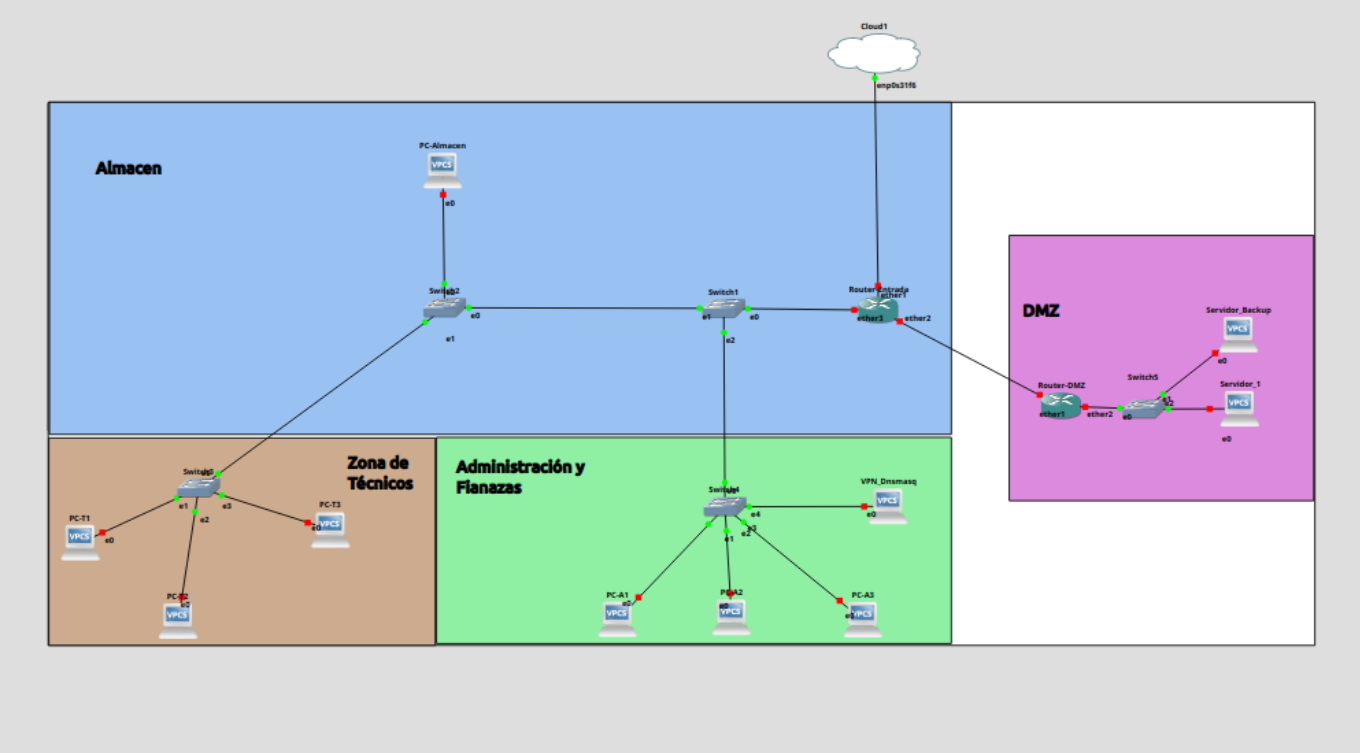
Proyecto Final de CFGS de José Ramón Peris Murcia 2º ASIR

Introducción

En este proyecto vamos a realizar la infraestructura de una empresa, en este caso será una empresa encargada de suministros escolares, material de oficina y reprografía (Gestión y reparación de máquinas fotocopiadoras).

Para ello nos vasaremos en los planos de la nave donde se encuentra la empresa y buscaremos **la mejor opción posible para realizar nuestro proyecto**.

Hay que cambiarla por la buena



Como vemos en el plano de generado en GNS3 nos encontramos con una nave de **dos plantas**. En la primera planta nos encontramos con toda la parte del **almacen** donde se alojan los suministros escolares. El **departamento de técnicos de las máquinas fotocopiadoras**, donde se encargarán de la reparación y configuración de las máquinas fotocopiadoras y la zona donde ubicaremos la **DMZ**.

En la primera planta se ubicará **el departamento de admininstración y finanzas**.

----¿Que es lo que nos pide el cliente?-----(Mejorar el titulo)

El cliente busca que realicemos una actualización de su infraestructura de red y una digitalización del almacen, de la zona de técnicos y administración.

Para ello nostros les planteamos las siguientes propuestas:

- Levantar una infraestructura de red mediante la configuración de routers Mikrotik que realizará la salida de internet, firewall y portforwarding.
- La creación de una aplicación web que permita hacer consultas sobre el stock que contiene el almacen de suministros y el stock de piezas para los técnicos.
- La implantación de una aplicación de tareas para que los técnicos puedan realizar un seguimiento de las máquinas desde que entran al taller hasta que salen del mismo.
- Desarrollo de una solución de almacenamiento de datos que permita la no dependencia de servicios de nube como Dropbox o Google Drive.
- La construcción de una DMZ que permita tener dos servidores encargados de los servicios implementados
- La implantación de una VPN que permita acceder a estos servicios.

La empresa acepta nuestra propuesta y nos pondremos **"Manos a la obra"**

Montaje de la infraestructura de red

El primer paso que vamos a realizar es el montaje de la infraestructura de la red. Este consistirá en los siguientes puntos:

- El montaje de un sistema de routers **Mikrotik** que se encargue del *enrutamiento de la infraestructura de la empresa*.
- Implantación de un servicio de **DNSMasq** responsable de *la asignación de los dns y el dhcp*.
- Configuración del **portforwarding** de los routers, buscando *una mayor seguridad y la redirección de los servicios a nuestra DMZ*.
- Compra y preparación de equipos para almacen, técnicos así como administración y dmz.

Presupuesto del proyecto

Después del estudio que hemos realizado en la nave, nos encontramos con que el *tema de infraestructura de cableado a sido montada con anterioridad al tratarse de una nave moderna*. Por lo tanto nosotros presupuestaremos tanto los equipos nuevos, la DMZ, y el tratamiento de switches, routers, etc.

Aquí nos encontramos con el presupuesto que le hemos realizado a la empresa con todos sus enlaces en la web de **PcComponentes**:

Modelo	Cantidad	Precio Unidad	Enlace
Armario Rack 19" 22U 600x600 (Para DMZ)	1	413,67€	Comprar

Modelo	Cantidad	Precio Unidad	Enlace
VidaXL Armario Rack 19" 12U 600x640mm (Para administración)	1	148,98€	Comprar
VidaXL Armario Rack 19" 6U 600x450x375mm	3	96,99€	Comprar
Mikrotik RB1100AHx4 Router Ethernet 13 Puertos RJ45 Gigabit PoE	2	314,39€	Comprar
TP-Link TL-SG1024DE Switch 24 Puertos Gigabit	5	107,43€	Comprar
Equip 326424 Patch panel 24 Puertos Cat 6	5	79,65€	Comprar
Dell PowerEdge R350 Intel Xeon E-2314/16GB/600GB	3	1699,00€	Comprar
HP Pavilion All-in-One 27-ca2008ns Intel Core i5-13400T/16GB/512GB SSD/27" (Equipos de trabajo)	7	899,01€	Comprar
Dell Vostro 3520 Intel Core i5-1235U/16GB/512GB SSD/15.6" (Portatiles de backup)	3	639,00€	Comprar
-	-	-	-
-	-	**Total: 15872,87€ IVA Incuido**	-

Configuración de la red

Ahora toca la configuración de la red. En este caso se realizará la configuración con unos routers de la marca **Mikrotik** que utilizan el sistema operativo *Router OS* que permite una gran configuración y personalización.

Para ello se realiza la configuración de los **2 routers** que se han comprado.

- Uno será el router de entrada/salida de internet.
- El otro será un router será el encargado de separar a la **DMZ** del resto de equipos. Haciendo una separación clara entre ellas.

A parte de todo esto, se creará diferentes VLANs que aportará una mayor seguridad, eficiencia y mejor gestión de las redes.

Configuración de los routers

Para configurar los routers se dividirá en dos partes:

- Configuración de la **red interna de la empresa**, en la que entrará todo el apartado de las *VLAN* y la salida a internet de los equipos.
- Configuración de la **DMZ**, donde se alojará los servidores y comprende el apartado de comuncicación entre los dos routers y la redirección de puertos.

***Falta configuracion de puertos y del router

Parte VPN

Ahora que están los routers configurados y totalmente funcionales, llega el momento a montar una VPn. Una VPN es una herramienta de red que nos permite hacer una extensión de nuestra red local. Esto es muy útil porque gracias a esto se podrá entrar a nuestra red interna desde cualquier lugar. Además, solo estará abierto el puerto de la VPN desde afuera ya que solo se puede entrar a los servidores desde la red interna como se ha realizado anteriormente en la configuración de los routers, proporcionandonos, una mayor seguridad al proyecto.

La VPN elegida para esta ocasión es **Wireguard**. **Wireguard** es una VPN creada en 2015, de código abierto y bastante popular en la comunidad. Una de las principales razones por las que hemos elegido **Wireguard** es la integración de esta VPN dentro de Mikrotik de manera nativa dentro de su S.O. *RouterOS* dando la fácilidad de configuración dentro del router. Dicho esto comienza la configuración.

1. Actualizar el Router y creación de la interfaz de Wireguard

El primer paso será realizar una actualización del router. Para eso comienza en:

```
# Busca si hay actualizaciones
/system package update check-for-updates
# Actualiza el S.O.
/system package update install
```

Después de tener el sistema operativo actualizado, tocará crear la interfaz de la VPN y su red interna.

***¡Muy importante!*. Está configuración ha sido realizada con fines explicativos. Las claves públicas y privadas mostradas en este proyecto, ya no existen porque representarían un agujero de seguridad importante.**

```
# Esto a parte de hacer la interfaz de la vpn,creará una private y public
key del servidor.
/interface/wireguard add name=wg0 listen-port=51820
```

```
[admin@MikroTik] /interface/wireguard> print
Flags: X - disabled; R - running
 0  R name="wg0" mtu=1420 listen-port=51820 private-key="GOM6dKbUgxems1Jpw0lr2PFEONFFlLZXsDH84MD0HGM=" public-key="0vBxXEqil+x3HGBA9d6daTjT0lNMZpuY83SyBaLD+F4="
[admin@MikroTik] /interface/wireguard>
```

```
# Creando la red de la VPN
/ip/adress add address=192.168.23.2/24 network=192.168.23.0 interface=wg0
```

```
[admin@MikroTik] /ip/address> print
Flags: D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
#   ADDRESS            NETWORK        INTERFACE
0   192.168.21.1/24    192.168.21.0   tecnicos
1   192.168.22.1/24    192.168.22.0   almacen
2   192.168.23.1/24    192.168.23.0   administracion
3   192.168.10.1/24    192.168.10.0   ether2
4   192.168.23.2/24    192.168.23.0   wg0
5 D 192.168.1.38/24    192.168.1.0    ether1
[admin@MikroTik] /ip/address>
```

Con esto ya estaría creada la interfaz de **Wireguard**

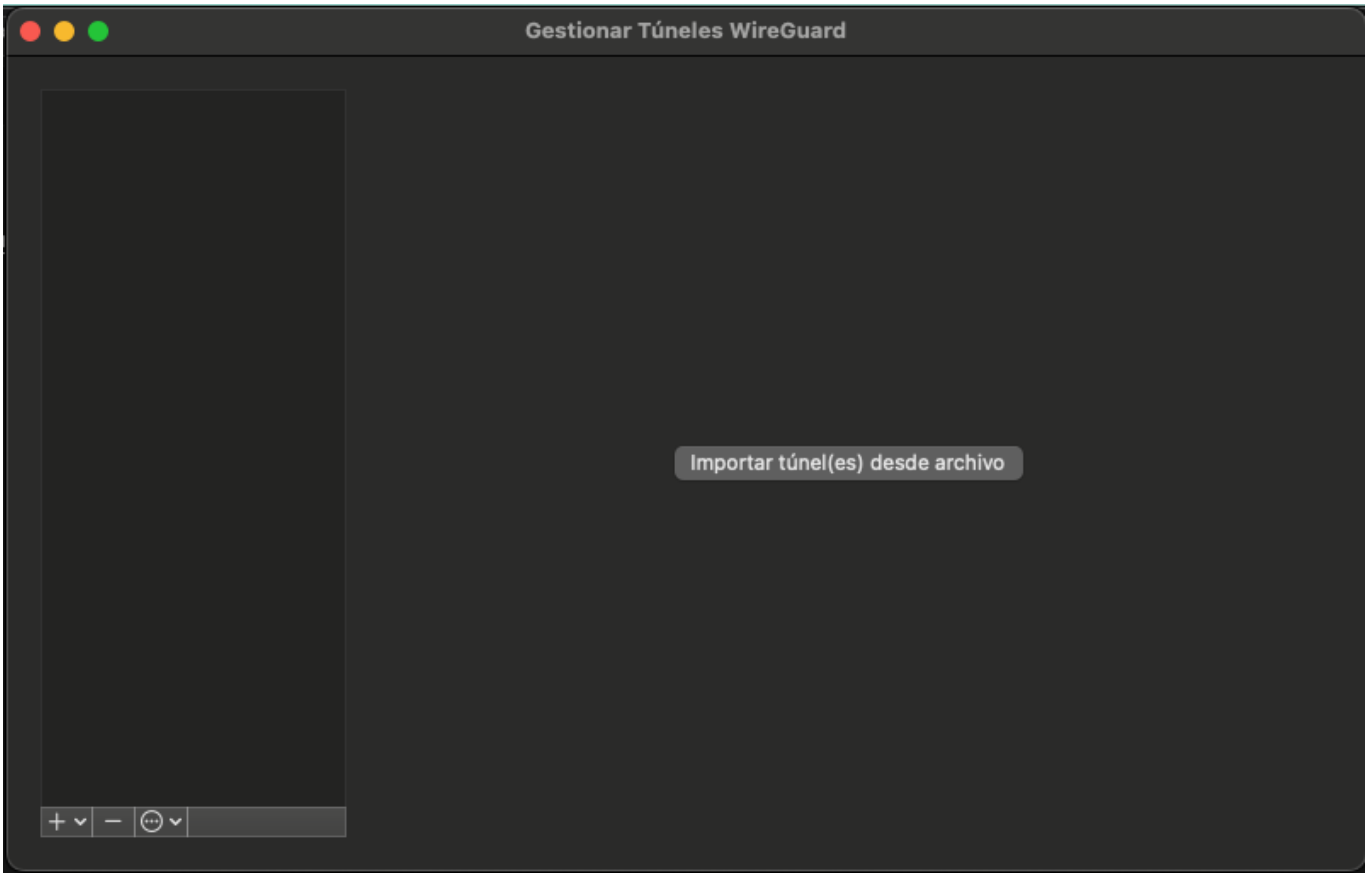
2. Configuración del Firewall

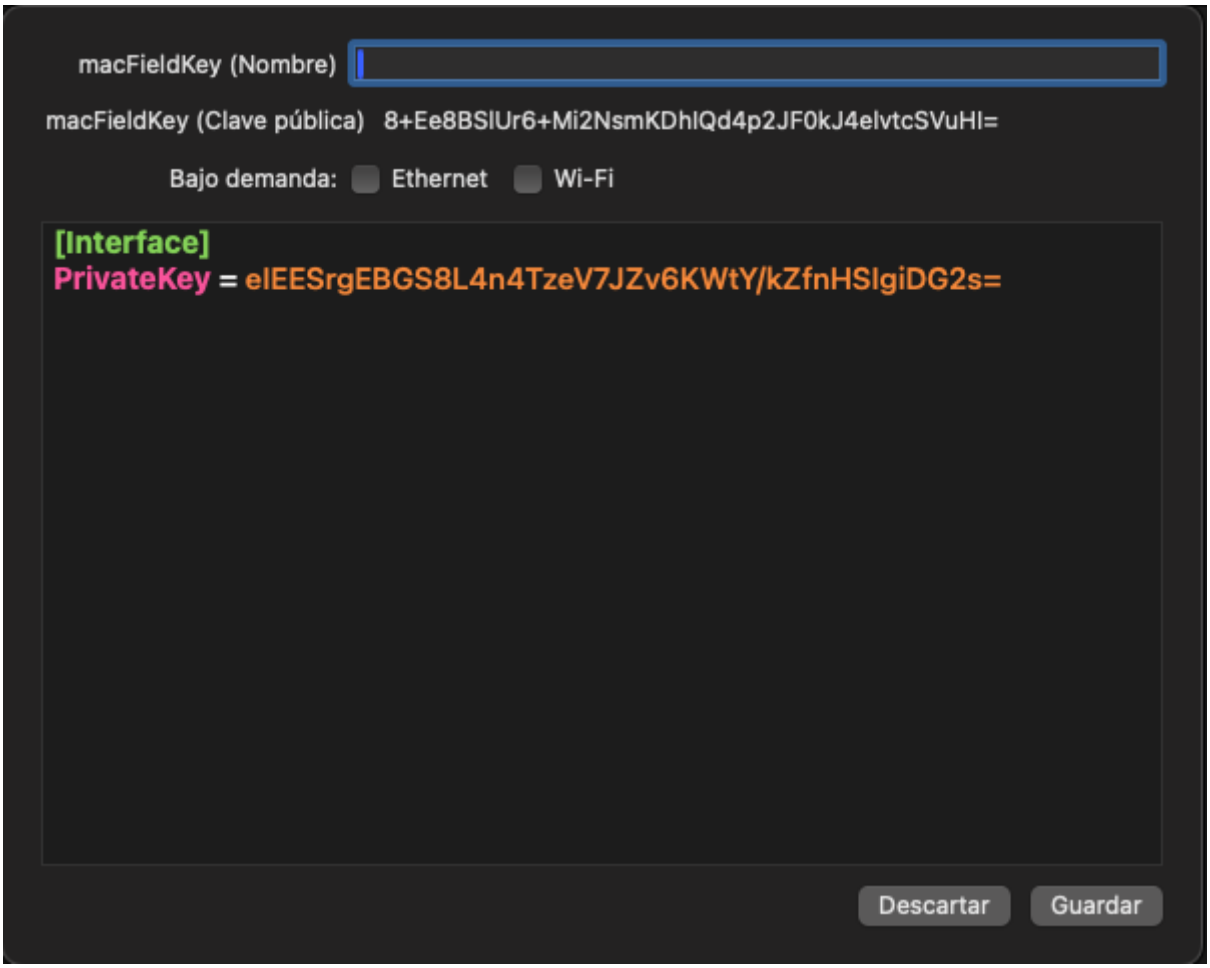
Continua con la configuración del firewall del router con el objetivo de que no solo se pueda entrar a nuestra red interna vía VPN.

```
# Bloquea cualquier acceso a la red
/ip/firewall/filter add chain=forward action=drop
# Permite el acceso tanto UDP como TCP el puerto configurado de nuestra
VPN
/ip/firewall/filter add chain=input action=accept protocol=udp dst-
port=51820
/ip/firewall/filter add chain=input action=accept protocol=tcp dst-
port=51820
# Habilita el acceso a internet al igual que hemos echo con las vlan
creadas antes
/ip/firewall/nat add chain=srcnat action=masquerade out-interface=wg0
```

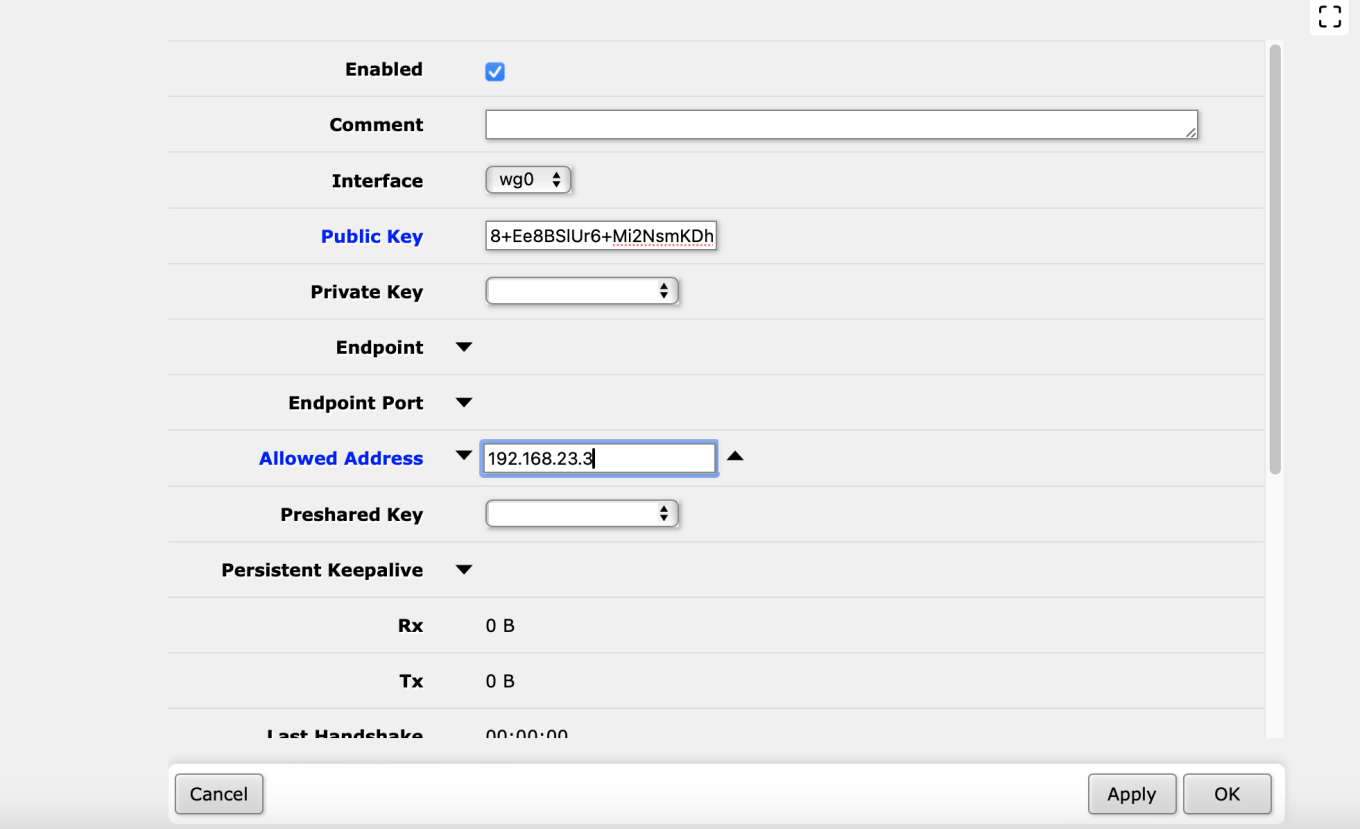
3. Creción de la peer

Ahora es el momento de la creación de la parte de *peer*. La primera parte será descargar el cliente de wireguard (En este caso el cliente de MACOSX). Después de instalar la aplicación, hay que hacer click en **crear un tunel vacio**, después se dentro de esta opción el nombre de la interfaz será el de la VPN "wg0" y se copia la clave pública.





Ahora en la web de *Mikrotik* en el apartado **Wireguard/Peers** se crea una nueva peer.



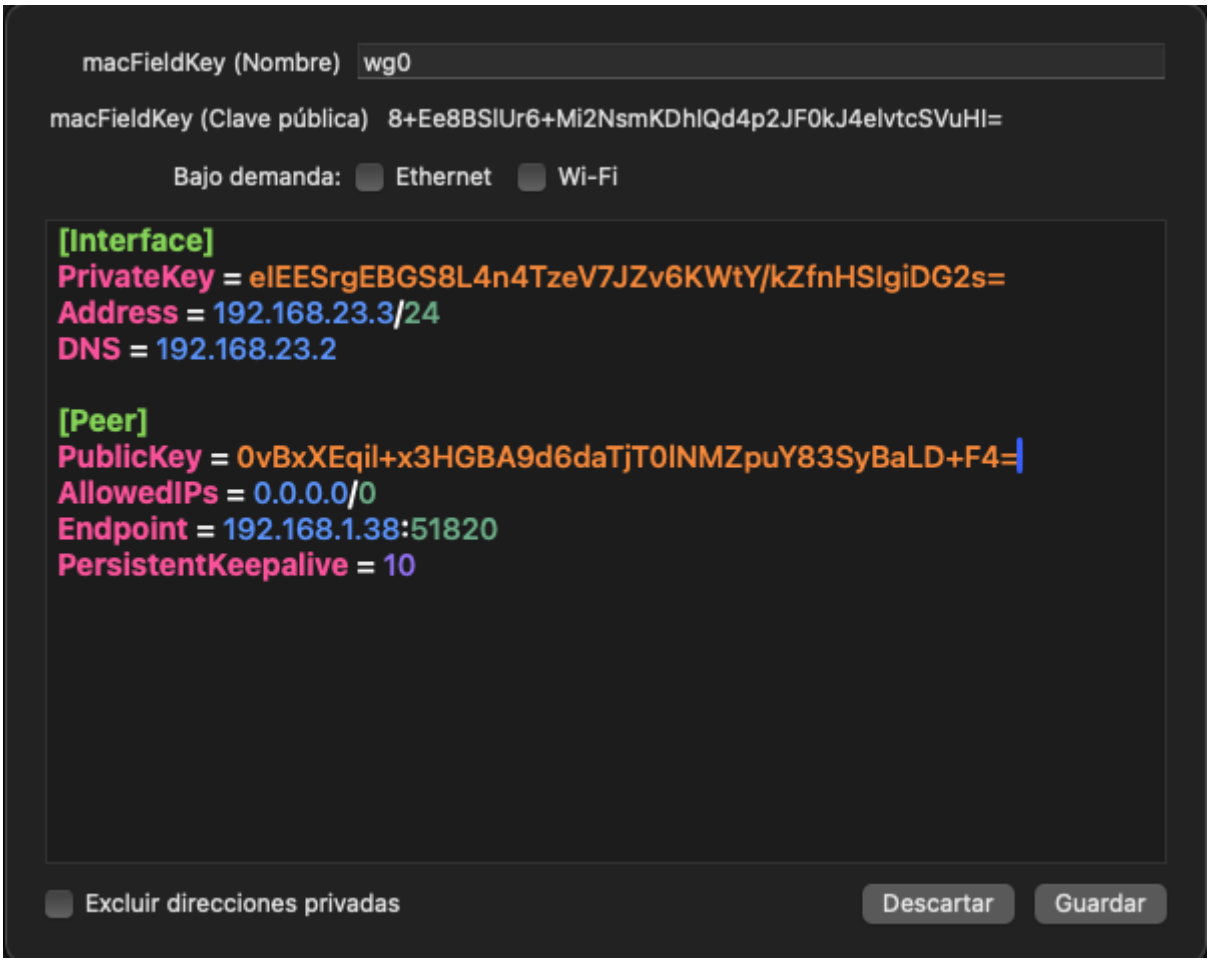
- Interface: Interfaz que utliza la peer. En este caso "wg0".
- Public Key: La clave pública del cliente de Wireguard.
- Allowed Address: La red que utilizará el cliente.

Después de configurar la Peer se generá un código QR que se puede utilizar en el cliente para móviles. Pero en este caso la configuración se realizará de forma manual.

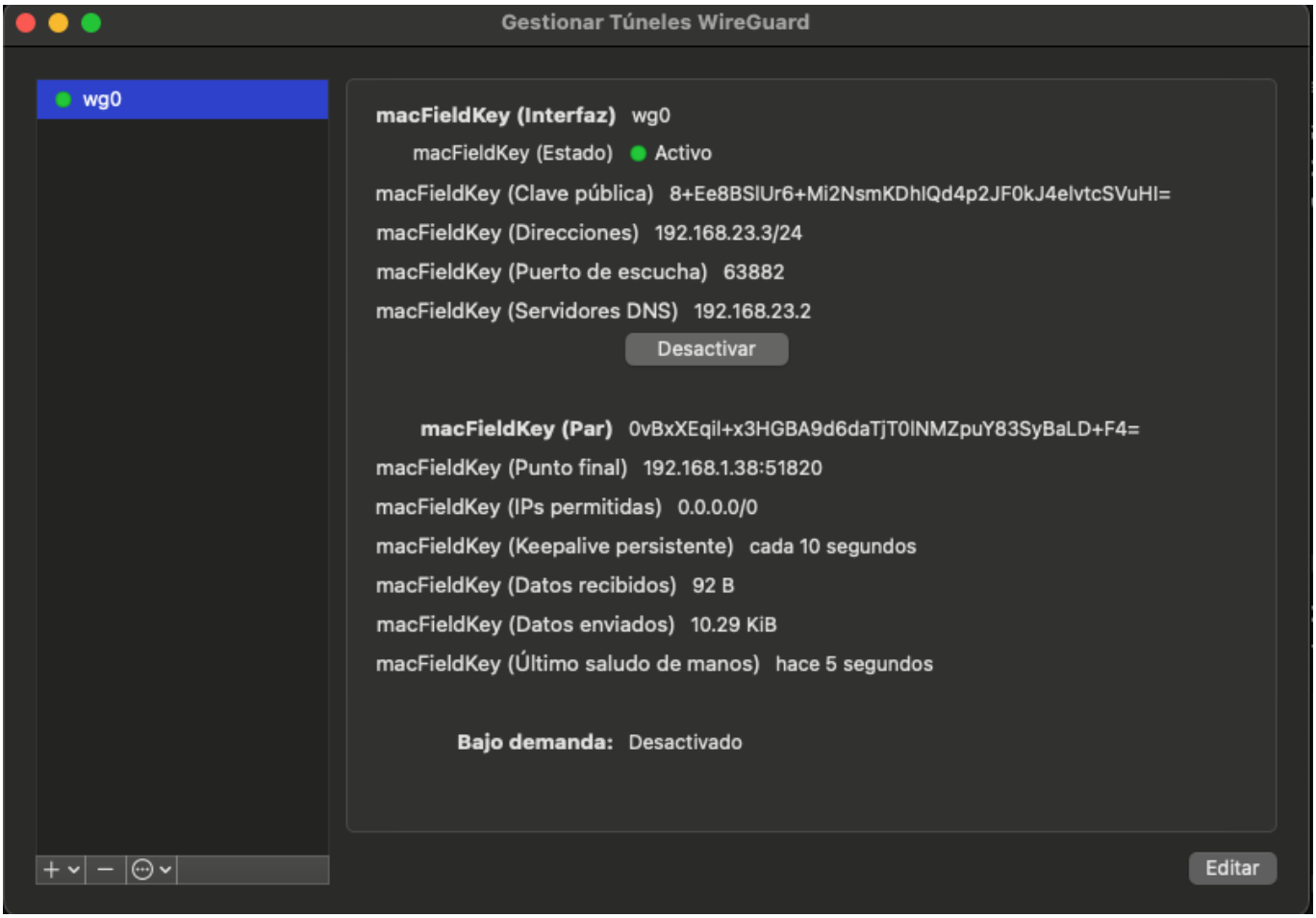
4. Configuración del cliente

Finalmente falta la configuración del cliente.

```
# Repesenta la configuración del equipo cliente
[Interface]
PrivateKey = [Generada por el cliente]
# Direccion que ocupará el equipo
Address = 192.168.23.3/24
#DNS Será la puerta de enlace del router
DNS = 192.168.23.2
# Respecto al servidor
[Peer]
PublicKey = [Clave Pública del servidor]
# Así permite cualquier ip de donde esté conectado el equipo
AllowedIPs = 0.0.0.0/0
# Donde tiene que llegar el equipo, si hubiera un dns dinamico sería esa
direccion más el puerto
Endpoint = 192.168.1.38:51820
# Manda paquetes para saber si sigue conectado
PersistentKeepalive = 10
```



Y finalmente se prueba la configuración



Sistema de almacenamiento

Terminada la parte de la red, llega la parte del almacenamiento. La empresa se le ha ofrecido una solución de almacenamiento interna por diversas razones:

- Tener una backup interno de los que se suba en la nube. Aunque la empresa tiene contratada una solución de almacenamiento en la nube, quiere poder realizar copias de seguridad de sus datos cuando ellos quieran.
- Las fotocopiadoras multifunción que reparan los técnicos utilizan para el escaneo diferentes opciones de almacenamiento (correo, ftp, pendrive...) y quieren realizar pruebas con ellas puesto que dependiendo donde vayan los equipos pueden usar opciones diferentes.

Con estás razones expuestas por el cliente se le ofrecen dos opciones:

- La creación de un servidor web **NGINX** que ofrezca un **WebDav** alojado en el servidor.
- Un servidor FTPS alojado en el servidor.

Estos dos servicios serán creados en docker con el objetivo de que puedan ser movidos al servidor de backup en caso de fallo.

Instalación de Docker

El primer paso será la instalación de Docker en el servidor. Esto permite la creación de contenedores donde se podrán alojar los diferentes servicios que se van a ir alojando dentro del servidor.

Esta instalación se realizará en Ubuntu Server 24.04

1. Conexión por ssh

```
ssh sepp@192.168.11.11
```

2. Añadir clave oficial y repositorio oficial de **Docker**

```
# Añadir clave GPG oficial de docker:
sudo apt update
sudo apt install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
# Añadir repositorio oficial a APT
echo \
  "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu
\
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update
```

3. Instalación de Docker

```
# Docker y sus plugins incluido compose
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-
plugin docker-compose-plugin
```

4. Inicio de servicio

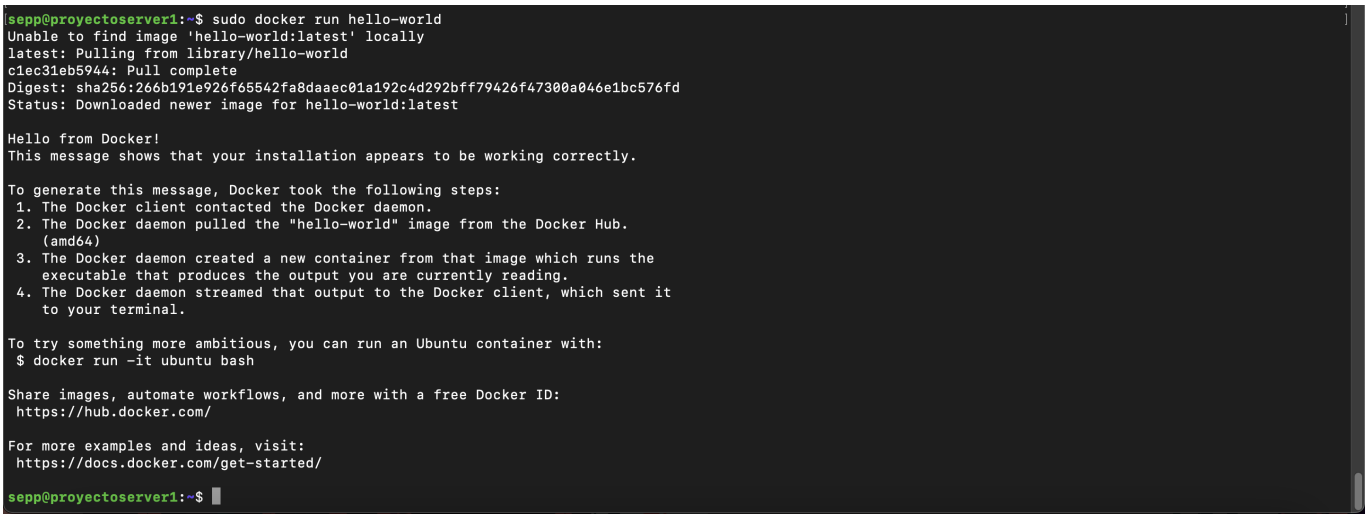
```
sudo systemctl start docker
sudo systemctl enable docker
```

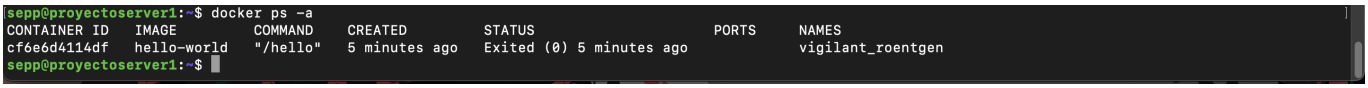
5. Hacer que no pida sudo cada vez que trabajamos con docker

```
sudo usermod -aG docker sepp
newgrp docker
```

6. Probar funcionamiento

```
docker run hello-world
```





WebDAV

Ahora que docker es completamente funcional, es el momento de centrarse en la instalación de los servicios de almacenamiento. El primer servicio instalado será WebDAV. El protocolo WebDAV permite guardar, copiar, editar o compartir archivos de manera rápida. De manera similar a Samba o FTP. WebDAV tiene soporte multiplataforma y muestra los archivos dentro de un directorio como si de un archivo local se tratase.

Para montar este sistema de almacenamiento se usará NGINX, un servidor web muy utilizado y de código abierto. El primer paso será el montaje de la imagen del NGINX.

1. Configuración

Dockerfile

```
# Imagen base utlizada
FROM debian:10.6-slim

# Argumento de uid y gid usados
ARG UID=${UID:-1000}
ARG GID=${GID:-1000}

# Actualización de los repositorios, instalación de NGINX y utilidades
```

```
necesarias más la eliminación de las listas de repositorios
RUN apt-get update && \
    apt-get install -y --no-install-recommends \
                    nginx \
                    nginx-extras \
                    apache2-utils && \
                    rm -rf /var/lib/apt/lists

# Modifica el UID y el GID de la carpeta www-data que almacena el WebDav
RUN usermod -u $UID www-data && groupmod -g $GID www-data


VOLUME /media

# Exposición del puerto 80
EXPOSE 80

# Copia el archivo de configuración creado a dentro del contenedor
COPY webdav.conf /etc/nginx/conf.d/default.conf
RUN rm /etc/nginx/sites-enabled/*

# Mueve el script creado a la raíz y le da permisos de ejecución
COPY entrypoint.sh /
RUN chmod +x entrypoint.sh

# Ejecuta el script y NGINX
CMD /entrypoint.sh && nginx -g "daemon off;"
```

docker-compose.yml

```
services:
  webdav:
    # Nombre del contenedor
    container_name: webdav
    # Nombre de la imagen o image id los dos funcionan
    image: seppwebdav:latest
    # Exposición de puertos
    ports:
      - 80:80
    # Volumenes creados
    volumes:
      - $HOME/docker/webdav:/media
    # Usuarios y contraseña
    environment:
      - USERNAME=paco
      - PASSWORD=12345
      - UID=1000
      - GID=1000
      - TZ=Europe/Madrid
    labels:
    # Opciones de traefik. Un balanceador de carga y proxy inverso
      - traefik.backend=webdav
```

```
      # Aquí si saliera fuera se pondría el dominio
      - traefik.frontend.rule=Host:localhost
      - traefik.docker.network=web
      # Reenvio de puertos
      - traefik.port=80
      # Habilita que lo gestiona trafic
      - traefik.enable=true
      # Medidas de seguridad
      -
traefik.http.middlewares.securedheaders.headers.forcestsheader=true
      - traefik.http.middlewares.securedheaders.headers.sslRedirect=true
      - traefik.http.middlewares.securedheaders.headers.STSPreload=true
      -
traefik.http.middlewares.securedheaders.headers.ContentTypeNosniff=true
      -
traefik.http.middlewares.securedheaders.headers.BrowserXssFilter=true
      -
traefik.http.middlewares.securedheaders.headers.STSIncludeSubdomains=true
      -
traefik.http.middlewares.securedheaders.headers.stsSeconds=63072000
      - traefik.http.middlewares.securedheaders.headers.frameDeny=true
      -
traefik.http.middlewares.securedheaders.headers.browserXssFilter=true
      -
traefik.http.middlewares.securedheaders.headers.contentTypeNosniff=true
networks:
  web:
   external: true
```

entrypoint.sh

```
#!/bin/bash
# Creación del usuario que pedimos en el compose
if [ -n "$USERNAME" ] && [ -n "$PASSWORD" ]
then
        htpasswd -bc /etc/nginx/htpasswd $USERNAME $PASSWORD
else
    echo Using no auth.
        sed -i 's%auth_basic "Restricted";% %g'
/etc/nginx/conf.d/default.conf
        sed -i 's%auth_basic_user_file htpasswd;% %g'
/etc/nginx/conf.d/default.conf
fi
# Cambio de propietario de /media
mediaowner=$(ls -ld /media | awk '{print $3}')
if [ "$mediaowner" != "www-data" ]
then
    chown -R www-data:www-data /media
fi
```

webdav.conf

```
dav_ext_lock_zone zone=a:10m;

server {
  set $webdav_root "/media/";
  # Necesitaran usuario y contraseña y la ubicación de esta
  auth_basic "Restricted";
  auth_basic_user_file /etc/nginx/htpasswd;
  dav_ext_lock zone=a;

  location / {

        root                    $webdav_root;
        error_page              599 = @propfind_handler;
        error_page              598 = @delete_handler;
        error_page              597 = @copy_move_handler;
        open_file_cache         off;

        access_log /var/log/nginx/webdav_access.log;
        error_log /var/log/nginx/webdav_error.log debug;

        send_timeout            3600;
        client_body_timeout     3600;
        keepalive_timeout       3600;
        lingering_timeout       3600;
        client_max_body_size    10G;

        if ($request_method = PROPFIND) {
                return 599;
        }

        if ($request_method = PROPPATCH) {
                add_header      Content-Type 'text/xml';
                return          207 '<?xml version="1.0"?><a:multistatus
xmlns:a="DAV:"><a:response><a:propstat><a:status>HTTP/1.1 200
OK</a:status></a:propstat></a:response></a:multistatus>';
        }

        if ($request_method = MKCOL) {
                rewrite ^(.*[^/])$ $1/ break;
        }

        if ($request_method = DELETE) {
                return 598;
        }

        if ($request_method = COPY) {
                return 597;
        }

        if ($request_method = MOVE) {
                return 597;
```

```
        }

        dav_methods             PUT MKCOL;
        dav_ext_methods         OPTIONS LOCK UNLOCK;
        create_full_put_path    on;
        min_delete_depth        0;
        dav_access              user:rw group:rw all:rw;

        autoindex               on;
        autoindex_exact_size    on;
        autoindex_localtime     on;

        if ($request_method = OPTIONS) {
                add_header      Allow 'OPTIONS, GET, HEAD, POST, PUT,
MKCOL, MOVE, COPY, DELETE, PROPFIND, PROPPATCH, LOCK, UNLOCK';
                add_header      DAV '1, 2';
                return 200;
        }
  }
  # Location establece directivas para mover, eliminar, copiar archivos
  location @propfind_handler {
        internal;

        open_file_cache off;
        if (!-e $webdav_root/$uri) {
                return 404;
        }
        root                    $webdav_root;
        dav_ext_methods         PROPFIND;
  }
  location @delete_handler {
        internal;

        open_file_cache off;
        if (-d $webdav_root/$uri) {
                rewrite ^(.*[^/])$ $1/ break;
        }
        root                    $webdav_root;
        dav_methods             DELETE;
  }
  location @copy_move_handler {
        internal;

        open_file_cache off;
        if (-d $webdav_root/$uri) {
                more_set_input_headers 'Destination: $http_destination/';
                rewrite ^(.*[^/])$ $1/ break;
        }
        root                    $webdav_root;
        dav_methods             COPY MOVE;
  }
}
```

Con todo esto ya está preparada la configuración estos archivos serán guardados en un repositorio aparte Inserta enlace del repositorio Webdav para poder guardarlo por si acaso.

2. Levantar Docker

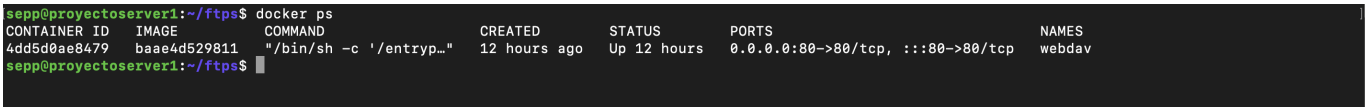
Ahora que está preparado el entorno se puede levantar el servicio de web, para ello se creará la imagen del contenedor:

```
# Se hace dentro de la carpeta que está el dockerfile
docker build -t seppwebdav:latest
```

Y ya se puede levantar el contenedor

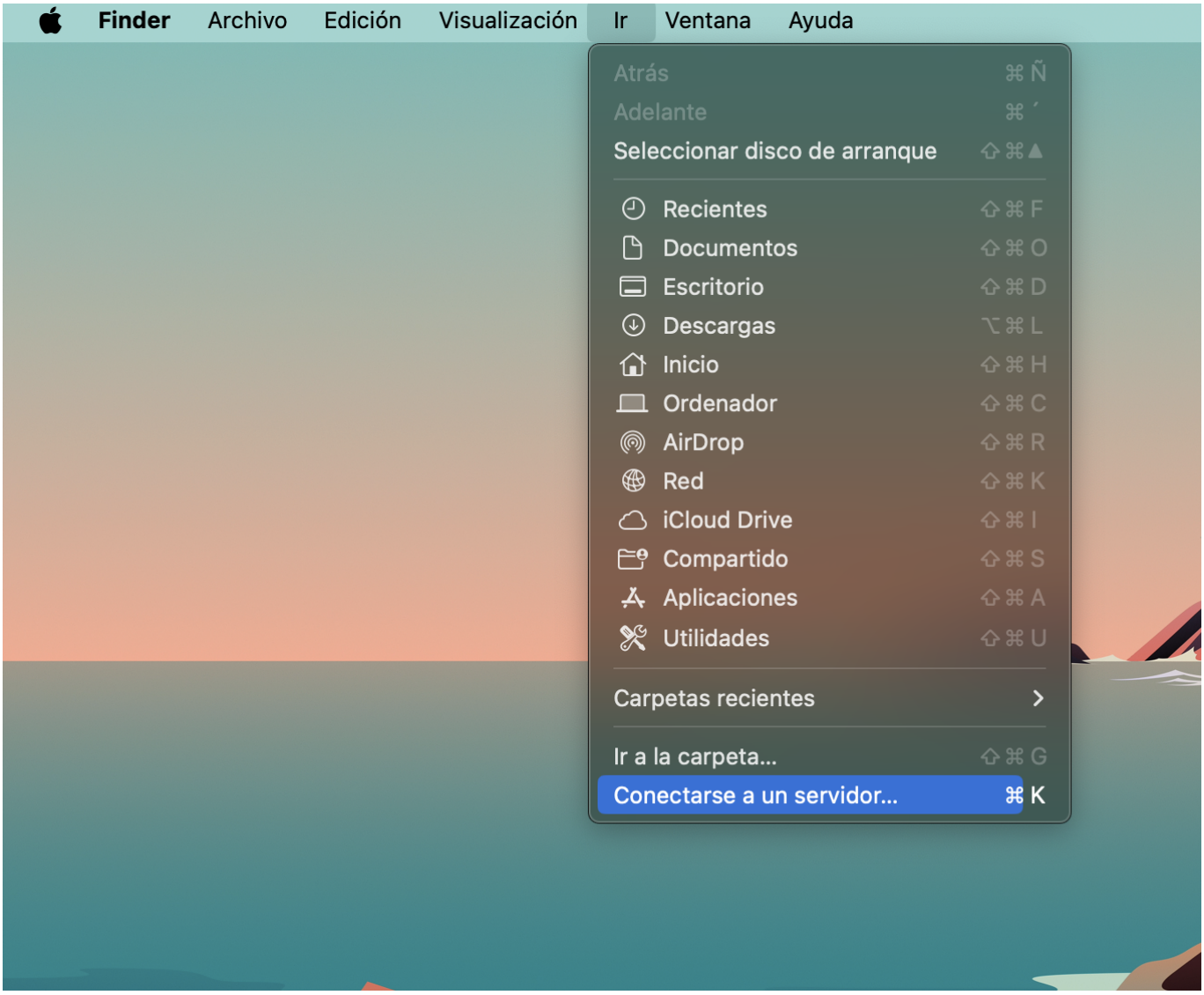
```
docker compose up -d
```

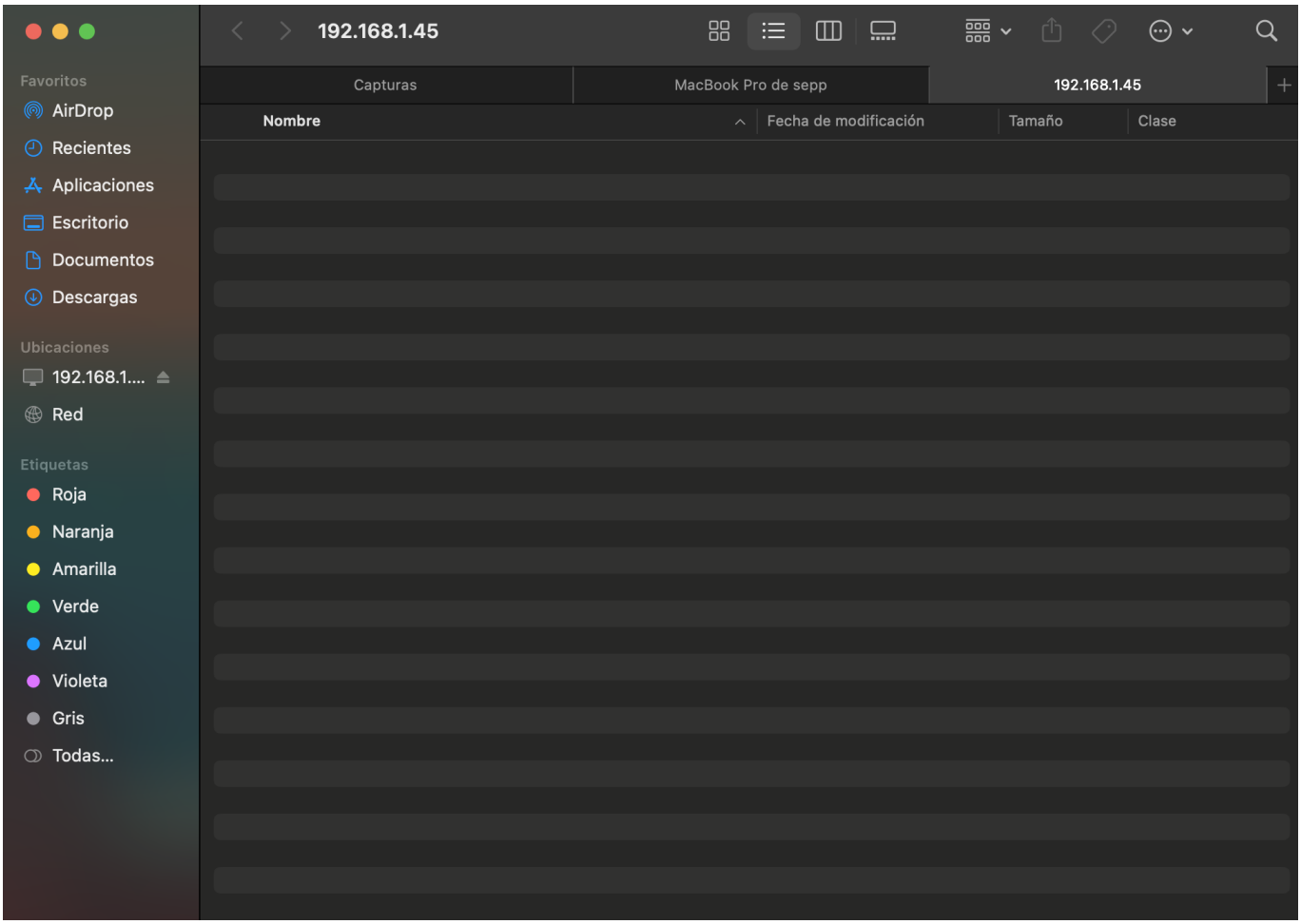
Y comprobamos que el contenedor está levantado



3. Comprobar funcionamiento

Ya levantado el contenedor, en este caso es en MACOSX, desde el finder nos dirigimos a **ir-conectarse a un servidor**. Desde allí ponemos la dirección de la puerta de enlace de nuestra red (ya que por la redirección de puertos nos mandará al servidor) y ya estaría preparado el WebDAV.


Finder
Archivo
Edición
Visualización
Ir
Ventana
Ayuda
Atrás ⌘ Ñ
Adelante ⌘ ´
Seleccionar disco de arranque ⇧⌘▲
Recientes ⇧⌘F
Documentos ⇧⌘O
Escritorio ⇧⌘D
Descargas ⌥⌘L
Inicio ⇧⌘H
Ordenador ⇧⌘C
AirDrop ⇧⌘R
Red ⇧⌘K
iCloud Drive ⇧⌘I
Compartido ⇧⌘S
Aplicaciones ⇧⌘A
Utilidades ⇧⌘U
Carpetas recientes
Ir a la carpeta... ⇧⌘G
Conectarse a un servidor... ⌘K


192.168.1.45
Favoritos
AirDrop
Recientes
Aplicaciones
Escritorio
Documentos
Descargas
Ubicaciones
192.168.1....
Red
Etiquetas
Roja
Naranja
Amarilla
Verde
Azul
Violeta
Gris
Todas...
Capturas
MacBook Pro de sepp
192.168.1.45
Nombre
Fecha de modificación
Tamaño
Clase

FTP