Lecture 24: Detection

Stephen Huang

Content

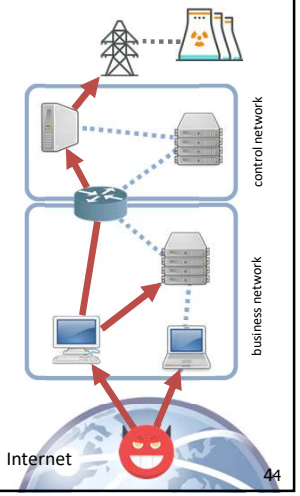
1. Adversarial Tactics
2. Intrusion Detection
3. Anonymity Networks

1. Adversarial Tactics

- Ukrainian Power Grid Attack
- In December 2015, attackers disrupted three energy distribution companies in Ukraine
  - several outages that caused approximately 225,000 customers to lose power across various areas
  - Widely reported

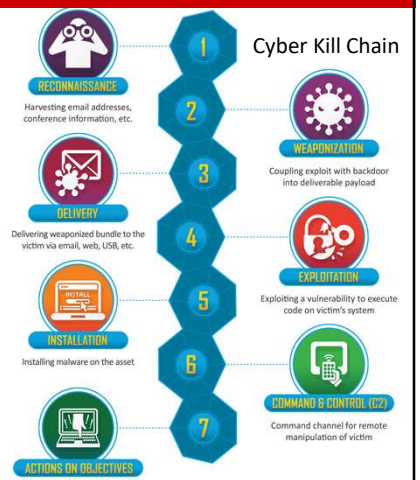
Attacker Tactics

- Reportedly, the attackers
  - used spearphishing to gain access to business networks, deploying the "BlackEnergy" malware
  - stole user credentials from business network
  - used VPN to enter the control-system network
  - abused existing remote access tools to issue malicious commands, switching power substations off
  - uploaded malicious firmware, erased master boot records, and misconfigured UPS to cause further disruption of availability



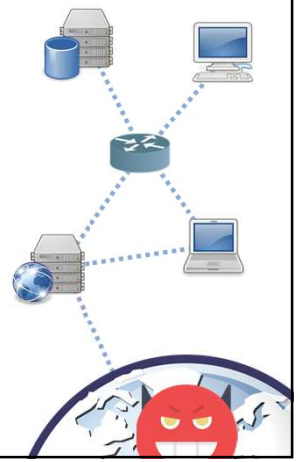
Modeling Intrusions

- Different cyber-attacks may have different goals, use different techniques, etc.
- However, there are some common techniques and approaches
- Cyber Kill Chain®
  - intrusion model developed by Lockheed Martin
- MITRE ATT&CK™
  - knowledge base of adversary tactics and techniques



Source: https://www.lockheedmartin.com/

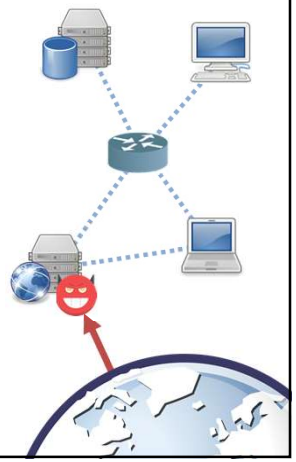
MITRE ATT&CK



Source: https://attack.mitre.org/

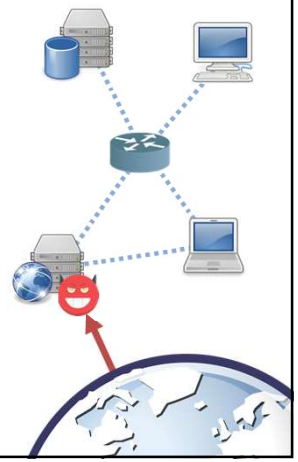
MITRE ATT&CK

1. Initial Access
   - gain initial foothold in target network or system
   - *examples techniques:* spearphishing, drive-by-compromise, exploiting public-facing service, supply chain compromise



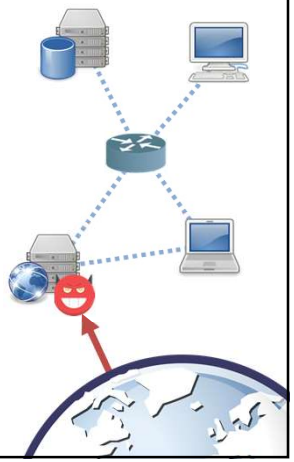
MITRE ATT&CK

1. Initial Access
2. Execution
   - execute malicious code on the system
   - *examples techniques:* command line interface, PowerShell, graphical interface, scripting, execution through API



MITRE ATT&CK

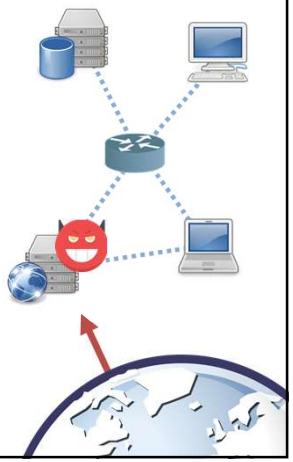
1. Initial Access
2. Execution
3. Persistence
   - maintain access across system or process restarts
   - *examples techniques:* bash profile, browser extensions, account creation, local job or task scheduling, kernel modules



UNIVERSITY of HOUSTON

MITRE ATT&CK

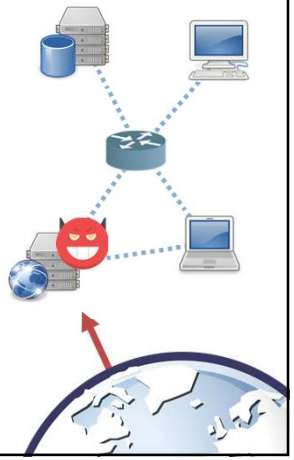
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
   - gain higher-level permissions on the system
   - *examples techniques:* vulnerability exploitation, setuid or setgid abuse, DLL search hijacking, path interception



UNIVERSITY of HOUSTON

MITRE ATT&CK

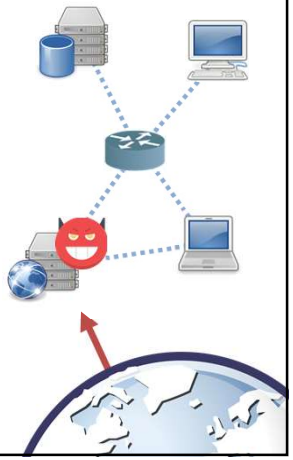
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
   - avoid detection
   - *examples techniques:* disabling security tools, file deletion or obfuscation, rootkits, clearing command history



UNIVERSITY of HOUSTON

MITRE ATT&CK

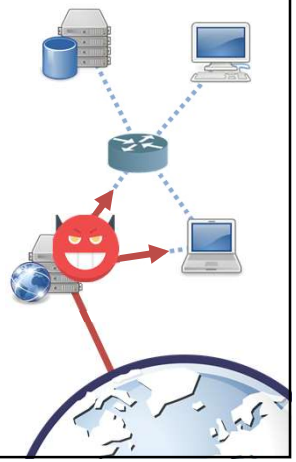
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
   - steal credentials (*e.g.*, usernames and passwords)
   - *examples techniques:* brute forcing, reading bash history and local files, keylogging, network sniffing



UNIVERSITY of HOUSTON

MITRE ATT&CK

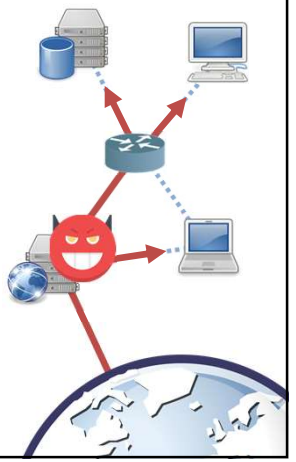
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
   - gain knowledge about the system or internal network
   - *examples techniques:* network service scanning, network sniffing, browser bookmarks, file and directory discovery, user discovery



UNIVERSITY of HOUSTON

MITRE ATT&CK

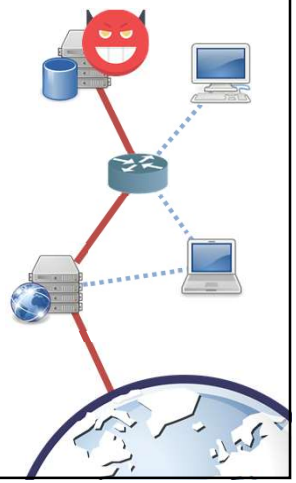
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
   - gain knowledge about the system or internal network
   - *examples techniques:* network service scanning, network sniffing, browser bookmarks, file and directory discovery, user discovery



UNIVERSITY of HOUSTON

MITRE ATT&CK

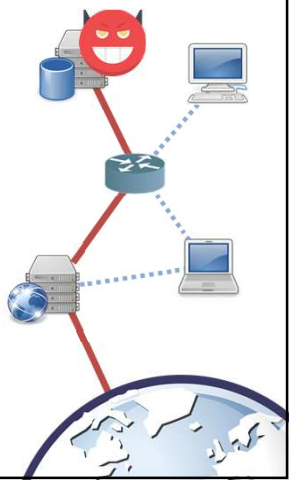
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
8. Lateral Movement
9. Command and Control
   - establish C&C, preferably in a stealthy way
   - *examples techniques:* remote access tools or custom C&C protocols, existing web services, commonly used ports, data obfuscation



UNIVERSITY of HOUSTON

MITRE ATT&CK

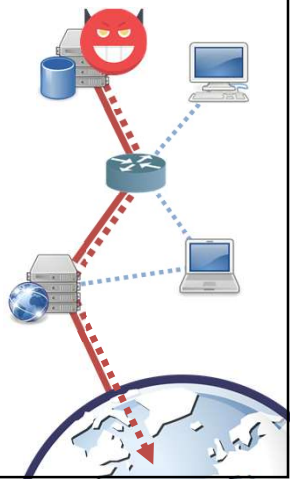
1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
8. Lateral Movement
9. Command and Control
10. Collection
    - gather sensitive information
    - *examples techniques:* audio or screen capture, keylogging, local files, shared network drives



UNIVERSITY of HOUSTON

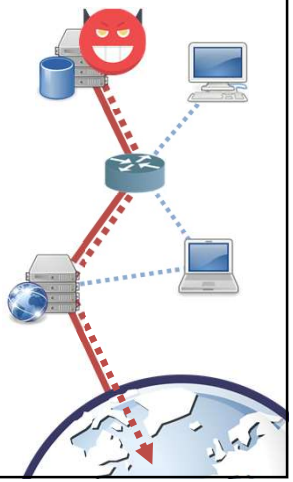
MITRE ATT&CK

1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
8. Lateral Movement
9. Command and Control
10. Collection
11. Exfiltration
    - steal sensitive data
    - *examples techniques:* transfer through cloud accounts or alternative protocols, encrypted or compressed data transfer

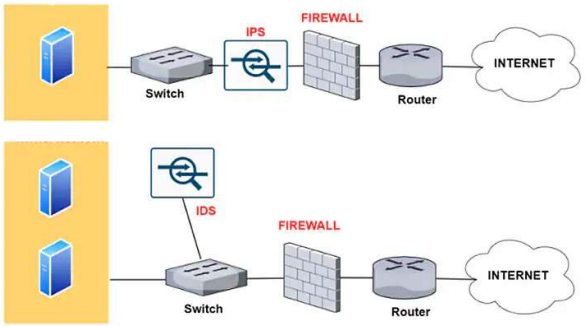


MITRE ATT&CK

1. Initial Access
2. Execution
3. Persistence
4. Privilege Escalation
5. Defense Evasion
6. Credential Access
7. Discovery
8. Lateral Movement
9. Command and Control
10. Collection
11. Exfiltration
12. Impact
    - disrupt availability or compromise integrity
    - *example techniques:* shutdown, wiping or encrypting data, firmware corruption, defacement



2. Intrusion Detection Systems



Stealthy Attacks

- Attackers often aim to keep compromises covert in order to maximize impact. Examples
  - cyber-espionage: attacker can spy on the victim only as long as the victim is unaware.
  - botnets: attacker can remotely control and benefit from compromised computers as long as owners are unaware.
- In practice, compromises may remain covert for long periods of time
  - FirEye M-Trends: median time to detect compromise was **24 days** in 2020.
- Detection of attacks is crucial for minimizing cybersecurity risks.
  - timely mitigation can reduce impact.

Intrusion Detection Systems

- Intrusion Detection System (IDS): application or device that monitors a network or system for malicious activity
  - malicious activity is reported to administrators (e.g., send an alarm, log activity)
- Classification by the monitoring location
  - **Network**-based IDS: Monitor network traffic
  - **Host**-based IDS: Monitor activities on a host

Characteristics

- Scope
  - Host-based,
  - Multi-host based,
  - Network-based
- Operation
  - Off-line,
  - Real-time
- Types of errors
  - false positive (i.e., false alarm): wasting system administrators' time/effort
  - false negative: undetected attack

Terminologies

		True Class	
		Pos	Neg
Hypothesized Class	Yes	True Positives (TP)	False Positives (FP)
	No	False Negatives (FN)	True Negatives (TN)
		P	N

$$TP\ \ Rate = \frac{TP}{P} = \text{recall}$$

$$FP\ \ Rate = \frac{FP}{N}$$

$$TN\ \ Rate = \frac{TN}{N}$$

$$FN\ \ Rate = \frac{FN}{P}$$

Accuracy

		True Class	
		Pos	Neg
Hypothesized Class	Yes	1 (TP)	0 (FP)
	No	1 (FN)	988 (TN)
		P	N

$$Accuracy = \frac{TP + TN}{P + N}$$

$$= \frac{1+998}{2+998}$$

$$= 99.9\%$$

By mixing Positives and Negatives in one measure, we cannot get a correct picture of the test.

We are failing to detect every other patient.

Recall

		True Class: Pos	True Class: Neg
Hypothesized Class	Yes	1 (TP)	0 (FP)
	No	1 (FN)	998 (TN)
		P	N

$$Recall = \frac{TP}{TP + FN}$$

$$= \frac{1}{1+1}$$

$$= 50\%$$

Recall shall be the model metric we use to select our best model when there is a high cost associated with False Negative

Detection Models

- Signature-based (Misuse Detection): recognize known attacks
  - Define a set of attack signatures
  - Detect actions that match a signature
  - Can detect known attacks only, add new signatures often
- Anomaly-Based: recognize atypical behavior
  - Define a set of metrics for the system
  - Build a statistical model for those metrics during "normal" operation
  - Detect when metrics differ significantly from normal
- Hybrid
  - Examples: CMDS, DIDS, EMERALD, INBOUNDS, NIDES, RealSecure

Snort

- Intrusion Prevention System (IPS)
  - also called Intrusion Detection and Prevention System (IDPS)
  - can actively prevent or block intrusions (e.g., block IP addresses, drop packets)
- Snort is a free and open-source network intrusion detection system
  - can detect a variety of attacks based on signatures
  - can be extended with custom rules and plug-ins
- Currently owned and developed by Cisco
- High-level components

read packets from network interface (or a file)

verify that packets are constructed according to specification

defragment packets, decode HTTP URI, re-assemble TCP streams, etc.

apply rules to each packet

generate final output

Snort Rule Type

- **Alert** rules: Snort generates an alert when a suspicious packet is detected.
- **Block** rules: Snort blocks the suspicious packet and all subsequent packets in the network flow.
- **Drop** rules: Snort drops the packet as soon as the alert is generated.
- **Logging** rules: Snort logs the packet as soon as the alert is generated.
- **Pass** rules: Snort ignores the suspicious packet and marks it as passed.

Snort Detection Rules

- Rule example:

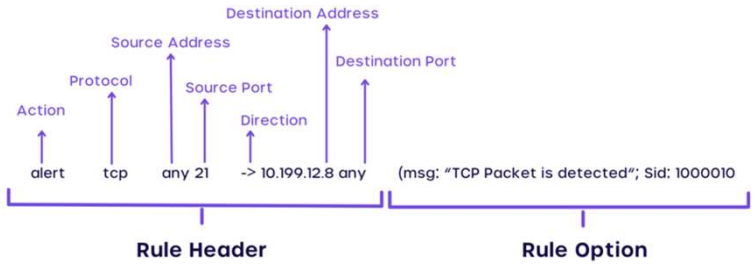
```
alert tcp any any -> 192.0.2.1 80 (msg: "Connect to webserver"; flags: S;)
```

- Rule header
  - Rule Action: what action to take when the rule matches (*e.g.*, `alert` or `pass`)
  - Protocol: IP, ICMP, TCP, UDP, …
  - Source IP and Port: single or multiple hosts / network addresses
  - Flow: ->
  - Destination IP and Port: TCP or UDP ports (or port ranges)
- Rule options
  - list of **`keyword: argument`** pairs, separated by `;` and enclosed in `( )`
  - may define constraints on header fields or payload contents

More Examples

```
alert tcp 192.168.1.0/24 any -> 131.171.127.1 25
   (content: "hacking"; msg: "malicious packet"; sid:2000001;)

alert tcp any 21 -> 10.199.12.8 any
   (msg:"TCP Packet is detected"; Sdi:1000010
```



https://cyvatar.ai/write-configure-snort-rules/

Rule Examples

- Directory traversal attack
  - *example file-inclusion exploit:* `GET /index.php?file=../../etc/passwd`
  - detection rule:

```
drop tcp any any -> $WEBSERVER 80 (msg: "Directory
traversal";
  content: "../"; http_uri;)
```

    - `$WEBSERVER`: address defined in the configuration file
    - `content` with `http_uri`: search for the argument of `content` in the HTTP URI
- Spam sent from compromised computers
  - botnets of compromised computers are often used to send spam e-mail
  - detection rule:

```
alert tcp !$SMTP_SERVERS any - !$SMTP_SERVERS 25 (msg:
"Botnet
  spam"; flags: A+;)
```

    - `!`: negation operator for address
    - `flags: A+`: that the ACK and at least one other TCP flag is set

Anomaly-Based Detection

- Disadvantages of signature-based detection
  - detects only known attacks (and basic traffic anomalies)
  - large number of signatures (thousands or more) → expensive
- Anomaly-based detection
  - characterize normal traffic or system behavior → raise alarm for anything "abnormal"
  - normal operations can be characterized
    - using AI / machine learning from training data
    - by expert from domain knowledge
  - *example:* monitor short sequences of system calls
    - if a previously unseen sequence is observed → raise alarm

Challenges in Anomaly-Based Detection

- Training data
  - abundant data for normal behavior
  - very little data on abnormal behavior (i.e., attacks)
- Modeling system behavior
  - computationally tractable and correct representation of normal behavior
- Large number of false positive errors
  - unusual but non-malicious activity may be detected as an attack

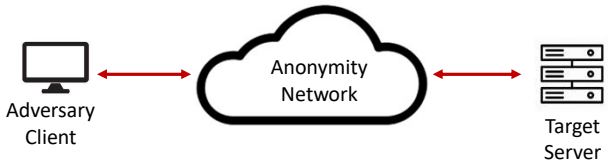
3. Anonymity Networks

- An anonymity network protects users' identity and privacy while using the internet.
- These networks employ a sophisticated system, routing data through a complex series of nodes and using multiple layers of encryption to effectively mask a user's IP address and other identifying information.
- Unfortunately, these networks may also be used to hide the identity of the intruders.

Remote Access

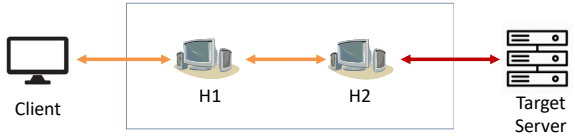
Secure Shell (SSH) protocol allows a user to access a server remotely.

Hiding Identity



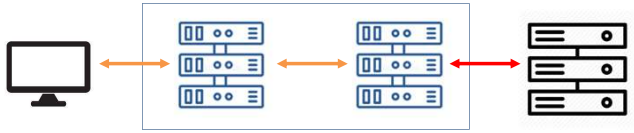
An anonymity network protects the identity of the client.

Ex 1: Stepping-Stone Network

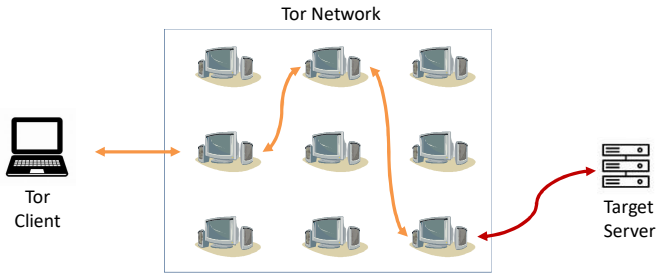


Build your own network with compromised hosts

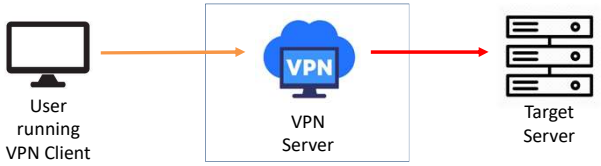
Ex 2: Anonymity Network: Proxy



Ex 3: Anonymity Network: TOR



Ex 4: Anonymity Network: VPN



Next Topic

- Intrusion Detection
- Isolation
- Denial of Service