SOC Tier 1 Notes

Jobs have to do in this carrier :

1. Monitoring : Investigating , Check Alerts
- Tools : 1- SIEM like Splunk 2-EDR 3- Antivirus …
2. Meeting and Group work
3. Working with IDS/IPS
4. Ticketing : On a platform that the organisation has you should submit the true alerts on it.

#Escalate : If a T1 notices an incident and doesn't know if this incident is valid or not they will pass it to T2. This passage is called escalate.

#SOC T2 is about hunting.(they almost do everything happens in T1)
#SOC T3 : In some companies you may see T3 of SOC and they are professionals on the job it's like pros of SOC T2

A brief Introduction to SOC :

As we said earlier in SOC the person should investigate the incident to prevent it and monitor and respond to it.

What things you have to do as an SOC Analyst :

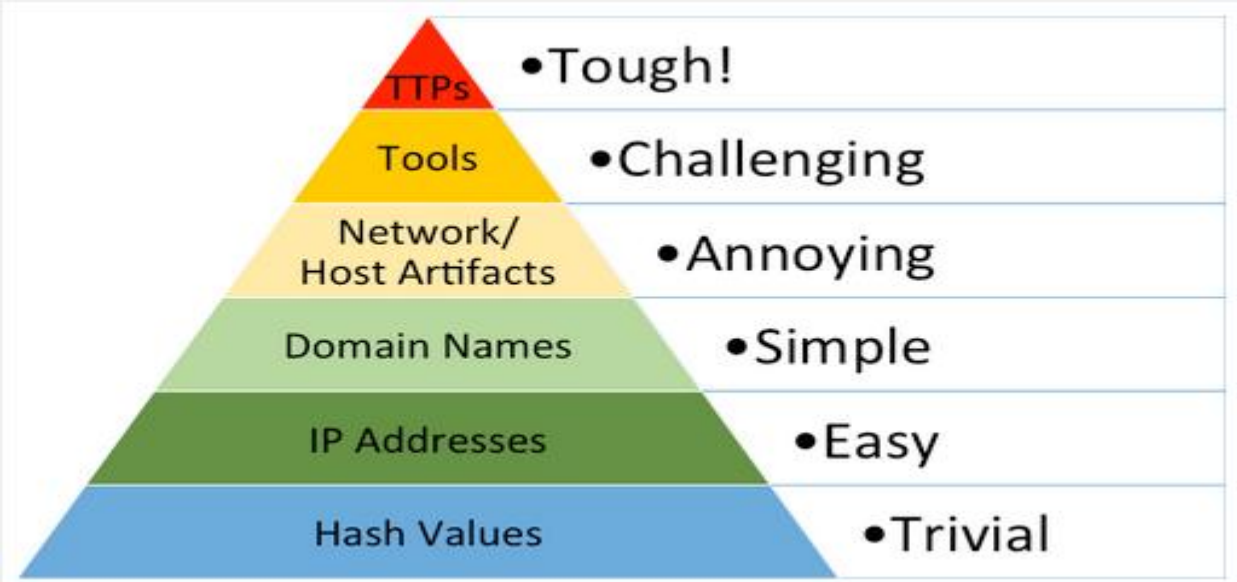
1. Reporting / Ticketing / Check Logs
2. KB(Knowledge Base) : Things We have Learned from an incident to Enhance our knowledge about things/Developing / Researching / TI(Threat intelligence) / Collect and Parsing Logs

3. Aggregation : It can happen in many things Like in many Incidents or Alerts that have been created They correlate to Reach Chains of Attack.
   # Agg and Correlation should be with Application but it can be manual.(In Companies can be Different).

Difference Between NOC & SOC:

NOC or Network Operation Centre is a unit that keeps an organization Network but in SOC they keep An organization network safe as we said earlier.
SOC Is About Security But NOC is About Keeping and Solving Network Problems.
#NOC is about incidents that humans are not Involved with it But in SOC humans are Involved with it.

Pyramid Of Pain:



The relation between different parameters to detect intruders. It means the evidence that was left from the intruder (the Evidence is shown in photo above).

It also means how we can make it hard for an intruder to escalate his access to our system.

-Hashes: the first step we can do is to analyse our process hashes and submit the hash that is not valid (process or program that is not usual).

Checking hashes is not so good because the intruder can change the hash only by changing a byte.

-IP Address : Checking the detected IP that has been marked as intruder but the intruder can change his IP too.

-Domain Names: Check the domains that the intruder has been used it but this also can be covered like Punycode or Homograph

-Networking/Host Artifacts: it's very difficult to perform activity without leaving a mark so Start looking for Files Directories or.. That has been detected or On network, check for distinctive transaction values, especially protocol errors or just misinterpretations (Distinctive URI patterns, User-Agent Strings, Typos).

-Tools: If you see the same tool over and over again, you eventually get really good at detecting it(known tools or the tools that have been set as an alert). No matter what incidental changes they make, your detection mechanism can deal with them. To continue they need new tools.

-TTPs: An expression of the attacker's training. Know his tactics, procedures… and stop his attacks. This step makes the attacker change his tactics. This change can be very expensive for the attacker.(Data staging tactics, Data Staging techniques, Data staging Procedure)

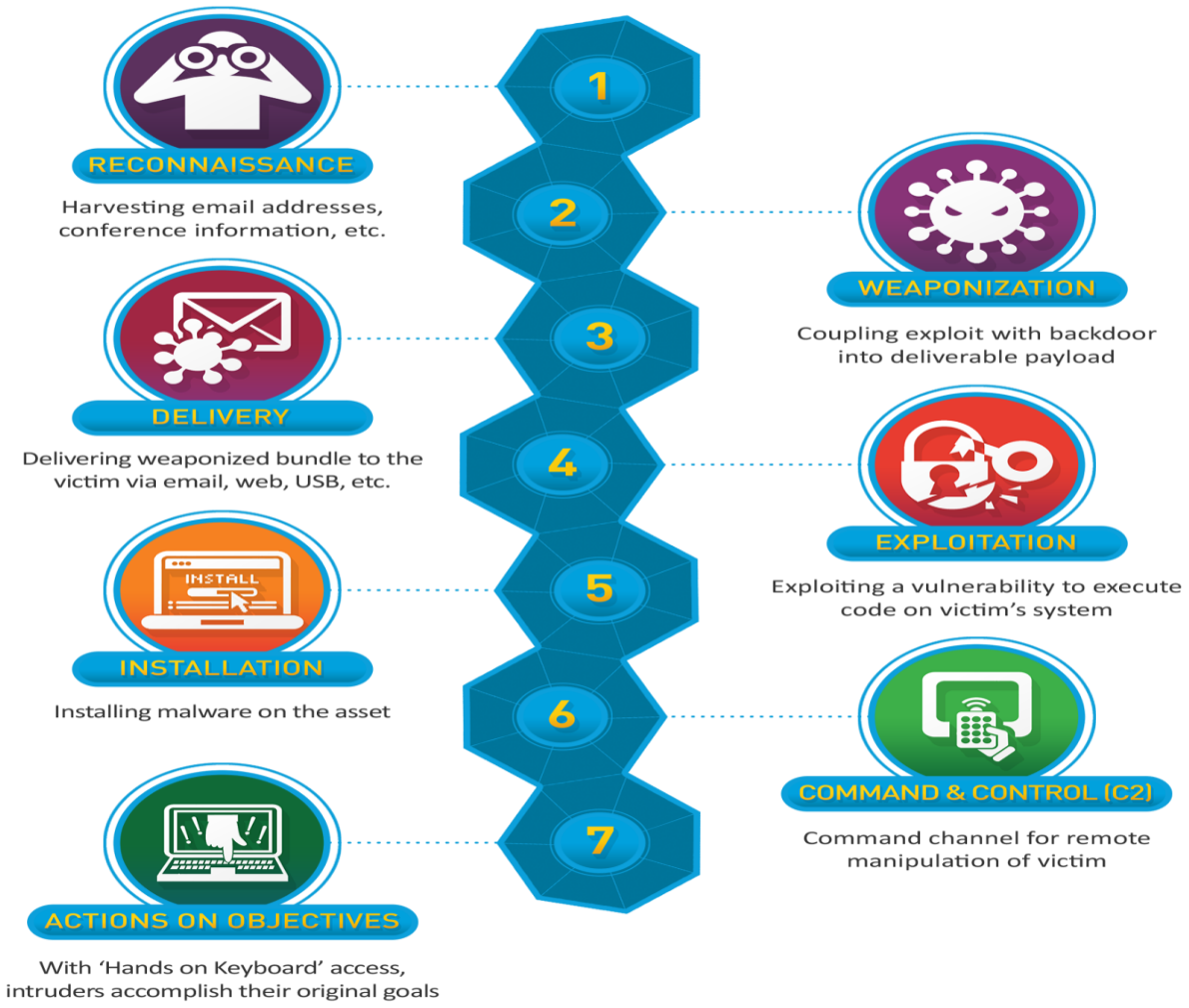
All the things said above are going to be set on rules or our alerting system to prevent the intruder.

Cyber Kill Chain:

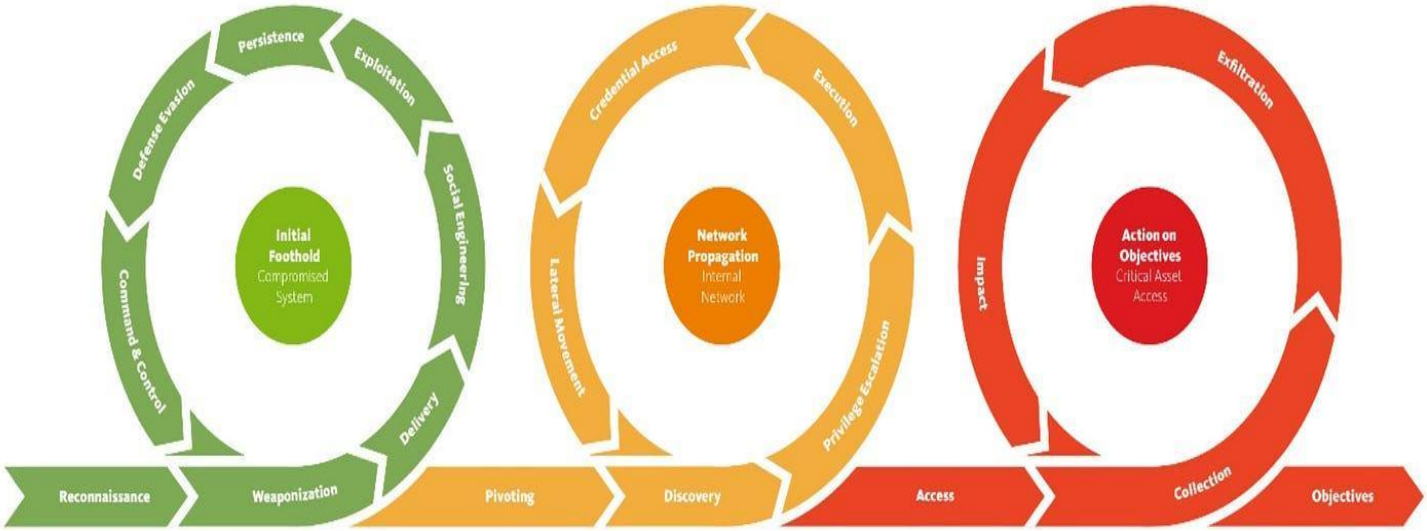
Steps or stages of an attack from start to end are called Cyber kill chain.

Cyber kill chain Steps:

1. Scan (Reconnaissance)
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command & control
7. Actions on Objective



Unified Kill Chain (UKC)



Phase 1 : (IN)

1. **Reconnaissance** – Gathering information about the target
2. **Weaponization** – Preparing attack tools and payloads
3. **Social Engineering** – Manipulating human targets (an explicit phase the original model lacks)
4. **Delivery** – Transmitting the weaponized payload
5. **Exploitation** – Triggering vulnerabilities to gain execution
6. **Persistence** – Maintaining access across reboots and credential changes
7. **Defence Evasion** – Avoiding detection by security controls
8. **Command & Control** – Establishing communication channels

Phase 2 : (Through)

9. **Pivoting** – Using compromised systems as a launching point
10. **Discovery** – Mapping the internal network, identifying assets
11. **Privilege Escalation** – Obtaining higher-level access rights
12. **Execution** – Running malicious code on additional systems

13. **Credential Access** – Harvesting passwords and authentication materials
14. **Lateral Movement** – Spreading across the network to reach targets

Phase 3 : (OUT)

15. **Collection** – Gathering target data from compromised systems
16. **Exfiltration** – Moving collected data out of the environment
17. **Impact** – Disrupting operations, destroying data, or deploying ransomware
18. **Objectives** – Achieving the overarching strategic goal

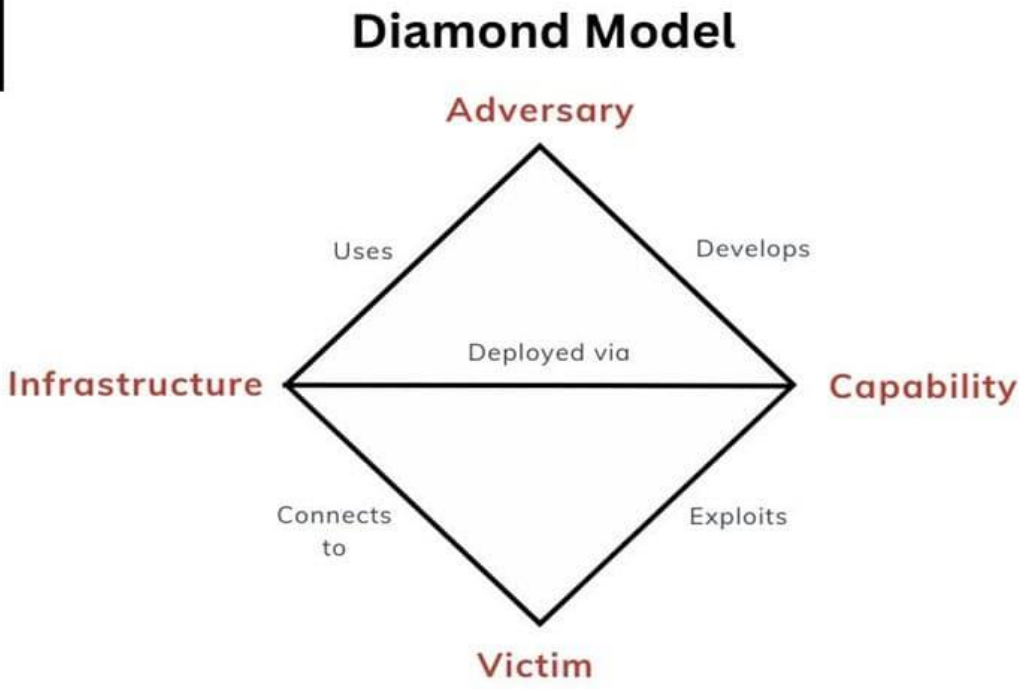
Difference between Pivoting and lateral movement :
Pivoting is more about expanding his authority and lateral movement is about going from a system to another system. (Typically lateral movement is for going from a system to another system that is in the same Network.)

Diamond Model

The Diamond Model of Intrusion Analysis (or Diamond Model for short) is a simple framework for analyzing and understanding cyber threats. Defenders use it to organize and structure their intrusion analysis by categorizing data into one of its four components.

- **Adversary**: Who did the attack?
- **Capability**: How did they do it?
- **Victim**: Who was targeted?
- **Infrastructure**: What was used?

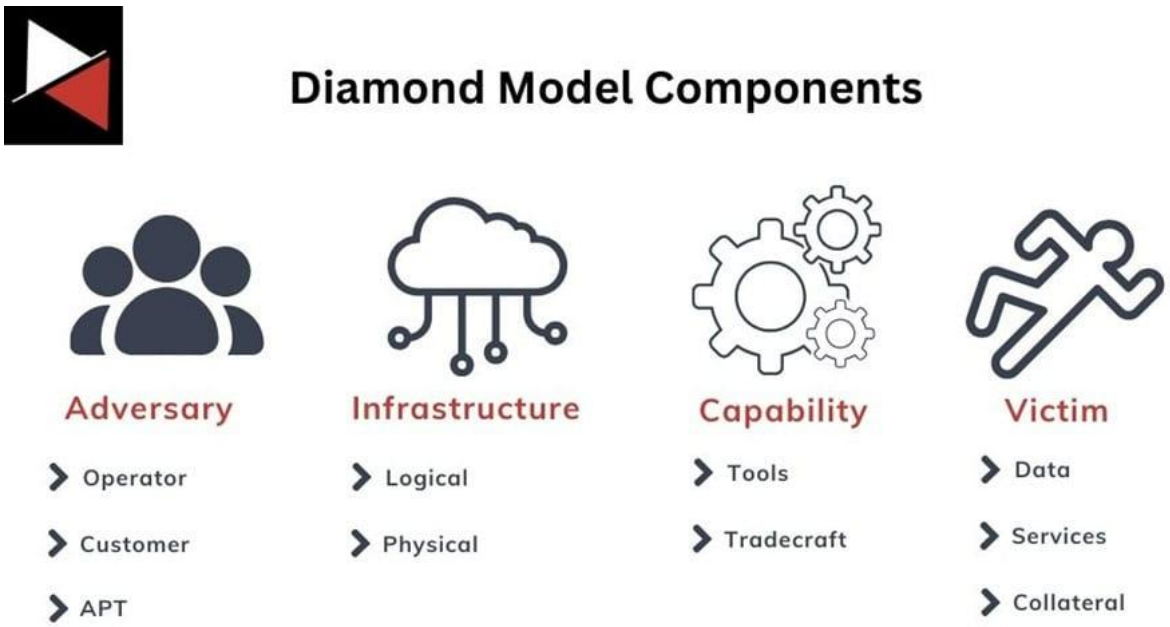
These four components are present in every cyberattack and are interrelated. In its simplest form, the model describes an "**adversary** deploys a **capability** over some **infrastructure** against a **victim**" (whitepaper), allowing an analyst to visualize the relationships between the key data points that fall under these categories.



The Four Components of the Diamond Model

The four components that make up the Diamond Model are Adversary, Capability, Infrastructure, and Victim.

Each of these components is linked. For example, an Adversary develops a Capability and then uses Infrastructure to connect a Victim to deliver this Capability. Mapping intrusion data to each component reveals links between them. You can then better understand an intrusion by asking questions about the components involved and pivoting between them to find answers.



Adversary

This is who is behind the attack. The nation-state, script kiddie, or cybercriminal who performed the attack. You can break the Adversary down into two roles:

- **Adversary operator**: The individual directly performing the attack.
- **Adversary customer**: The entity that benefits from the attack being performed.

An adversary can assume either of these roles or both. Typically, high-profile cyberattacks involve multiple operational teams: one for initial access, another for developing the malware, and a third for exfiltrating data. The ransomware scene has adopted a structure where an initial access broker provides a service, a ransomware gang licenses their software, and an affiliate actually performs the attack.

The number of operations teams and the structure of who is actually performing the attack can get complicated quickly. Nation-states

will utilize proxy groups, while ransomware gangs will employ affiliates; the line between government operations and government-sanctioned operations can become blurred. As such, it is usually more productive to track the Adversary customer, who benefits most from an attack, rather than the specific individual or malware used.

This is not always the easiest thing to do using intrusion data.

To get started, track the online presence used, accounts seen (email, social media, etc.), and the intent of an attack rather than immediately labeling a state or gang. This will enable you to correlate these Adversary data points with other campaigns and attribute the attacks more accurately to an entity or campaign based on available data.

Capability

Capability refers to the tactics, techniques, and procedures (TTPs) the Adversary uses to perform the attack. This can be categorized into:

- **Tools**: The hacking tools, malware, or exploits used during an attack.
- **Tradecraft**: The hacking techniques used by the Adversary, such as exploiting living-off-the-land binaries and scripts (LOLBAS) or commands run on systems. The MITRE ATT&CK matrix covers these well.

When mapping Capabilities to the Diamond Model, focus on specifics. General tools or tradecraft are not very useful if you want to track them. Focus on data points that stand out, usually represented by choices an Adversary will make, including specific malware

configuration options chosen, custom malware used, and novel attack techniques or command-line options.

#These can be considered key indicators. Indicators that remain consistent across intrusions uniquely distinguish an attack campaign and align with a phase of the Cyber Kill Chain. They are indicators you can use to track attack campaigns or threat actors, differentiate intrusions, and hunt for them in your environment.

Infrastructure

An Adversary will deploy their Capabilities using Infrastructure. This is anything an attacker can use to deliver their capabilities. It can be physical, like a command-and-control (C2) server, or logical, like an email address or service account.

The attacker may use this infrastructure directly, such as a C2 server they connect to when performing their attacks, or something the Victim connects to, like a third-party file-sharing service where data is exfiltrated (e.g., file.io or Pastebin).

Almost anything can be infrastructure, from the process that runs a malicious DLL to the name badge an attacker cloned to gain access to the target building. Don't just think of domains and IP addresses as infrastructure.

Here are some common types of infrastructure you will regularly come across:

- Service accounts
- Email addresses
- IP addresses
- Domains
- C2 servers

- Personas (social media handles, phone numbers, Telegram channels, etc.)
- Cloud services
- File-sharing websites
- Tor nodes
- Compromised websites

Victim

The Victim is the recipient of the attack or "Capabilities deployed across Infrastructure by the Adversary." The victim is either an individual or an organization with assets impacted by the attack (computer systems, networks, data, etc.). Victims rarely have a direct relation to the attacker; instead, they are connected by Infrastructure or Capability.

A Victim is just a means to an end for the Adversary. A threat actor's intention is to compromise the confidentiality, integrity, or availability of the data or services a Victim controls. They do not care about the victim; they only care about the assets that the victim holds and the benefits they can gain from exploiting them.

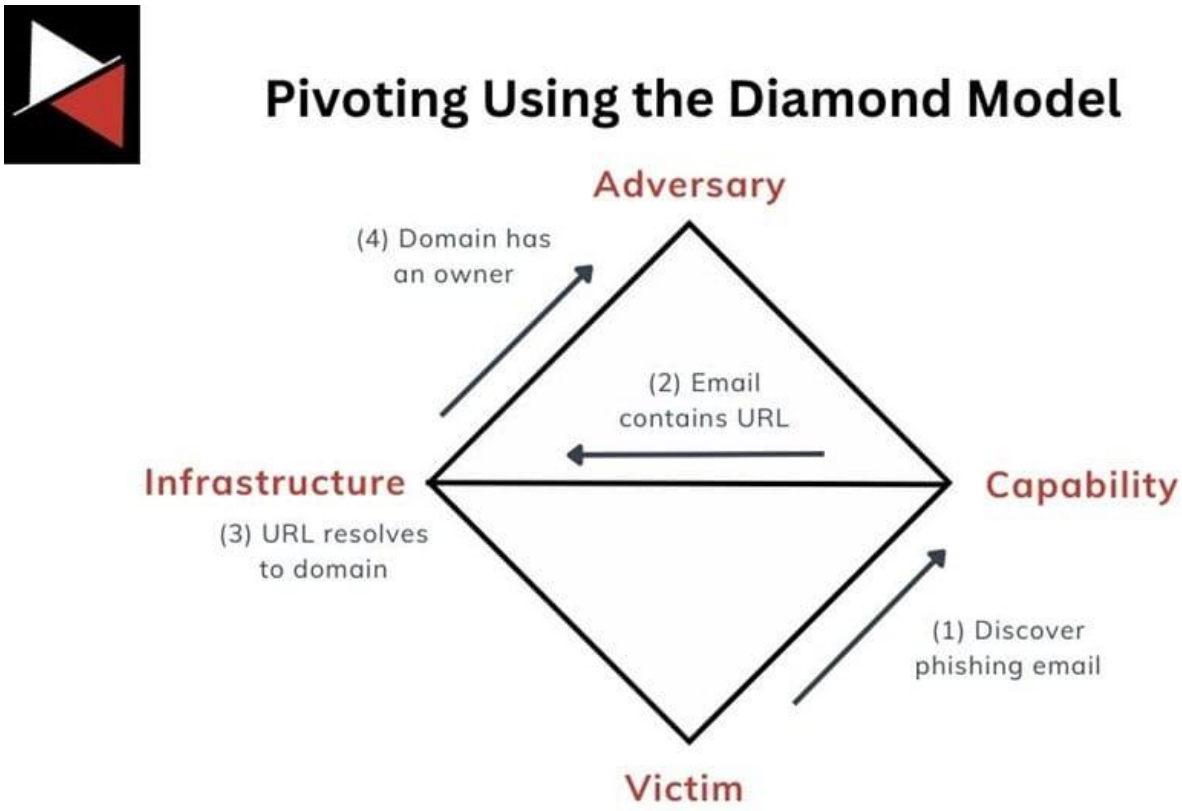
It is also important to determine who the actual victim of an attack is. An attacker may compromise a software vendor to perform a supply chain attack against their real target. The actual victim of the attack will mean different things for your analysis.

An analyst can use the Diamond Model from different viewpoints:

- They could take the Victim's view and look for Infrastructure or Capability that links an Attacker.
- They could be an Infrastructure provider and look for Attackers targeting Victims using their services.

- They could even take the Attacker's viewpoint and use Capability and Infrastructure to find a Victim.

An Example:



Applying the Diamond Model

The best way to apply the Diamond Model to real-world use cases is to combine it with a complementary attack framework, such as the Cyber Kill Chain. This structured framework categorizes the different phases of a typical cyberattack and allows you to organize your analysis into a coherent attack narrative. It helps you add the technical details of an attack to your intrusion analysis.

Read Combining the Diamond Model, Kill Chain, and ATT&CK to learn more about how the diamond model, cyber kill chain, and MITRE ATT&CK complement one another.

Think of each stage of the Cyber Kill Chain as having an associated Diamond Model, where every data point or piece of evidence you collect related to that stage can be mapped to an Adversary, Victim, Infrastructure, or Capability.

Aim to fill each Diamond Model component with as much data as possible for every stage of the kill chain. Some stages will be more difficult than others, and you may find that some stages don't involve an Adversary or Victim at all.

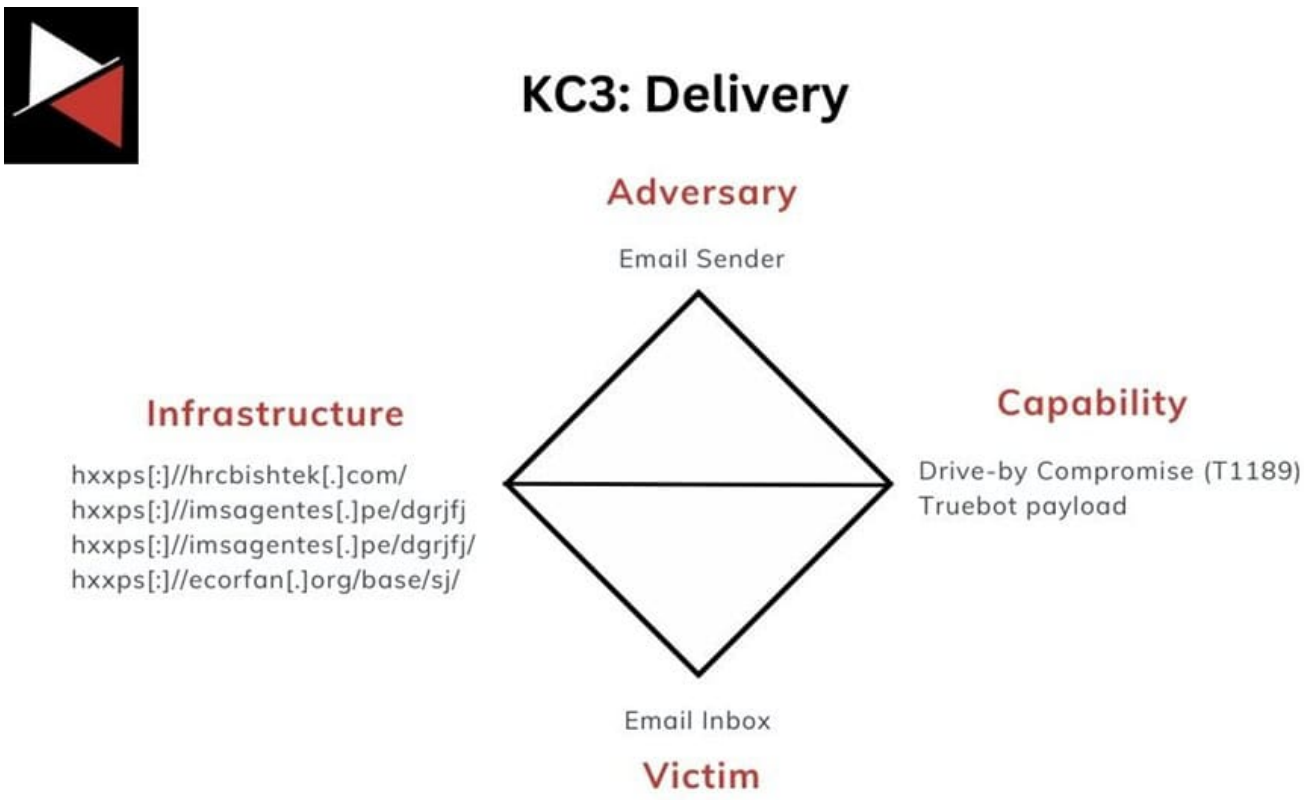
Nevertheless, one component must be populated in each stage of the kill chain, at least between stages two (Weaponization) and six (Command and Control), to declare your intrusion analysis complete.

Any gaps you find represent an intelligence requirement that must be fulfilled with supplementary data. This could include additional intelligence gathering, malware analysis, reverse engineering, or digital forensics. Based on your data, you can also assign a confidence level to each link between components. This lets you qualify assessments made using the model with estimative language.

This example demo won't map every stage of the kill chain (this is left as an exercise for the reader). Instead, it focuses on the Delivery, Installation, and Command and Control (C2) stages.
Delivery

The threat actor behind this intrusion used email as their initial delivery mechanism. They sent a phishing email to employees that redirected through a series of URLs before downloading a fake Adobe Acrobat document that was actually a True bot malware executable.

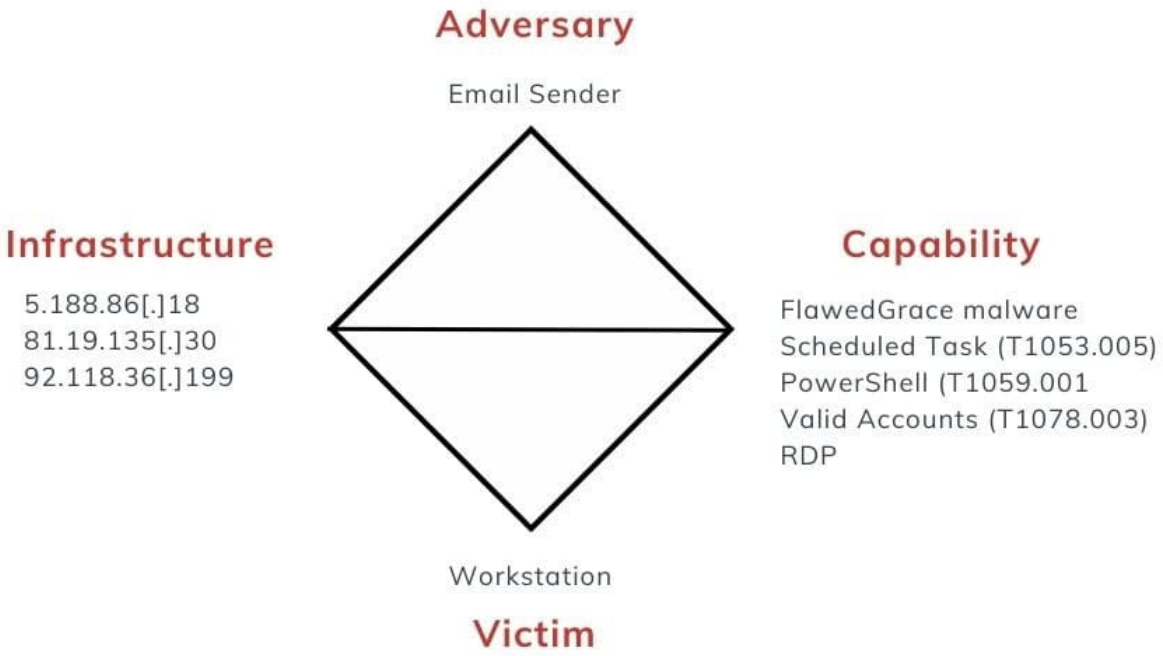
The components related to this stage of the attack are as follows:



Installation

After gaining initial access to a corporate workstation, the attackers utilized scheduled tasks and local administrator accounts with RDP access to maintain persistence. These persistence

mechanisms, created using PowerShell, gave the attackers an RDP connection to the compromised workstations.
The components related to this stage of the attack are as follows:

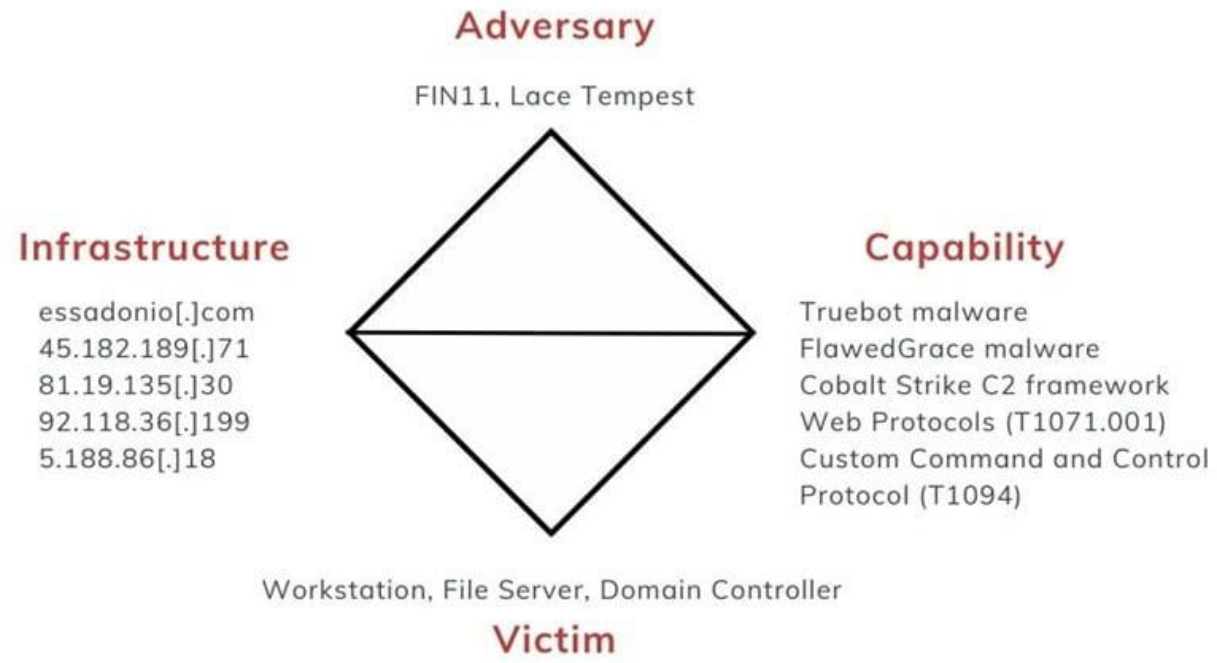


The IP addresses are those that the FlawedGrace malware contacted as part of the scheduled tasks created and RDP connections made.
Command and Control

Finally, for command and control over the systems they compromised, the attackers deployed Truebot, FlawedGrace, and Cobalt Strike malware to pivot around the corporate network and perform malicious actions. They communicated with their C2 agents through common web protocols (HTTPS) and custom protocols on port 443.

The components related to this stage of the attack are as follows:

KC6: Command & Control (C2)



The network infrastructure used by the adversary has been previously observed in campaigns by FIN11 and Lace Tempest, allowing it to be labeled as the Adversary. The new IP addresses now relate to the C2 servers used in the attack.

Once you have concluded your analysis of each kill chain stage, you can merge your individual Diamond Models into a single, comprehensive one, as the team at the DFIR report has done. However, it is often useful to keep the stage-specific models to compare them against other intrusions and cluster data together to track capabilities, campaigns, and actors at each stage of an attack.

These three Diamond Models allow you to begin asking questions about your data set, pivot to new intelligence, and connect the dots:

What Adversary is related to the Infrastructure I am seeing?
What Capabilities do these Adversaries have that I could look for in my environment?

What goals does the Adversary try to achieve, and what vulnerabilities will they exploit?

Conclusion

The Diamond Model is an essential tool in your cyber threat intelligence arsenal. It allows you to ask the right questions about your intrusion dataset and pivot to find the answers. All CTI analysts should know and understand how to use this model to perform intrusion analysis that can be shared with the wider cyber security community.

This article demonstrated how to utilize the Diamond Model for effective intrusion analysis and highlighted use cases where this model excels. It also discussed potential limitations of the model that you may encounter in the real world. To start using the model, use the practical demonstration showcased as a building block and map out the rest of the attack described in the DFIR Report article. Good luck!

The Diamond Model is just one structured analytical technique you should know how to use to become a CTI analyst.

MITRE ATTACK

MITRE ATT&CK® is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.

Linke To Source : Mitre Attack

You can us Mitre Attack navigator for Easy Access on projects.

Link: Attack Navigator

Also you can use it as a tool on linux and use it Offline.

Car Mitre Attack

The MITRE Cyber Analytics Repository (CAR) is a knowledge base of analytics developed by MITRE based on the MITRE ATT&CK adversary model. CAR defines a data model that is leveraged in its pseudocode representations, but also includes implementations directly targeted at specific tools (e.g., Splunk, EQL) in its analytics. With respect to coverage, CAR is focused on providing a set of validated and well-explained analytics, in particular with regards to their operating theory and rationale.

Visit this.

Data Model

The Data Model, strongly inspired by CybOX, is an organization of the objects that may be monitored from a host-based or network-based perspective. Each object on can be identified by two dimensions: its actions and fields. When paired together, the three-tuple of (object, action, field) acts like a coordinate, and describe what properties and state changes of the object can be captured by a sensor.

Object	Actions	Fields
authentication	error failure success	ad_domain app_name auth_service auth_target decision_reason fqdn hostname method response_time target_ad_domain target_uid target_user target_user_role target_user_type uid user user_agent user_role user_type
driver	load unload	base_address fqdn hostname image_path md5_hash module_name pid sha1_hash sha256_hash signature_valid signer
email	block delete deliver quarantine redirect	action_reason attachment_mime_type attachment_name attachment_size date dest_address dest_ip dest_port from message_body message_links message_type return_address server_relay smtp_uid src_address src_domain src_ip src_port subject to
file	acl_modify create delete modify read timestomp write	company content creation_time extension file_name file_path fqdn

		gid group hostname image_path link_target md5_hash mime_type mode owner owner_uid pid ppid previous_creation_time sha1_hash sha256_hash signature_valid signer uid user
flow	end message start	application_protocol content dest_fqdn dest_hostname dest_ip dest_port end_time exe fqdn hostname image_path in_bytes network_direction out_bytes packet_count pid ppid proto_info src_fqdn src_hostname src_ip src_port start_time tcp_flags transport_protocol uid user
http	get post put tunnel	hostname http_version request_body_bytes request_body_content request_referrer requester_ip_address response_body_bytes response_body_content response_status_code url_domain url_full url_remainder url_scheme user_agent_device user_agent_full

		user_agent_name user_agent_version
module	load unload	base_address fqdn hostname image_path md5_hash module_name module_path pid sha1_hash sha256_hash signature_valid signer tid
process	access create terminate	access_level call_trace command_line current_working_directory env_vars exe fqdn guid hostname image_path integrity_level md5_hash parent_command_line parent_exe parent_guid parent_image_path pid ppid sha1_hash sha256_hash sid signature_valid signer target_address target_guid target_name target_pid uid user
registry	add key_edit remove value_edit	data fqdn hive hostname image_path key new_content pid type user value
service	create delete pause start stop	command_line exe fqdn hostname image_path name

		pid ppid uid user
socket	bind close listen	family image_path local_address local_path local_port pid protocol remote_address remote_port success
thread	create remote_create suspend terminate	hostname src_pid src_tid stack_base stack_limit start_address start_function start_module start_module_name tgt_pid tgt_tid uid user user_stack_base user_stack_limit
user_session	lock login logout reconnect unlock	dest_ip dest_port hostname login_id login_successful login_type src_ip src_port uid user

What is the data model?

Objects

In the Data Model an object is much like an object in computer science. These are the items that data actually represent, such as hosts, files, connections, etc. Objects are the nouns of the Data Model vocabulary.

Actions

An action refers to a state change or event that happens on an object, such as an object's creation, destruction, or modification. These are the verbs that describe that an object can do, and what can happen to an object. However, there are cases where sensors do

not monitor actions in objects but merely scan for and check the presence of an object. Each action is represented in a coverage matrix (the 2D table). The actions are on the y-axis.

Fields

A field refers to the observable properties of an object. These properties may contain flags, identifiers, data elements, or even references to other objects. In terms of vocabulary, fields are like the adjectives. They describe properties about an object. A sensor monitors fields in the context of an object, and outputs these in some form of structured data. Once the data is ingested into a SIEM, the logs can be queried by forcing restrictions or patterns upon one or more objects, such as in an analytic. On the coverage matrix fields are on the x-axis.

Coverage

In order to gauge the usefulness of a sensor with respect to analytics, its output must be mapped into the Data Model. For each object that a sensor measures, it captures state. Some sensors periodically scan for objects, instead of monitoring for state changes. In these cases, state may be inferred by looking for changes in the properties of an object.

MITRE Defend

Visit this site for more information about defensing (It's Mitre site).

MITRE AEP

Follow this repo for simulation and tips on defense side. It provides you with explanations and commands to use.

DEMO TRY

Use TryHackMe Eviction room and solve the room by using Car Mitre Attsck website that I introduced before.