



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

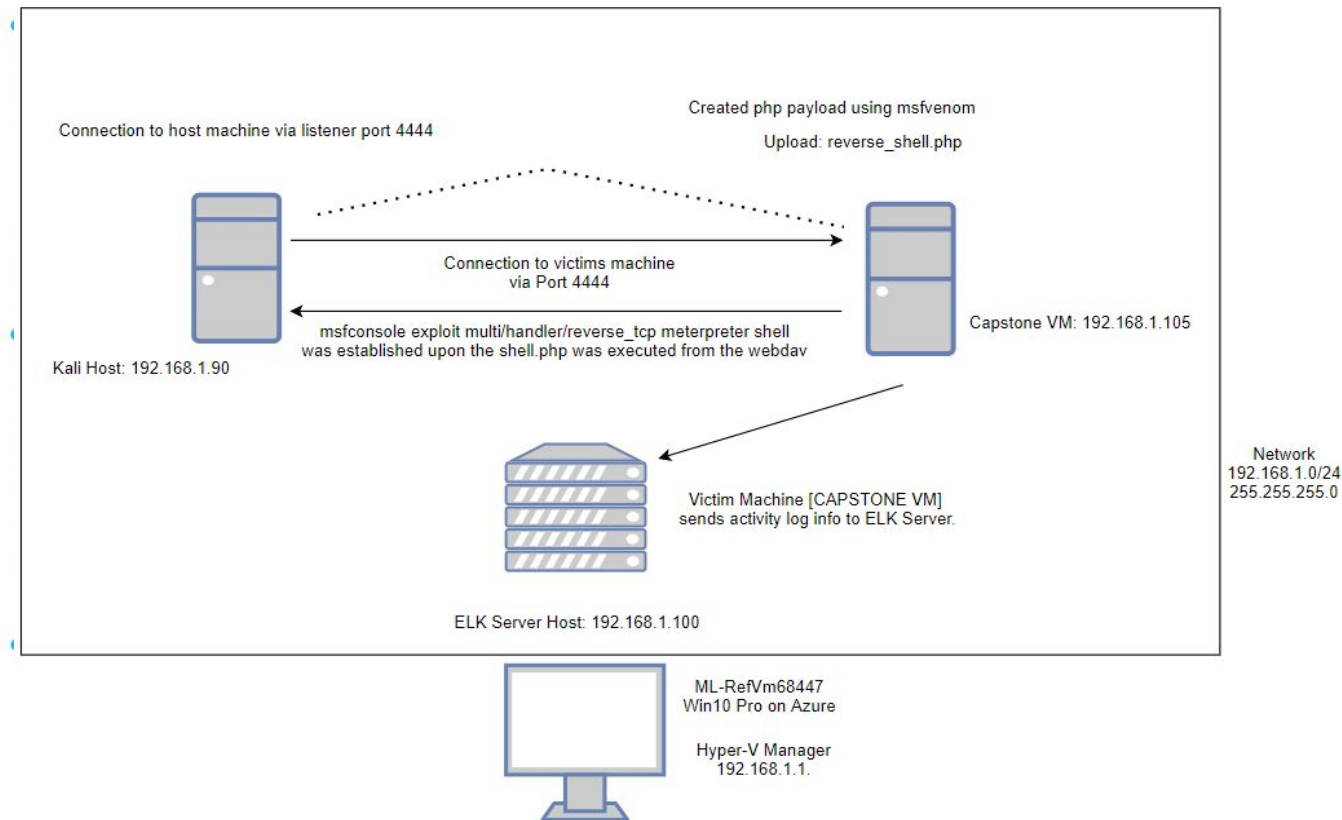
Blue Team: Log Analysis and Attack Characterization

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: 192.168.1.1
OS: Win10 Pro
Hostname: ML-RefVm68447

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.105
OS: Linux
Hostname: Server 1
(victim)

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

The background of the slide is a dark red, almost black, field filled with a complex, repeating geometric pattern of triangles and polygons in various shades of red and maroon, creating a textured, mosaic-like effect.

Red Team Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ML-RefVM-684427	192.168.1.1	VM Base hosting the 3 VM's
Kali VM	192.168.1.90	Box used for pen testing
Ubuntu	192.168.1.100	Hosting Kibana server and capturing activity on 192.168.1.105
Capstone VM / Server 1	192.168.1.105	Victim machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Open Port 80	Open ports can allow attackers to access private information and increase the risk of a data breach.	This allowed the red team to find private directory with accessible files.
Accessible Files	LFI allows access into confidential files on a site.	An LFI vulnerability allows attackers to gain access to sensitive credentials.
Brute Force Password	When the password is easy to guess it can be found in brute force tool wordlist to be hacked.	This allowed the red team to brute force Ashton's password, which was leopoldo and access to the secret.
Hashed Password	Hashed password can be cracked through different tools like John the ripper, hashcast, and other online tools. It can take only minutes to crack if the password is not salted.	This allowed the red team to use md5cracker to identify the password for John, which was linux4u.

Exploitation: Open Port 80

01

Tools & Processes

We performed a `sudo nmap` `-F` to scan for any open ports and services in the network.

02

Achievements

We found that 192.168.1.105 had an open port 80, through which we were able to access a directory with important files.

03


```
Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
root@Kali:~# nmap -F 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-07-07 17:09 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00057s latency).
Not shown: 96 filtered ports
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
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 (0.00044s latency).
Not shown: 99 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 (0.00044s latency).
Not shown: 98 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap scan report for 192.168.1.90
Host is up (0.000070s latency).
Not shown: 99 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 256 IP addresses (4 hosts up) scanned in 3.75 seconds
root@Kali:~#
```



Exploitation: Accessible Files

01

Tools & Processes

Using the open port 80, we opened firefox to view directories and files.

02

Achievements

Accessing the files gave us intel about the users; what access they have; & where their secret files were located.

03

The screenshot shows a web browser window with the address bar at `192.168.1.105/webdav/shell.php`. The main content area displays an "Index of /" directory listing. The listing table has columns for Name, Last modified, Size, and Description. The files listed are:

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	-	

Below the listing, it says "Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80".

Overlaid on the right is a terminal window showing a command prompt where the user has entered `http-ls: Volume /`. The output shows a list of files and folders with their sizes and timestamps, including `company_blog/`, `company_folders/`, `company_share/`, `meet_our_team/`, `meet_our_team/ashton.txt`, and `meet_our_team/hannah.txt`. The terminal also shows the output of `_http-server-header: Apache/2.4.29 (Ubuntu)` and `_http-title: Index of /`.

Below the browser window, there is a terminal window showing the command `192.168.1.105/meet_our_team/ashton.txt` being entered. The output shows a detailed profile of a user named Ashton, including their age, education, and job title.

Ashton is 22 years young, with a masters degree 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!

Exploitation: Brute Force Password

01

Tools & Processes

We used Hydra to brute force Ashton's password using the username: ashton

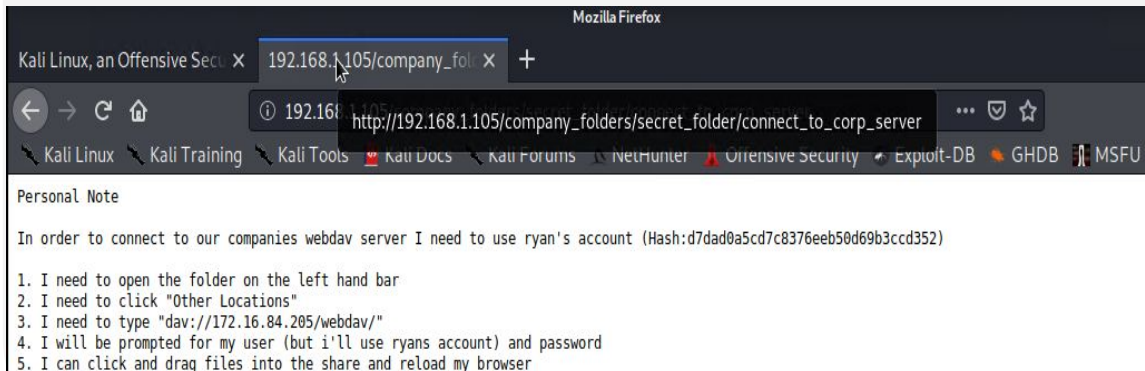
03

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 of 14344399 [child 6] (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 2021-07-07 17:50:06
root@Kali:~# cat /usr/share/wordlists#
```

02

Achievements

The exploit granted us user shell access into the victim machine so we can navigate to the secret files.



Exploitation: Hashed Password

01

Tools & Preferences

We used the website crackstation to find the plaintext of the hashed password for john.

QubesV3.1BackupDefaults

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u

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

Index of /webdav

Name	Last modified
Parent Directory	
passwd.day	2019-05-07 18:19

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

Before

02

Achievements

The password granted us access to the system through the WebDAV connection, which later allowed us to upload a shell script to attack.

Index of /webdav

Name	Last modified	Size	Description
Parent Directory		-	
passwd.day	2019-05-07 18:19	43	
shell.php	2021-07-08 04:20	1.1K	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

After

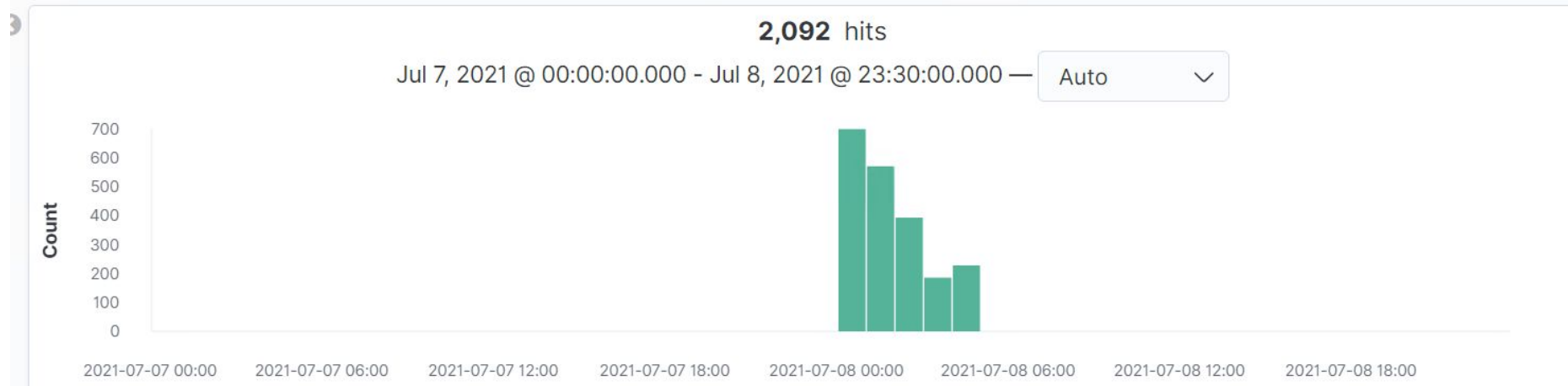


Blue Team

Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

- The port scan began around 2300 on July 8th, 2021
- 2,092 hits were sent from 192.168.1.90
- The nmap ping scan sends requests to port 443, so filtering that, we saw the ports below.



Analysis: Finding the Request for the Hidden Directory

- The event happened on 07-08-21 @ 00:00.
- 1,368,955 total requests were made.
- The secret folder contained instructions on how to access the webdav server using Ryans account which also included a hashed password.

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending

Count

http://192.168.1.105/company_folders/secret_folder

1,368,955

Mozilla Firefox

Kali Linux, an Offensive Security x 192.168.1.105/company_folders/secret_folder x +

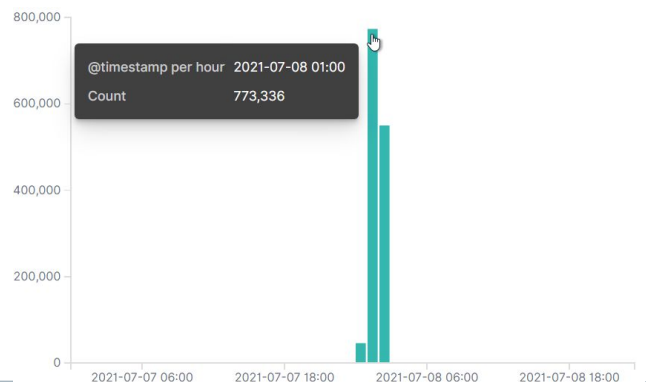
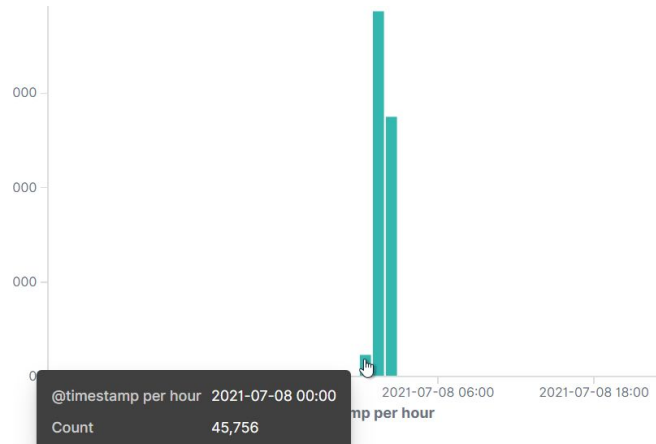
← → ↻ 🏠 ⓘ 192.168.1.105 http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server ... 🛡️ ☆

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

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



Analysis: Uncovering the Brute Force Attack

- 1,368,951 requests were made in the brute force attack.
- Out of the 1,368,951 requests, only 313 were successful in the attacker discovering the password.

server.ip	192.168.1.105
# server.port	80
# source.bytes	1678
source.ip	192.168.1.90
# source.port	36450
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)

user_agent.original:"Mozilla/4.0 (Hydra)"

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending

Count

http://192.168.1.105/company_folders/secret_folder	1,368,951
--	-----------

Export: Raw Formatted

user_agent.original:"Mozilla/4.0 (Hydra)" and not http.response.status_phrase: "unauthorized"

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending

Count

http://192.168.1.105/company_folders/secret_folder	313
--	-----

Export: Raw Formatted

Analysis: Finding the WebDAV Connection

- 112 requests were made to the Webdav directory.
- The shell.php file was requested. This was part of the red team's shell attack to start listening for activity on the victim machine.

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending ▾

Count ▾

http://192.168.1.105/webdav

112

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending ▾

Count ▾

http://192.168.1.105/webdav/shell.php

12



Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

- We will setup an alarm for when a firewall detects more than 10 port scans in a minute or 100 consecutive (ICMP) request.
- Most firewalls and IPS' can detect such scanning and cut it off in real time.

System Hardening

- Regularly run a system port scan to proactively detect and audit any open ports.
 - Only enable traffic from certain computers and deny everything else.
 - Ensure firewall is regularly patched to minimize new zero-day attacks.
-

Mitigation: Finding the Request for the Hidden Directory

Alarm

- Hydra was used to brute-force the password. An alarm can be set in the “user_agent.original” field and block the offending IP once detected.
- I would recommend a threshold of max 5 attempts per hour that would trigger an alert to be sent.

System Hardening

- Highly confidential directories should not be shared in public access.
 - Directory listing should be turned off to secure access to all of the files.
 - Encrypt data contained within confidential folders.
 - Enable MFA authentication.
-

Mitigation: Preventing Brute Force Attacks

Alarm

- Set an alert if “401 Unauthorized” is returned from any server that would weed out forgotten passwords. Start with 10 attempts in one hour and refine from there.
- I would start with five failed login attempts within 30 seconds from the same IP address.

System Hardening

- I would create a policy that locks out accounts for 30 minutes after 5 unsuccessful attempts.
- Strong password policy.
- A threshold of 10 - 401 unauthorized codes have returned from the server it can automatically drop traffic from the offending IP address for a period of one hour.

Mitigation: Detecting the WebDAV Connection

Alarm

- Create an alert anytime the directory is established by a node other than the machine should have access and drop the connection
- The threshold for this alarm would be zero so we are notified for all instances.

System Hardening

- I would set up a whitelist of authorized IPs access to the server via WebDav and block all others. I would set an alert for any attempted connections by IPs not on the whitelist for investigation.
 - Every 3 months I would review user access and check last login; if user hasn't logged in a few weeks have them request access.
 - Have user change password every few days.
 - Get rid of webdav all together and use SSH.
-

Mitigation: Identifying Reverse Shell Uploads

Alarm

- We can set an alert for any traffic moving over port "4444".
- We can set an alert for any ".php" ext file that is uploaded to the server.
- Additionally I would set up alarms whenever new files are uploaded that are executable.

System Hardening

- Ensure only necessary ports are open.
 - A browser safe filter should be used to block any malicious executable code.
 - Remove the ability to upload files to the directory over the web interface would block future file uploads.
-

*The
End*