

# Capstone Engagement

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

Tyler Windes

# 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

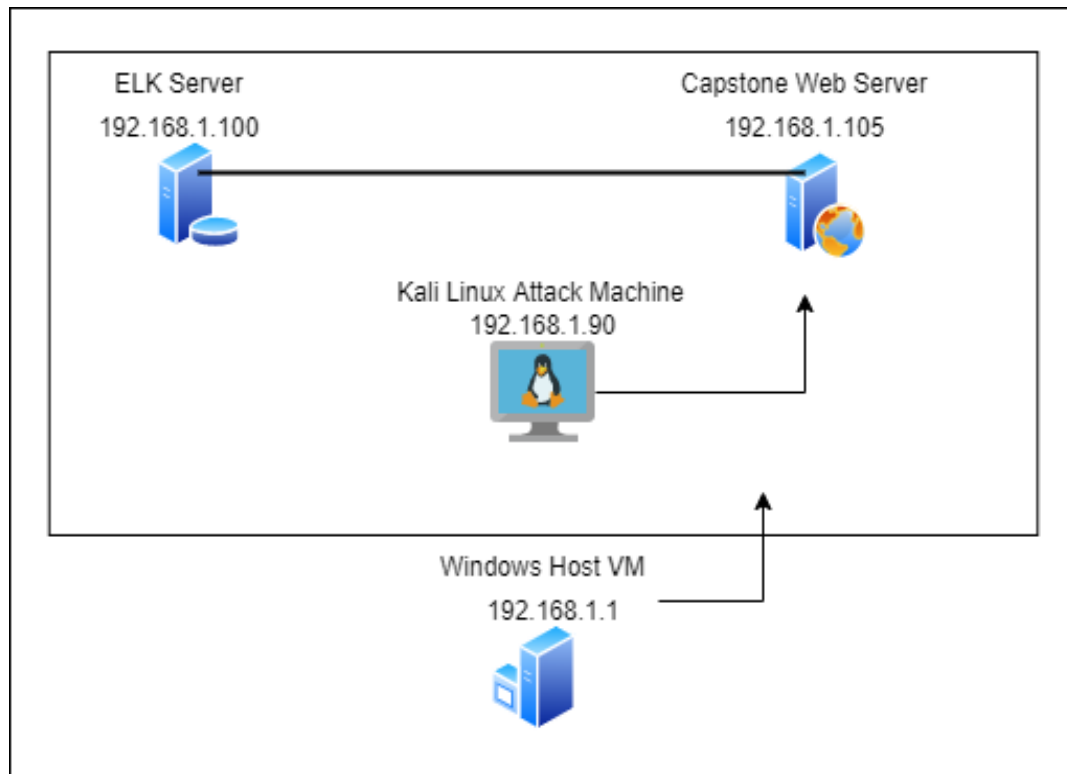
**Hardening:** Proposed Alarms and Mitigation Strategies

---

# Network Topology

# Network Topology

192.168.1.0/24



## 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

IPv4:192.168.1.105  
OS: Linux  
Hostname: Capstone

IPv4:192.168.1.90  
OS: Linux  
Hostname: Kali

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 squares, creating a mosaic-like effect.

# **Red Team** Security Assessment

# Recon: Describing the Target

---

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Windows Host VM	192.168.1.1	Host VM
Capstone VM	192.168.1.105	Vulnerable Target VM
ELK	192.168.100	Network Monitoring and Logging VM
Kali	192.168.1.90	Attacking VM

---

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Weak Passwords	<i>By requiring users to create more complex passwords, brute force attacks become less effective and easier to crack.</i>	<i>This allowed a software program to try multiple passwords using a wordlist (bruteforce) to gain access to the network.</i>
Unsecured SSH	By allowing ssh traffic from any IP address opens several vulnerabilities to the server. Using a range of tools, attackers can gain access to a shell.	This vulnerability allowed me to gain access to their server and upload a reverse shell payload. This allowed me to run commands on the system.
Open Ports (internet facing)	This can allow for malicious actors to use a variety of exploits (depending on the target system).	Depending on the exploit, sensitive data can then be downloaded, encrypted, or deleted. Also, some commands may be able to be executed.
DDOS Attacks	Due to a known exploit in the version of Windows running the web server, it is vulnerable to denial-of-service attacks affecting server availability.	By updating/upgrading to a newer version of Windows webserver should mitigate this vulnerability.

# Exploitation: Weak Passwords

---

01

## Tools & Processes

Accessing the vulnerable website, I used Hydra to brute force Ashton's password (jeopoldo).

02

## Achievements

After gaining access via Aston's account, I discovered in the company's "secret\_folder" directons left for himself with instructions to connect to webdav. It expains how to use Ryan's account to gain access. Included is Ryan's hashed password, which was easily cracked.

03

Screenshot of Hydra Brute Force:

[https://drive.google.com/file/d/1VGoZCubdoufKka2NWRGkld88iv4\\_iJD\\_/view?usp=sharing](https://drive.google.com/file/d/1VGoZCubdoufKka2NWRGkld88iv4_iJD_/view?usp=sharing)

Screenshot of Ashton's Directions to webdav:

[https://drive.google.com/file/d/1fjg1\\_ho2RvgZ1DVIbsdyZ1ebf\\_gllANG/view?usp=sharing](https://drive.google.com/file/d/1fjg1_ho2RvgZ1DVIbsdyZ1ebf_gllANG/view?usp=sharing)



# Exploitation: Unsecured SSH

---

01

## Tools & Processes

Using an Nmap scan, it was revealed several ports were open, including port 22.

02

## Achievements

I was able to connect to the server using ssh and Ryan's credentials. This opened a shell and allowed me to navigate through the system.

03

Screenshot of Nmap scan showing open ports:

[https://drive.google.com/file/d/1g380p8mTnwU4Nolx\\_UdzQQewReVmmmJg/view?usp=sharing](https://drive.google.com/file/d/1g380p8mTnwU4Nolx_UdzQQewReVmmmJg/view?usp=sharing)

Screenshot showing Ryan's shell via ssh:

<https://drive.google.com/file/d/1cVsaZ2SYDRrP98Nn9a71I9HgYQDhSqwu/view?usp=sharing>

# Exploitation: Open Ports (internet facing)

---

01

## Tools & Processes

Again, using an Nmap scan, it was revealed several ports were open, including port 22. Exploiting this, using msfvenom I created a payload to initiate a reverse meterpreter shell. The payload was uploaded to the target server using the access gained previously. The specific exploit used was "multi/handler".

02

## Achievements

Using Metasploit's msfconsole, I created a listening port on the attacking machine. I then executed the payload (backdoor\_shell.php) via web browser. This triggered the attacking machine to open a reverse metepreter shell. This was then used to navigate throughout the system.

03

Screenshot of Venom payload created:


<https://drive.google.com/file/d/1ps91lj2VrtOGkh7kPVFB04TzGR7FyUyz/view?usp=sharing>

Screenshot of payload on target machine:

[https://drive.google.com/file/d/1xcLuzDRXTNN9hH\\_JjYN1WmeVqnNVy5Dr/view?usp=sharing](https://drive.google.com/file/d/1xcLuzDRXTNN9hH_JjYN1WmeVqnNVy5Dr/view?usp=sharing)

Screenshot of open Meterpreter shell:

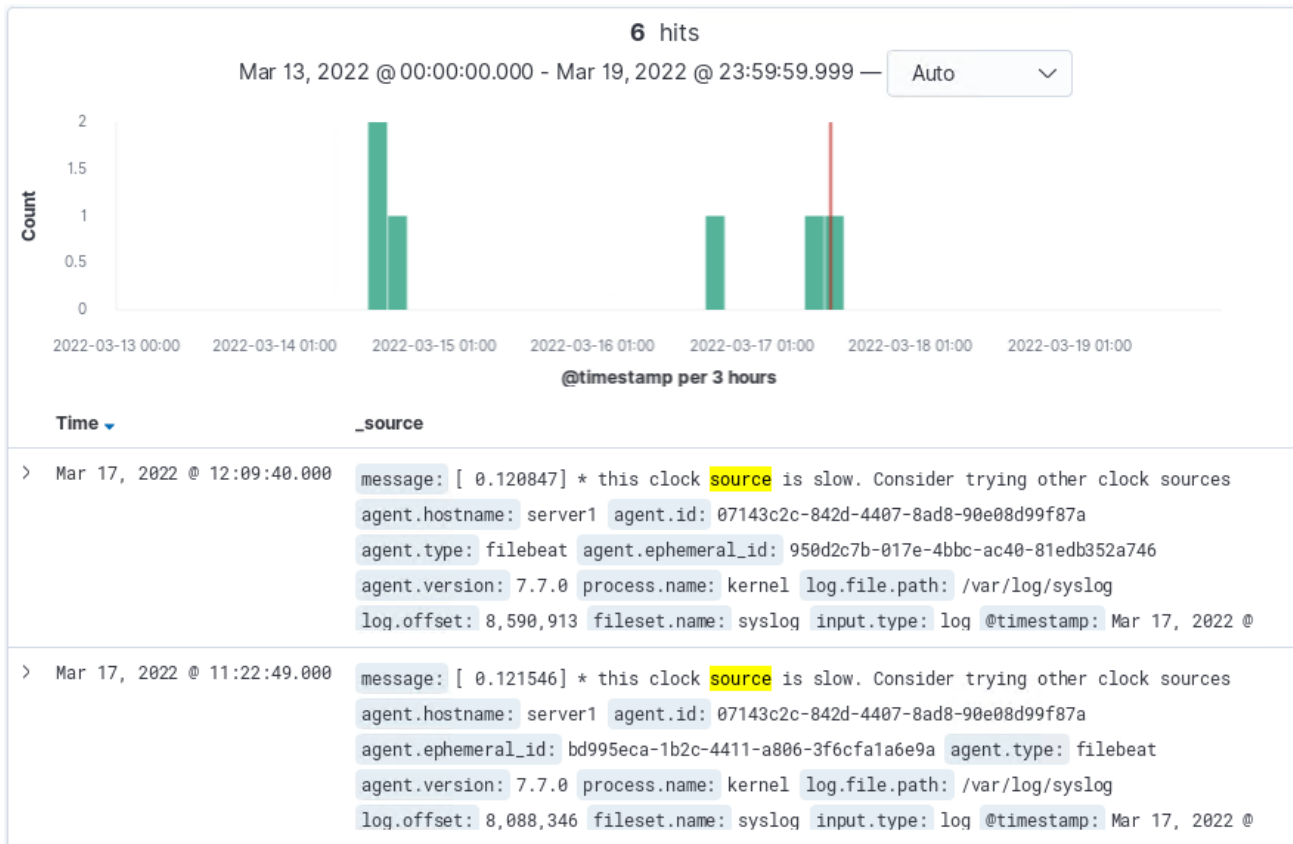
<https://drive.google.com/file/d/1Y6v6szakSFgvm16wJzlsbDySPmcPtUk9/view?usp=sharing>



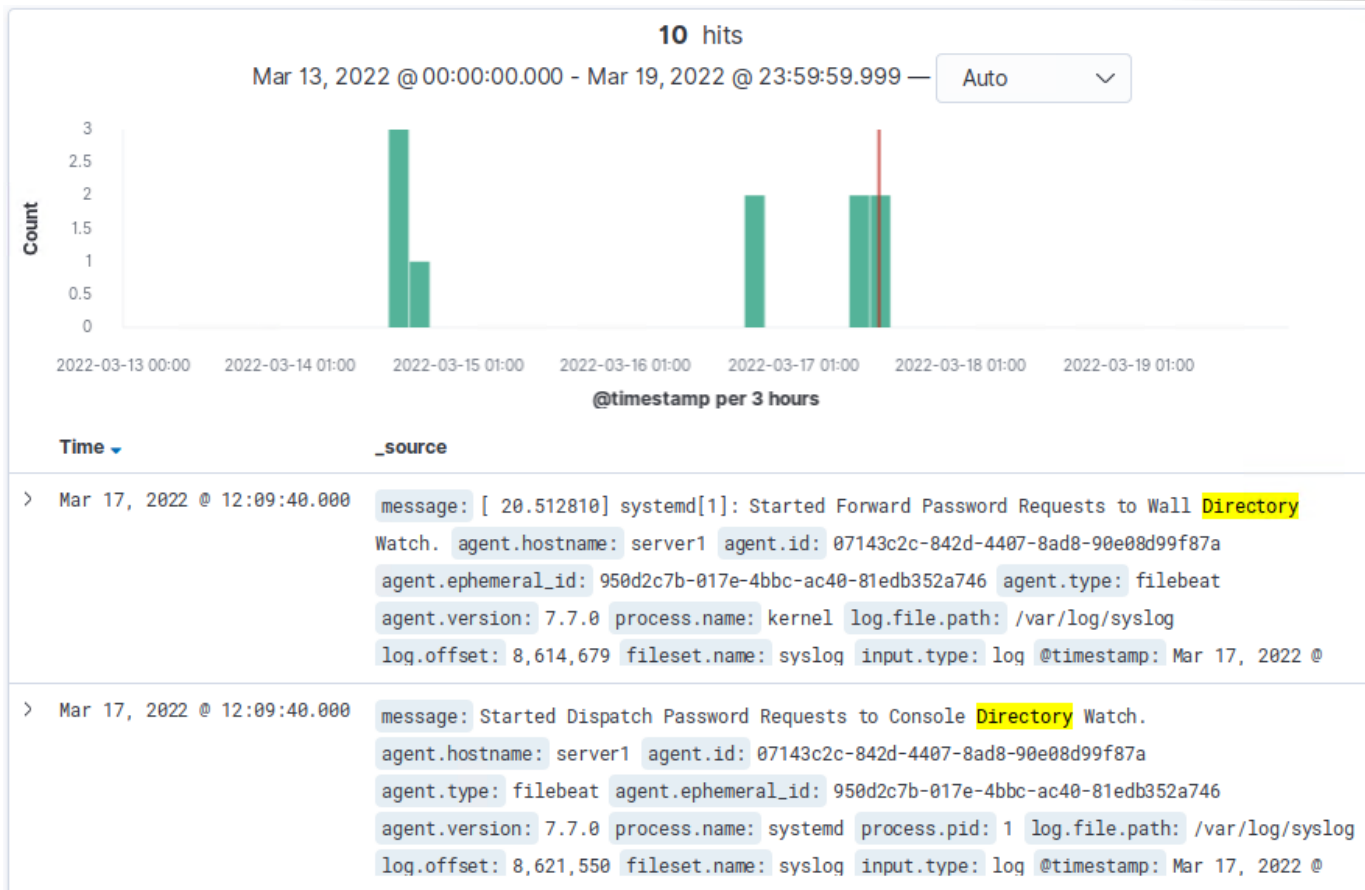
# **Blue Team**

## Log Analysis and Attack Characterization

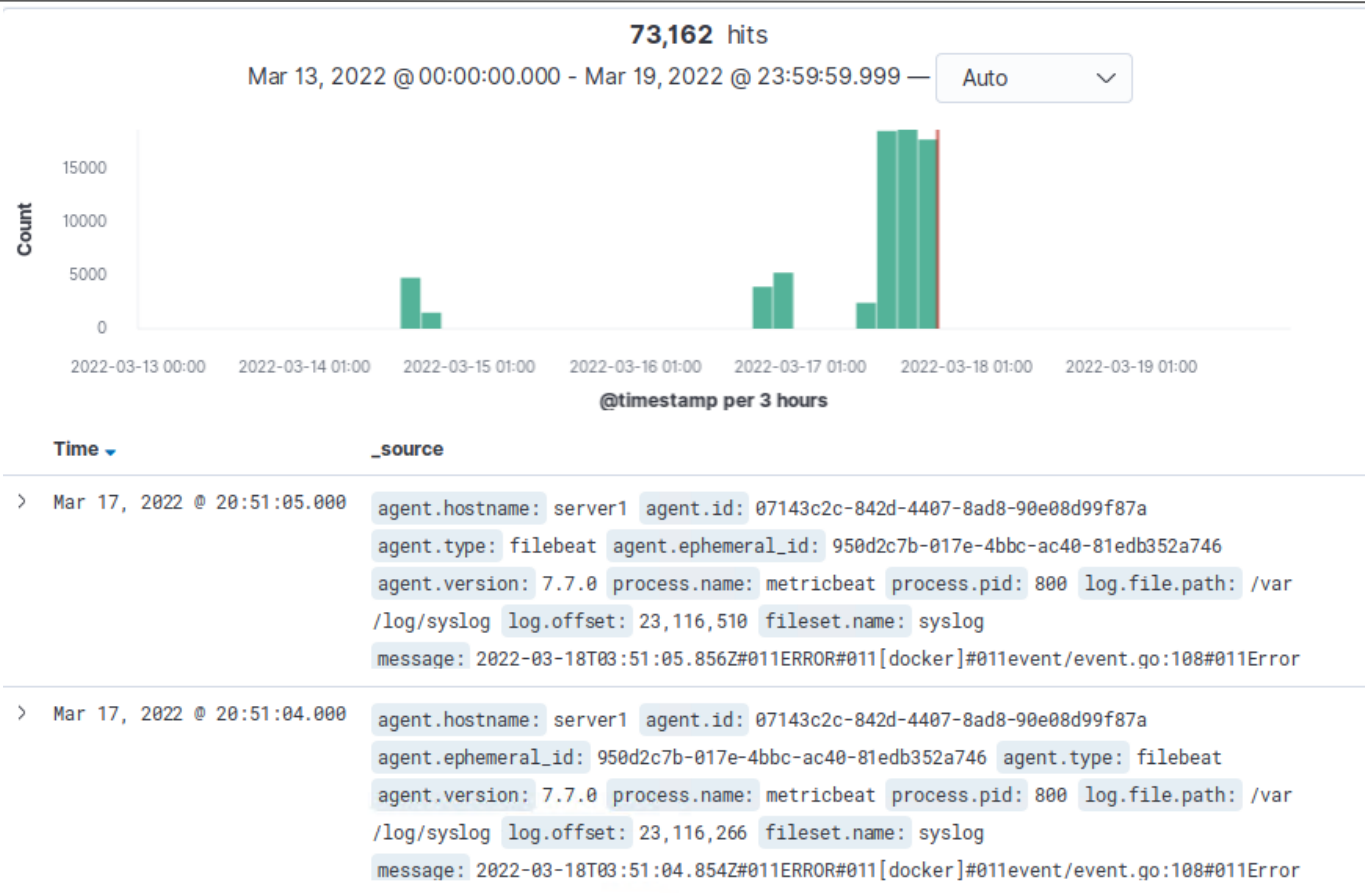
# Analysis: Identifying the Port Scan



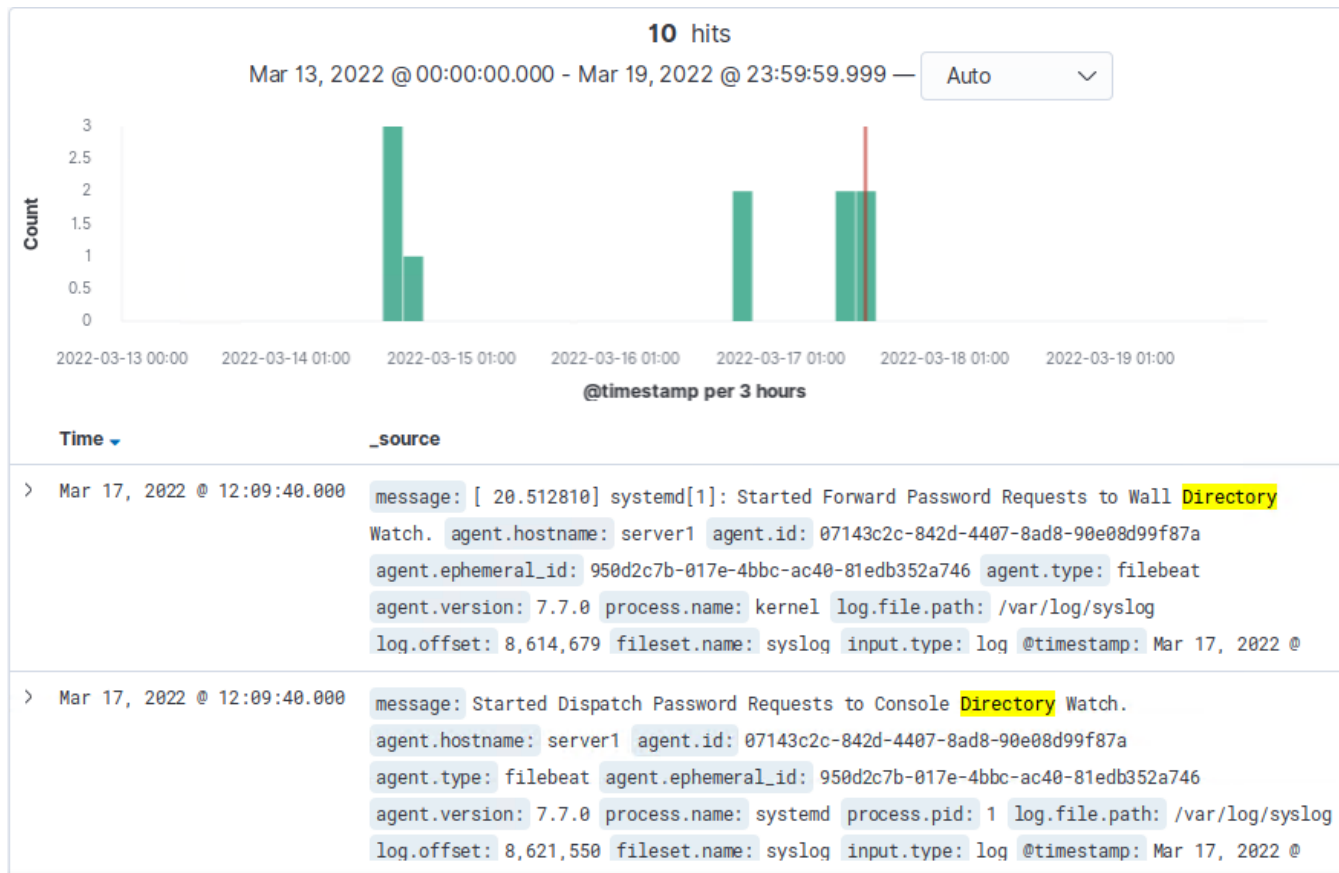
# Analysis: Finding the Request for the Hidden Directory




# Analysis: Uncovering the Brute Force Attack



# Analysis: Finding the WebDAV Connection





# **Blue Team**

## Proposed Alarms and Mitigation Strategies



# Mitigation: Blocking the Port Scan

---

## Alarm

I would suggest implementing alarms that trigger when an unusually high number of port scans and/or requests are attempted. Once a baseline is established, a trigger can be set alert the security team to monitor the activity.

The threshold to set the alarm will depend on the company's current needs, as well as future expansion. In addition, it may need to be changed depending on changes in staffing and/or business operations.

## System Hardening

Configurations can be made to the firewall on the host machine to entirely prevent or filter port scans. This is suggested as any open ports being utilized should be known by the IT team and exceptions can be made when needed.

Properly setting the firewall configurations to prevent port scans depends on how the machine was initially configured. If it is a virtual machine hosted by a cloud provider, they will generally offer firewall configuration options.

---

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

I would suggest setting an alarm detecting failed login attempts. This would allow responders to possibly prevent the attack entirely.

I would suggest a threshold of about five failed attempts before a five-minute lock-out period. This could prevent a disastrous attack and should minimize inconvenience for authorized users.

## System Hardening

To prevent access to the secret directory or any other directory containing sensitive information, I would suggest segregating those directories from the server. This would prevent unauthorized access if the server were compromised.

In addition, I would suggest such directories be password protected using strict password standards preventing the use of weak passwords.

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

What kind of alarm can be set to detect future brute force attacks? As with the previous alarm, I would set an alert for failed login attempts. Brute forcing can take a long time to complete without lock-out periods. Introducing lock-outs should give ample time to respond.

Like the previous alarm, I would suggest a threshold of about five failed attempts before a five-minute lock-out period. This could prevent a disastrous attack and should minimize inconvenience for authorized users.

## System Hardening

Again, lock-out periods can make a brute force attack far harder for the attacker. In addition, the server could be configured to prevent any traffic from IP addresses with unusually high numbers of failed login attempts.

While brute force attacks can be successful, they are time consuming for attackers and having robust password requirements in place can make a successful attack substantially more difficult and resource intensive.

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

An alarm could be set to trigger when a user accesses this directory. The cybersecurity team could then monitor the activity to determine if it is malicious and take necessary steps if required.

The threshold for this alert would depend on how much this gets accessed on a regular basis. Also, any future requirements should be considered to prevent “alert fatigue”.

## System Hardening

Configuration can be set on the host to only allow specific users to access this directory. Only those that require access should be able to interact with this directory.

Most likely, the only users that would require access to this directory would be admins. The directory should be made inaccessible to other users in the organization unless the need arises.

---

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

I suggest setting an alarm anytime a file is uploaded to the server. This is an internet facing server and should not have files being uploaded to it on a regular basis (unless designed to do so). If any files are uploaded, they would likely be uploaded by either the IT team or a malicious actor.

The threshold would depend on how the server is being used and whether employees regularly upload files. Since that doesn't seem to be the case, I would set the threshold at a single event.

## System Hardening

The host server can be set to prevent any file transfers from outside of the network. Exceptions can be set using several variables such as NAT IP address, MAC addresses, unique device identifiers/two factor authenticators...

In addition to the alert stated, keeping current on patches and updates can mitigate many reverse shell attacks.

# Additional Screenshots:

---

Flag: [https://drive.google.com/file/d/1qIpHmSP8mDv-\\_B\\_VboB3GbSgaJT4mjd/view?usp=sharing](https://drive.google.com/file/d/1qIpHmSP8mDv-_B_VboB3GbSgaJT4mjd/view?usp=sharing)

Wordlist "rockyou.txt" gunzip (used in Brute Force Attack):

<https://drive.google.com/file/d/1m4ievtbnePIHp5j46AqzINOrwRzTBdgN/view?usp=sharing>

Set Reverse Shell Payload in Meterpreter:

[https://drive.google.com/file/d/1EkBNhjeeu6\\_VSHVK\\_613SwwN4q1aSnT/view?usp=sharing](https://drive.google.com/file/d/1EkBNhjeeu6_VSHVK_613SwwN4q1aSnT/view?usp=sharing)

Dirb Scan: [https://drive.google.com/file/d/1XyFWgJIHH0JfODd21G6q\\_J9F-G520J59/view?usp=sharing](https://drive.google.com/file/d/1XyFWgJIHH0JfODd21G6q_J9F-G520J59/view?usp=sharing)

Kibana Apache Logs Uploaded:

<https://drive.google.com/file/d/1p3LqXaCJBH09YWQbnAp3vAJhCYMBgQi0/view?usp=sharing>

Hydra Brute Force Attack: <https://drive.google.com/file/d/1Gvnok2MQpcUr4IFZOTL9284CNtIfY-bS/view?usp=sharing>

---

*The  
End*