



Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Report created by Matej Tinkovic

Overview of Engagement

The purpose of this engagement is to use a Kali Linux machine to look for risks and exploit critical vulnerabilities on the Capstone target machine in order to get a better insight into different ways that attackers can harm this machine from a Red team perspective. Then, from a blue team perspective, analysis as well as mitigations will be provided as part of recommended security measures as to how these vulnerabilities can be fixed by hardening the Capstone machine and preventing future attacks.

The scope of this engagement is to discover and exploit vulnerabilities **ONLY** on the Capstone machine with IP address 192.168.1.105. There is otherwise no limit in terms of what files, folders can be accessed on that machine or in which ways attacks can be attempted on the machine.

Table of Contents

This document contains the following sections:

01

Network Topology

02

Red Team: Security Assessment

03

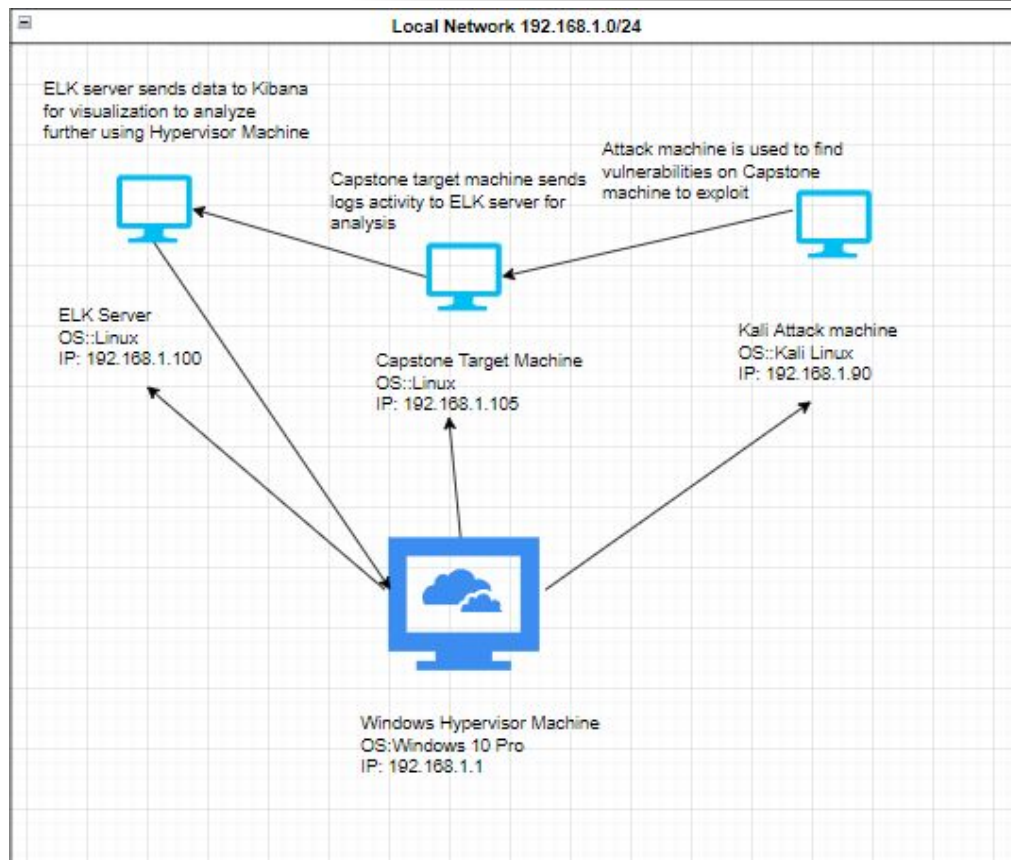
Blue Team: Log Analysis and Attack Characterization

04

Hardening: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1


Machines

IPv4: 192.168.1.1
OS: Windows 10
Hostname:
ML-RefVm-684427

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: server1

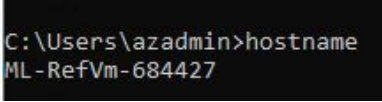
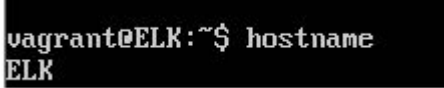
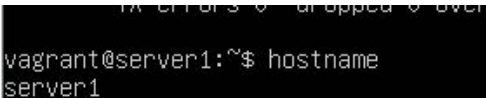
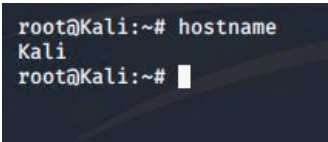
IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

The background of the slide is a dark red, almost black, geometric pattern composed of numerous overlapping triangles and polygons, creating a complex, crystalline texture.

Red Team Security Assessment

Recon: Describing the Target

Nmap identified the following live hosts on the network (256 IP addresses were scanned in total on the network):

Hostname	IP Address	Role on Network
ML-RefVm-684427 	192.168.1.1	Windows Hypervisor Machine (NAT Switch)
ELK 	192.168.1.100	ELK Server
server1 	192.168.1.105	Main target machine with vulnerabilities to exploit
Kali 	192.168.1.90	Attacker machine used to hack into vulnerable machine in network

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Open ports 80 and 22 on the target Capstone machine	Ports 80 and 22 are open with unsecured access for any user	Unsecured remote access to target machine. Also sensitive files and folders can be accessed.
Web server directory index vulnerability (Sensitive data exposure)	Users are able to see index directory and files on the server.	This is a serious vulnerability as the index pages should never be publicly accessible. This exposes the sensitive files on the server to unauthorized access and file exfiltration
Webdav protocol access vulnerability	Users are able to connect to the webdav of the server through their local machine and upload files (like .php exploit file we uploaded)	This breaks the authentication and identification category of OWASP top 10 as only authorized admin users should be able to access the server
Hidden directories vulnerability	Users are able to find hidden directories like the secret_folder directory on the target machine	This breaks the broken access control and Insecure design OWASP top 10 categories

Exploitation: Web Server directory index vulnerability

Tools & Processes

How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use?

After identifying the IP address of the target host using nmap -sn 192.168.1.0/24, I searched the IP in the URL of a browser and discovered that the directory index pages are publicly accessible

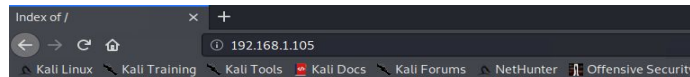
```
Network Distance: 1 hop
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

```
TRACEROUTE
HOP RTT ADDRESS
1 0.87 ms 192.168.1.105
```

Achievements

What did the exploit achieve?
For example: Did it grant you a user shell, root access, etc.?

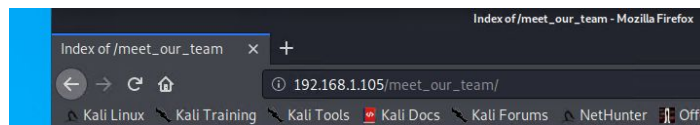
The exploit is the ability to access the web server index directory itself. This should NEVER be allowed as it exposes sensitive data on the server to unauthorized external access. It even exposes the server version and open port(s)



Index of /

Name	Last modified	Size	Description
company_blog/	2019-05-07 18:23	-	
company_folders/	2019-05-07 18:27	-	
company_share/	2019-05-07 18:22	-	
meet_our_team/	2019-05-07 18:34	-	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80



Index of /meet_our_team

Name	Last modified	Size	Description
Parent Directory		-	
ashton.txt	2019-05-07 18:31	329	
hannah.txt	2019-05-07 18:33	404	
ryan.txt	2019-05-07 18:34	227	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

Exploitation: Hidden directories vulnerability

Tools & Processes

How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use?

After accessing the directory index pages for the target machine server, there were several files there referencing a “secret-folder” directory that is not actually visible in the index. After searching for this directory in the URL, I was able to gain access to this directory. Using hydra tool I was able to brute force my way into the directory and see the contents.

Achievements

What did the exploit achieve?
For example: Did it grant you a user shell, root access, etc.?

The fact that users are able to access this hidden directory that has specific instructions regarding how to access webdav is a major vulnerability as this allows ANY external user to connect to the company’s webdav for example but also it means the server is vulnerable to directory traversal attacks as well.

Command: `hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get 192.168.1.105/company_folders/secret_folder/.`

I know ashton was the user because after looking around in the company directories I found a file stating that she is in charge of credit card info located in that directory.

```
f 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 15] (0/0)
[80][http-get] host: 192.168.1.105 login: ashton password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-05-14 0
8:53:03
root@Kali:/usr/share/wordlists#
```



Exploitation: Webdav access vulnerability

Tools & Processes

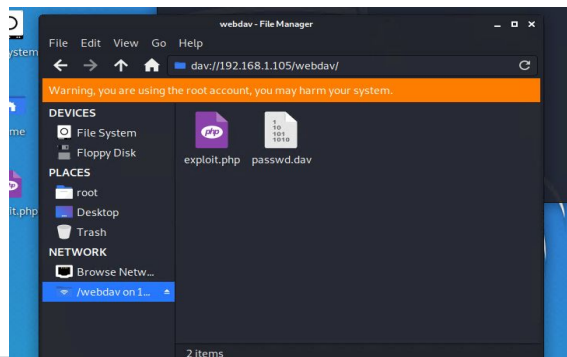
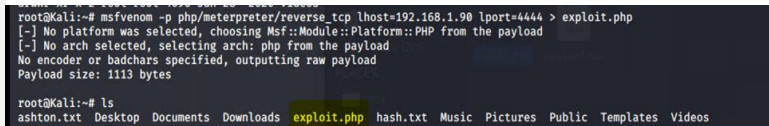
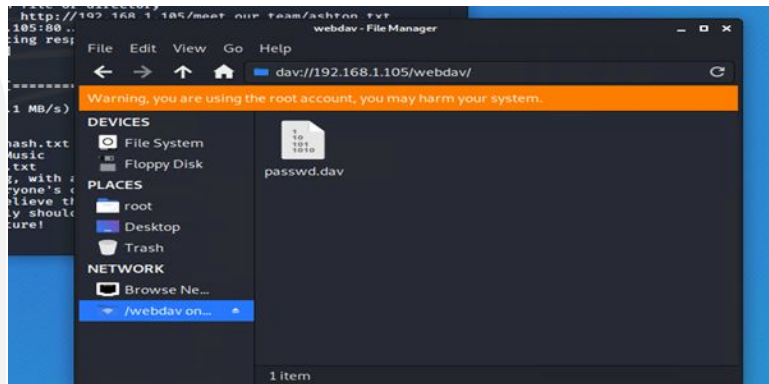
How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use?

After accessing the secret_folder directory, I followed the instructions in the connect_to_corp_server file to connect to the webdav protocol of the server through my local Kali machine as per the screenshot on the top right.

Achievements

What did the exploit achieve? For example: Did it grant you a user shell, root access, etc.?

By gaining access to webdav, I was then able to upload any file I wanted to the server. In this case, I created a PHP reverse shell payload using msfvenom and then simply uploaded the file to the web server as per the screenshots to the right. This is an example of unauthorized file upload as well as remote code execution.



Exploitation: Port 80 and 22 open to public access

Tools & Processes

How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use?

After using nmap to identify the IP address for the Capstone machine, I the used the same tool (specifically `nmap -sV 192.168.1.105`) to identify any open ports and running services associated on the machine and discovered that ports 80 and 22 are open for public access


Achievements

What did the exploit achieve?
For example: Did it grant you a user shell, root access, etc.?

This means that external users could connect to the machine through either of these ports especially port 22 where they can establish a remote connection, but also malicious payloads can be uploaded to the server through port 80 as well such as the `exploit.php` payload discussed in the previous slide.

```
root@Kali:~# nmap -sV 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-17 15:32 PDT
Nmap scan report for 192.168.1.105
Host is up (0.0068s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 7.10 seconds
root@Kali:~#
```



Blue Team

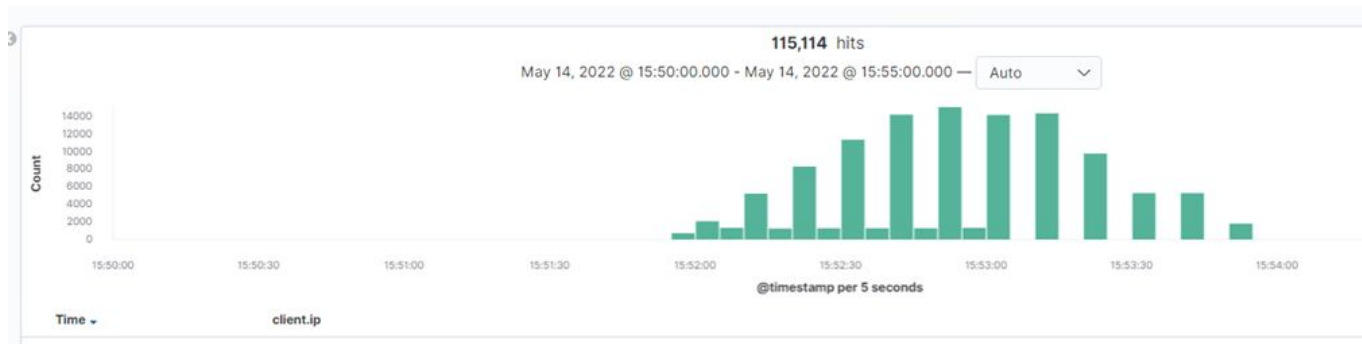
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.



- What time did the port scan occur?
- How many packets were sent, and from which IP?
- What indicates that this was a port scan?



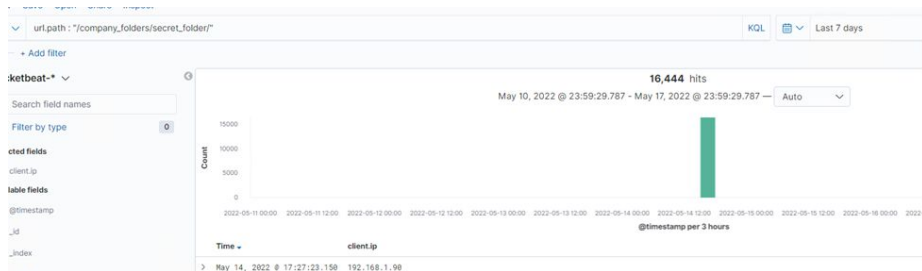
- Port scan occurred on May 14th, 2022 between 15:50 and 15:55
- 115,114 packets from source_IP 192.168.1.90?
- There is an abnormal amount of traffic coming from the same source_IP 192.168.1.90 to the web server IP in a very short time period over port 80 indicating a potential port scan

Analysis: Finding the Request for the Hidden Directory

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.



- What time did the request occur? How many requests were made?
- Which files were requested? What did they contain?



```
t network.transport
t network.type
t query
# server.bytes
# server.ip
# server.port
# source.bytes
```

```
tcp
ipv4
GET /company_folders/secret_folder/connect_to_corp_server
674B
192.168.1.105
80
470B
```

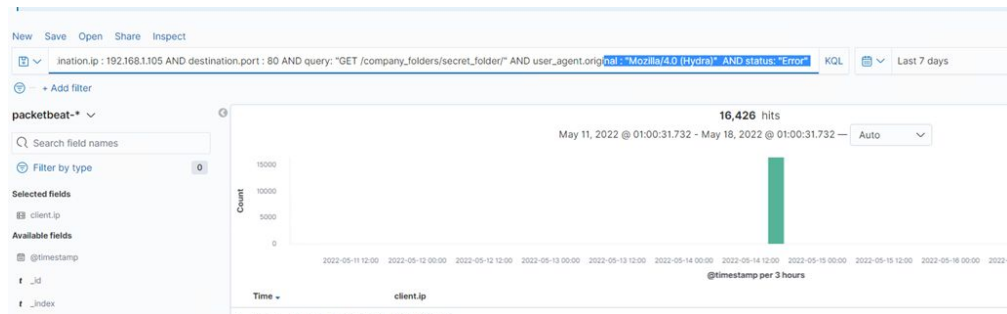
- There were 16444 http get requests to this directory between 15:51:55 and 15:53 from the same IP address 192.168.1.90.
- The secret folder directory was requested which contained the instructions for how to obtain access to the corp webdav server in the connect_to_corp_server file. It also contained the hashed password for the CEO that I could crack using crackstation and use to access the webdav server later

Analysis: Uncovering the Brute Force Attack

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.



- How many requests were made in the attack?
- How many requests had been made before the attacker discovered the password?



- 16428 HTTP GET requests in total with the GET /company_folders/secret_folder/ query and from user agent Mozilla 4.0/(Hydra)
- There were 16426 requests with status marked as “error” meaning the credentials were not correct.

Analysis: Finding the WebDAV Connection



Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.



- How many requests were made to this directory?
- Which files were requested?

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending ▾	Count ▾
http://192.168.1.105/company_folders/secret_folder/	16,464
http://127.0.0.1/server-status?auto=	5,280
http://192.168.1.105/webdav/exploit.php	400
http://192.168.1.105/webdav	234
http://snnmnkxdhflwgthqismb.com/post.php	112

Export: [Raw](#)  [Formatted](#) 

- 234 requests to the /WebDav directory
- Exploit.php

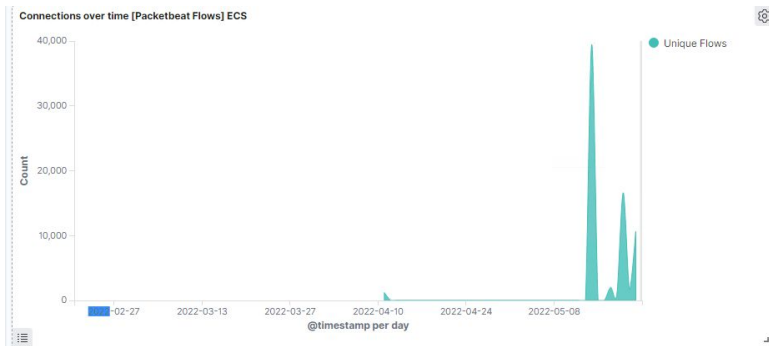


Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm



Search criteria:

destination.ip 192.168.1.105 and source.ip: (not 192.168.1.105) and destination.port: (not 443 or 80)

Report criteria:

Number of ports accessed per source IP per second.

Alarm criteria/threshold:

Alert email and log when > 3 none port 443 or port 80 scans detected at the same timestamp from the same IP occur

System Hardening

What configurations can be set on the host to mitigate port scans?

Describe the solution. If possible, provide required command lines.

Set up firewall to drop packets when the threshold of packet traffic is exceeded

This can be done by enabling filters 7000 to 7004 and 7016 as these filters have threshold values that can be configured and then the filter becomes active when the number of connection attempts from a source IP address exceeds the threshold. Host scans and port sweeps are blocked through the Quarantine feature.

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?

What threshold would you set to activate this alarm?

Search criteria:

source.ip: (not 192.168.1.105 or 192.168.1.1) and
destination.ip:192.168.1.105 and url.path:

secret_folder

Report criteria:

Number of times “secret_folder” accessed from
external IP

Alarm criteria/threshold:

Alert email and log when > 0 access is detected on
“secret_folder” from IPs other than 192.168.1.105 or
192.168.1.1

System Hardening

What configuration can be set on the host to block unwanted access?

- Improve the password for accessing this directory (if it really must be accessible by certain users)
- Blacklist source IP 192.168.1.90 with a firewall rule
- Remove the directory. Why is a secret folder directory needed on a web server? This information should be stored on a standalone and much more secure web server if it is truly of secret and of utmost importance.
- Files and folders should be encrypted.

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future brute force attacks?

What threshold would you set to activate this alarm?

Search criteria:

source.ip: (not 192.168.1.105 or 192.168.1.1) and
destination.ip: 192.168.1.105 and user.agent:

Mozilla 4.0/(Hydra)

Report criteria:

Number of http requests coming from user.agent:

Mozilla 4.0/(Hydra)

Alarm criteria/threshold:

Alert email and log when > 0 http requests is
detected from user.agent: Mozilla 4.0/(Hydra)

from IPs other than 192.168.1.105 or 192.168.1.1

System Hardening

What configuration can be set on the host to block brute force attacks?

Describe the solution. If possible, provide the required command line(s).

- Complex and long passwords to make it harder for brute force attacks to succeed
- Blacklist IPs associated with brute force attacks. For example, if there is an IP with an x amount of unsuccessful attempts in the past x months.
- Lockout accounts for 45 mins after 5 unsuccessful attempts

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory?

What threshold would you set to activate this alarm?

Whitelist specific machines IPs that are allowed access. Alarm should be set for any machine IP trying to access webdav (through HTTP GET request) that is not on the whitelist.

System Hardening

What configuration can be set on the host to control access?

Describe the solution. If possible, provide the required command line(s).

Whitelist machines that are allowed access to webdav on web server firewall and block all other IP addresses.

Let's assume we'd like to allow incoming traffic to our server machine from machine with IP 192.168.0.1 for example.

The run this command:

```
iptables -A INPUT -s 192.168.0.1 -p tcp --dport 80 22 -i eth0  
-j ACCEPT
```

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads?

What threshold would you set to activate this alarm?

Search criteria:

source.ip: (not 192.168.1.105 or 192.168.1.1) and
destination.ip: 192.168.1.105 and source.port: 4444

Report criteria:

Number of http requests coming from any external IP from source port 4444

Alarm criteria/threshold:

Alert email and log when > 0 http requests is detected from source port 4444

from IPs other than 192.168.1.105 or 192.168.1.1

System Hardening

What configuration can be set on the host to block file uploads?

Describe the solution. If possible, provide the required command line.

- Using tools like Antipwmy or Antimeter. These tools help find and kill the meterpreter session that the attacker has gained.
- Run applications/processes on server with least privileges principle. Access to webdav folder should be read only by default except if an admin is accessing
- Limit network access to only trusted hosts
- Firewall rule to block packets with source port 4444

Assessment Summary

During this engagement we have achieved the following:

- Used nmap to scan the local network and discover live hosts IP addresses on the network and ultimately identify the Capstone target machine as well as ports/services running on each host
- Discovered 4 vulnerabilities on the Capstone machine and provided an explanation for each regarding which tools & processes were used to exploit the vulnerability as well as the impact that is caused by these exploits
- Log Analysis outlining the attacker's activity on the host network was explained and the attacks themselves were identified using data and information from logs in Kibana. It was discovered that the attacker accessed the machine through HTTP port 80.
- Various types of alarms were provided that could help the security team managing the Capstone machine to identify future potential attack attempts on the web server and investigate accordingly
- Finally, mitigations were provided for each attack activity to advise how the system can be configured to harden the Capstone machine and make it more difficult for attackers to succeed with their malicious activities

The background is a dark blue field filled with a complex, repeating geometric pattern of triangles. The triangles are in various shades of blue, creating a subtle gradient and a textured effect. The pattern is composed of many small triangles that interlock to form larger, irregular shapes.

THANK YOU