all this information is called the indicators of compromise. You feed that into your protection devices, like your firewall, endpoint detection, and so on. Hopefully, next time, if the same adversary comes with the same infrastructure and techniques, you should be able to stop them. However, if it is a different adversary with different indicators, you may not be able to stop them based on this learning from December.

You might have other sources of threat intelligence about other threat actors that you have also fed into your devices. From that, you might be protected from this other threat actor. Each threat actor has a different set of indicators of compromise, so one threat actor's indicators may not always help in stopping other threat actors. This is an ongoing process for the cyber defense team of an organization to continue learning new threat indicators, new indicators of compromise, new ways of attacking, and new threat models, and then using that information very effectively. If you do not use that information effectively, then it is not useful.

That is the idea. Again, the Lockheed Martin Cyber Kill Chain is a way of organizing your thoughts and your defense in response to what the attackers might do. It is a way of thinking or conceptualizing what the attackers might do. It does not mean that the attacker is exactly following these seven stages or that there aren't other things outside these seven stages that the attacker might do, which may not be properly captured in this model. So, think of it as a model and a way of thinking about the adversary and the threats, and a way of organizing your defense based on that framework. We will now move on to other frameworks where you will see more fine-grained ways of looking at these things. That is where we will stop this lesson.