

Unidad didáctica nº 4- Actividad de Desarrollo-
"Servicio FTP: Ubuntu Server "
Servicios en Red e Internet – 2º ASIR
I.E.S. Miguel Herrero
Curso 2021-22

Objetivos: En esta práctica vamos a instalar y configurar un servidor FTP (ssh/sftp y ftp/vsftpd) en un SO libre, Linux Ubuntu Server así mismo vamos a configurar los servicios FTP para subir archivos a los servidores webs. También vamos a implementar un servidor FTP/VSFTPD en dockers.

Duración: 4 h

Recursos:

Vsftpd, instalación y configuración básica de un servidor FTP->
https://ubunlog.com/vsftpd-instalar-un-servidor-ftp-ubuntu/?utm_source=dlvr.it&utm_medium=facebook

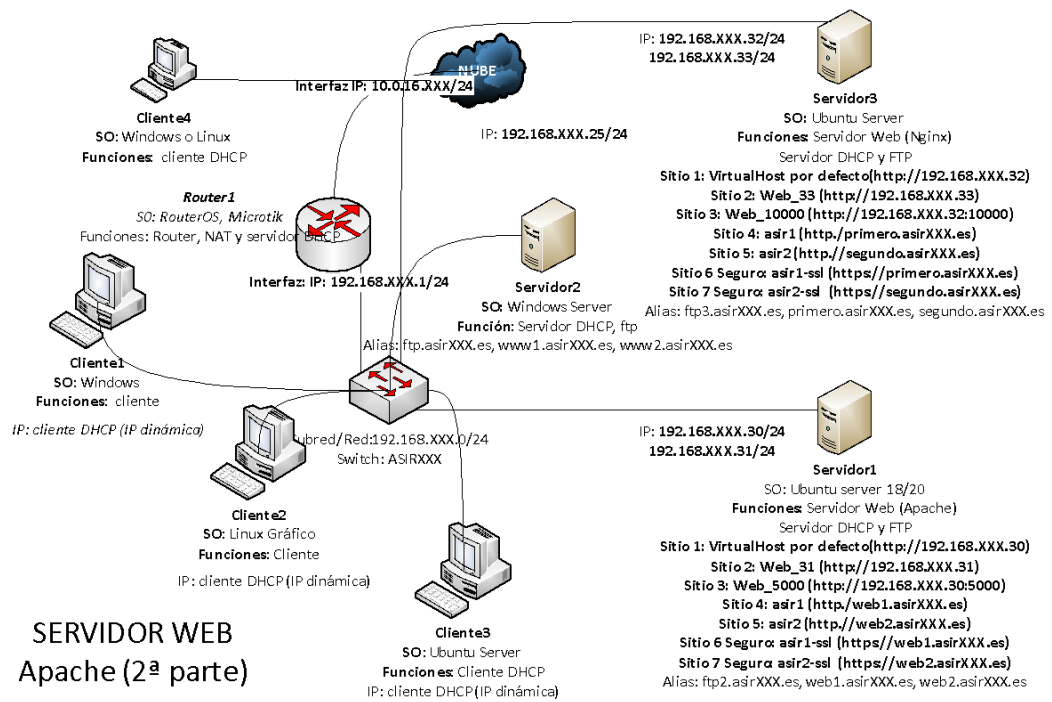
How To Install an FTP server (vsftpd) on Ubuntu 20.04/20.10->
<https://devanswers.co/install-ftp-server-vsftpd-ubuntu-20-04/>

How To Configure SFTP for a Web Server Document Root->
<https://devanswers.co/configure-sftp-web-server-document-root/>

Modo y formato de entrega: Entregar a través de la plataforma online *google classroom* en pdf llamado A2 UD4 Tu nombre y apellidos.pdf generado a partir de este y en el cual deberás añadir las respuestas a las cuestiones planteadas así como las capturas de pantalla, logs y demás evidencias necesarias para demostrar que la práctica se ha realizado cumpliendo los requisitos solicitados en el enunciado.

Escenario:

En esta práctica deberás instalar y configurar un servidor ftp/sftp en Ubuntu Server. Para ello vamos a utilizar las mismas máquinas y mismo escenario que en las Actividades 2 y 3 de la unidad 2. **Todas las máquinas estarán en la misma subred privada interna ASIRXXX**, con dirección de red **192.168.XXX.0/24 y 2001:DB8:1:XXX::/64**, donde XXX son los dígitos de tu nombre de usuario del dominio.



AMPLIACIÓN 3

Ejercicio 8: Configura un servidor VSFTPD con dockers y acceso al “documentroot” del servidor web. Usa la imagen panubo/vsftpd:

Recursos:

panubo/vsftpd->

<https://hub.docker.com/r/panubo/vsftpd/>

Lee la documentación de la imagen y configura de manera adecuada un contenedor con el servicio FTP/VSFTPD de forma que un usuario llamado **webdev_asirXXX.es** pueda llevar a cabo el mantenimiento todos los sitios 4,5,6 y 7 del servidor web (servidor 3).

Nota: No olvides parar el servicio VSFTPD en la VPS (máquina virtual servidor 3).

-Conéctate al servidor **VSFTPD** del docker (¿Cuál es su FQDN?) mediante **FTP** desde uno de los clientes 1 o 2 como usuario **webdev_asirXXX.es** utilizando un **cliente FTP**. Modifica en local la web index.html, súbela mediante ftp al servidor web y accede al sitio web desde un navegador.

Vamos a llevar a cabo la ampliación 3 paso a paso:

1ª PARTE

Objetivo: Mantenimiento del sitio/servidor web de tu máquina virtual (host) con un servidor FTP (vsftpd) en contenedor “dockerizado”..

- Instala docker en una máquina virtual Linux.
- Verifica que Docker está instalado

```
javier@javier-VirtualBox:~$ sudo systemctl status docker
[sudo] contraseña para javier:
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset:
   Active: active (running) since Mon 2022-01-24 08:45:40 CET; 5min ago
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 797 (dockerd)
      Tasks: 38
     Memory: 69.1M
    CGroup: /system.slice/docker.service
            └─ 797 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/cont>
               1111 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-por>
               1116 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 808>
               1139 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-por>
               1143 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 820>

ene 24 08:45:35 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:35.34151>
ene 24 08:45:35 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:35.45463>
ene 24 08:45:35 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:35.47461>
ene 24 08:45:35 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:35.56827>
ene 24 08:45:35 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:35.56964>
ene 24 08:45:40 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:40.66207>
ene 24 08:45:40 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:40.75016>
ene 24 08:45:40 javier-VirtualBox dockerd[797]: time="2022-01-24T08:45:40.75386>
lines 1-23
```

Pista: docker --v....

-Crea en \$HOME (/home/tu_usuario) un directorio llamado **servidorFtp** y en su interior los siguientes archivos:

```
javier@javier-VirtualBox:~$ mkdir servidorFTP
javier@javier-VirtualBox:~$ cd servidorFTP/
javier@javier-VirtualBox:~/servidorFTP$
```

servidorFtp

- vsftpd.conf
- docker-compose.yml
- Dockerfile

```
javier@javier-VirtualBox:~/servidorFTP$ sudo nano vsftpd.conf
javier@javier-VirtualBox:~/servidorFTP$ sudo nano docker-compose.yml
javier@javier-VirtualBox:~/servidorFTP$ sudo nano Dockerfile
javier@javier-VirtualBox:~/servidorFTP$ ls
docker-compose.yml  Dockerfile  vsftpd.conf
javier@javier-VirtualBox:~/servidorFTP$
```

- Descarga la imagen **panubo/vsftpd**

Pista: docker p.....

```
javier@javier-VirtualBox:~/servidorFTP$ sudo docker pull panubo/vsftpd
Using default tag: latest
latest: Pulling from panubo/vsftpd
844c33c7e6ea: Pull complete
733472f55b4a: Pull complete
536619de91e8: Pull complete
85fd04420380: Pull complete
05bafd24bc71: Pull complete
a90d209903f3: Pull complete
27c3de81e9ef: Pull complete
Digest: sha256:3842741f4bfbc2f8c7c293ac243fce5f25ad2b83468e195649431c2a060691c6
Status: Downloaded newer image for panubo/vsftpd:latest
docker.io/panubo/vsftpd:latest
javier@javier-VirtualBox:~/servidorFTP$
```

- Muestra las imágenes que tienes descargadas

Pista: docker im.....

```
javier@javier-VirtualBox:~/servidorFTP$ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
javersanchezgonzalez/mi_servidor_web  v1         01a1d7f0ea15  10 days ago  472MB
mi_servidor_web     latest     01a1d7f0ea15  10 days ago  472MB
joomla              latest     de6fac0e496d  4 weeks ago  548MB
php                 8.0-apache 4a8710f05615  4 weeks ago  472MB
mysql               8.0         3218b38490ce  4 weeks ago  516MB
phpmyadmin/phpmyadmin  latest     2e5141bbcbfb  7 months ago  474MB
panubo/vsftpd        latest     dafd22f924fa  2 years ago  107MB
javier@javier-VirtualBox:~/servidorFTP$
```

-Haz un inspect de la imagen de panubo/vsftpd e indica qué puertos (ExposedPorts) y volúmenes (Volumes) expone a título informativo

Pista: docker in.....

```
javier@javier-VirtualBox:~/servidorFTP$ docker inspect panubo/vsftpd
[
  {
    "Id": "sha256:dafd22f924fa7d8bb14675ecc7d8235abfcfc03982a574b23a05c31ebdc3d808",
    "RepoTags": [
      "panubo/vsftpd:latest"
    ],
    "RepoDigests": [
      "panubo/vsftpd@sha256:3842741f4bfb2f8c7c293ac243fce5f25ad2b83468e195649431c2a060691c"
    ],
    "Parent": "",
    "Comment": "",
    "Created": "2019-12-02T04:22:31.819702624Z",
    "Container": "fd0328d573ef8a101a552db760a08f3cd4140a5011fd955aff993096c680b554",
    "ContainerConfig": {
      "Hostname": "fd0328d573ef",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "ExposedPorts": {
        "21/tcp": {},
        "4559/tcp": {},
        "4560/tcp": {},
        "4561/tcp": {},
        "4562/tcp": {},
        "4563/tcp": {},
        "4564/tcp": {}
      }
    }
  }
]
```

- Acude a la documentación de la imagen panubo/vsftpd en docker-hub y responde a las siguientes cuestiones:

- ¿Qué aplicaciones tiene en general el servidor vsftpd?

- Explica para que sirven las siguientes variables de entorno:

FTP_USER: Establece el usuario FTP predeterminado.

FTP_PASSWORD: Contraseña de texto sin formato (no recomendado).

FTP_CHOWN_ROOT: si se establece chown se ejecutará contra /srv configurando el usuario y el grupo FTP como propietario y grupo del directorio.

- Indica cuál es el directorio en la imagen donde está ubicada la configuración y raíz del servicio ftp:

-Configuración del servidor FTP (vsftpd.conf):

-Raíz(root) servidor FTP:

Pista: Volumes del inspect

- Ejecuta un contenedor demonio, de nombre *servidorFtp1*, a partir de la imagen *panubo/vsftpd* de forma que lleves el mantenimiento de algún sitio web ubicado en tu máquina virtual o host (por ejemplo recuerda que en *nginx* el directorio del sitio web por defecto es: /var/www/html). Utiliza el **usuario**: asirXXX y **contraseña**: usuario@1

Pista (sustituye los interrogantes por el correspondiente valor): *docker run -d - -name ¿? -p 20:20 -p 21:21 -p 4559-4564:4559-4564 -v ¿?:/srv -e FTP_USER=¿? -e FTP_PASSWORD=¿? -e FTP_CHOWN_ROOT=true panubo/vsftpd*

Nota: La variable de entorno FTP_CHOWN_ROOT=true asigna como propietario y grupo al directorio del contenedor /srv (De esa manera se puede escribir/subir archivos al servidor FTP). PERO NO LO HACE DE MANERA RECURSIVA. Ello puede generar problemas a la hora de subir archivos al servidor FTP. Lo ideal sería que el directorio a mapear estuviera, **vacío**; sin embargo eso no es habitual.

Por ello, tienes que acceder al contenedor, **servidorFtp1**, y establecer como usuario y grupo propietario del directorio raíz `/srv`, a `ftp:ftp`. Para ello sigue los siguientes pasos:

1º.- Inicia en el contenedor **servidorFtp1** una terminal interactiva (bash)

Pista: docker exec -it

-Ejecuta en el contenedor **servidorFTP1** el siguiente comando: ***chown -R ftp:ftp /srv***

```
javier@javier-VirtualBox:~$ docker exec -it servidorFTP1 bash
root@ae8a58a4171f:/# chown -R ftp:ftp /srv
root@ae8a58a4171f:/# exit
exit
```

```
javier@javier-VirtualBox:~$ docker run -d -p 20:20 -p 21:21 -p 4559-4564:4559-4564 -v /var/www/html:/srv --name servidorFTP1 -e FTP_USER=asir222 -e FTP_PASSWORD=usuario@1 -e FTP_CHOWN_ROOT=true panubo/vsftpd
ae8a58a4171f2a8f846cf347086e84acc199c69bc91368007b0f9a3c8fa536ca
javier@javier-VirtualBox:~$
```

- Lista los contenedores que se están ejecutando.

Pista: docker p....

```
javier@javier-VirtualBox:~/servidorFTP$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS
73504063ad47   panubo/vsftpd                      "/entry.sh vsftpd"      About a minute Up About a minute
0.0.0.0:21->21/tcp, :::21->21/tcp, 0.0.0.0:4559-4564->4559-4564/tcp, :::4559-4564->4559-4564/tcp   servidorFTP1
f4cd97edb602   joomla                             "/entrypoint.sh apac..." 10 days ago    Up 55 minutes
0.0.0.0:8200->80/tcp, :::8200->80/tcp
joomla
87b0015e73b6   phpmyadmin/phpmyadmin             "/docker-entrypoint..." 10 days ago    Up 55 minutes
0.0.0.0:8080->80/tcp, :::8080->80/tcp
joomlapma
fb50f7ad9250   mysql:8.0                         "docker-entrypoint.s..." 10 days ago    Up 55 minutes
3306/tcp, 33060/tcp
joomladb
```

- Haz un inspect del contenedor *servidorFtp1* e indica la red a la que está conectado, Ip, máscara y puerta de enlace

Pista: docker i.....

```
8e1",
      "Gateway": "172.17.0.1",
      "IPAddress": "172.17.0.2",
      "IPPrefixLen": 16,
      "IPv6Gateway": "",
      "GlobalIPv6Address": "",
      "GlobalIPv6PrefixLen": 0,
      "MacAddress": "02:42:ac:11:00:02",
      "DriverOpts": null
    }
  }
}
]
```

- Conectate al servidor ftp

Pista: ftp ip_servidor_ftp (No utilices localhost)

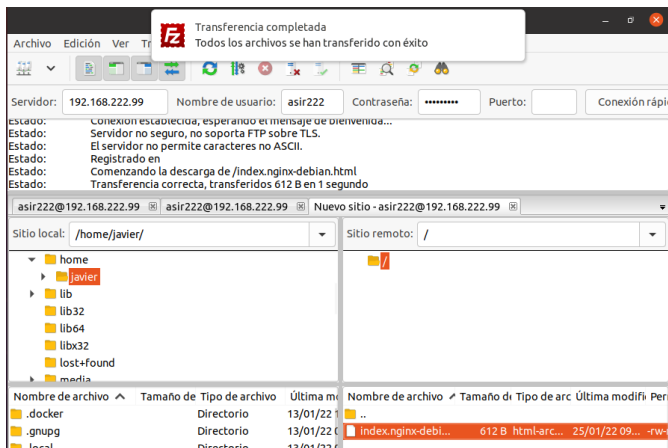
```
javier@javier-VirtualBox:~$ ftp 192.168.222.99
Connected to 192.168.222.99.
220 (vsFTPd 3.0.3)
Name (192.168.222.99:javier): asir222
331 Please specify the password.
Password:
230 Login successful.
```

- Muestra el contenido de directorio raíz

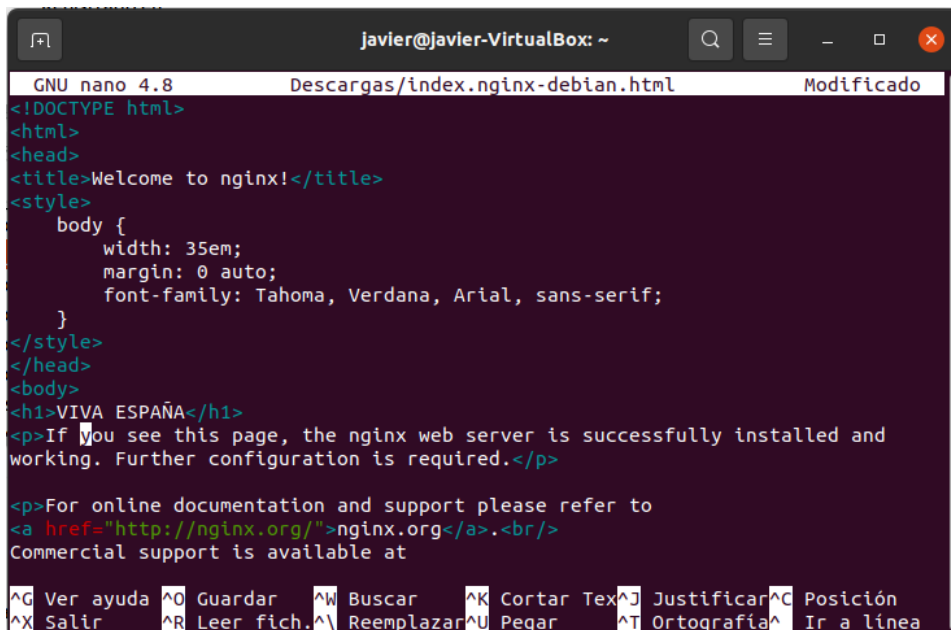
```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 ftp ftp 612 Jan 25 08:10 index.nginx-debian.html
226 Directory send OK.
ftp> █
```

Nota: Si no puedes mostrarlo, activa el modo pasivo en el cliente

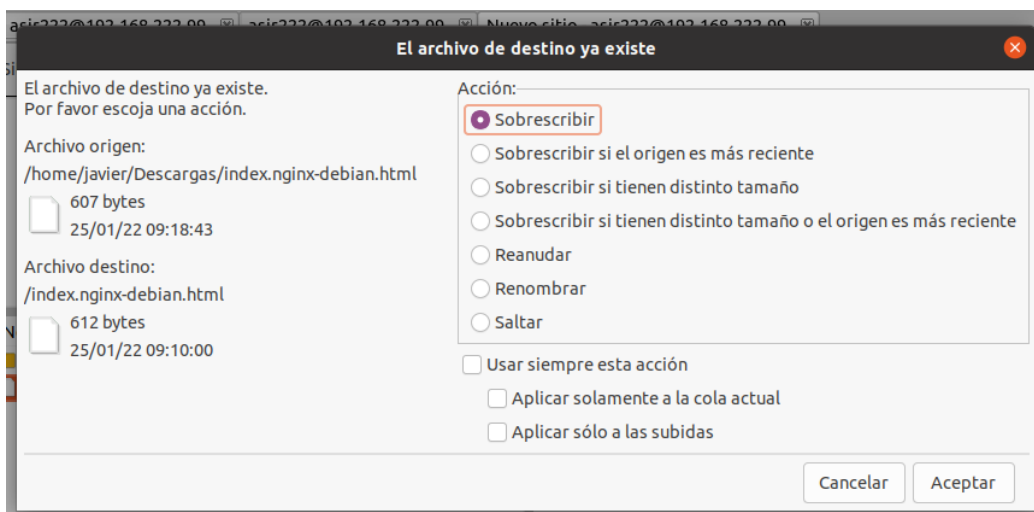
- Descarga el index.html



- Modifica el index.html



- Sube el index.html



Nota: Si no puedes subir el index.html, puede ser por problemas de permisos en el directorio raíz, es decir, el usuario *ftp* del contenedor no tiene permisos para escribir en el directorio raíz.

.- Para resolver el problema tienes dos posibles soluciones. Te recomiendo que pruebes ambas:

SOLUCIÓN 1: Accede al contenedor, *servidorFtp1*, y establece como usuario y grupo propietario del directorio raíz */srv*, a *ftp:ftp*. Sigue los siguientes pasos:

.-Inicia en el contenedor *servidorFtp1* una terminal interactiva (bash)

Pista: docker exec.....

.-Ejecuta en el contenedor el siguiente comando: *chown -R ftp:ftp /srv*

SOLUCIÓN 2: Generar un nuevo contenedor y establecer la variable de entorno *FTP_CHOWN_ROOT=true*

.-Borra el contenedor *servidorFtp1*

Pista: docker r.....

```
javier@javier-VirtualBox:~$ docker stop ae8a58a4171f
ae8a58a4171f
javier@javier-VirtualBox:~$ docker rm ae8a58a4171f
ae8a58a4171f
javier@javier-VirtualBox:~$
```

2ª PARTE

Objetivo: Instalación de un servidor/sitio web *nginx* en contenedor “dockerizado” y llevar a cabo su mantenimiento con un servidor FTP *vsftpd* también en contenedor.

SERVIDOR WEB EN CONTENEDOR

.-Crea en \$HOME (/home/tu_usuario) un directorio llamado *servidorWeb1* y en su interior los siguientes archivos y directorios:

servidorWeb1

```
|— conf.d
|   |— default.conf
|— docker-compose.yml
|— Dockerfile
|— html
|   |— index.html
```

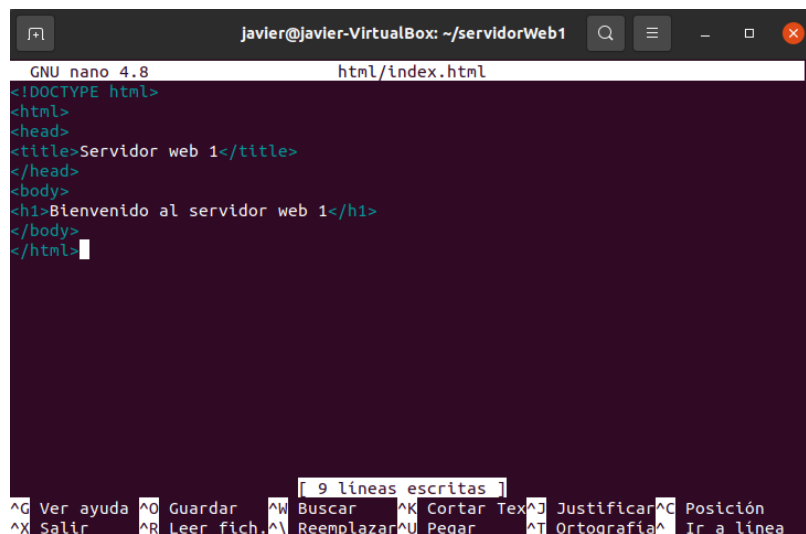
```
javier@javier-VirtualBox:~$ mkdir servidorWeb1 servidorWeb1/conf.d servidorWeb1/html
javier@javier-VirtualBox:~$ ls servidorWeb1/
conf.d  html
javier@javier-VirtualBox:~$
```

```
javier@javier-VirtualBox:~/servidorWeb1$ sudo nano docker-compose.yml
javier@javier-VirtualBox:~/servidorWeb1$ sudo nano Dockerfile
javier@javier-VirtualBox:~/servidorWeb1$ sudo nano conf.d/default.conf
javier@javier-VirtualBox:~/servidorWeb1$ sudo nano html/index.html
javier@javier-VirtualBox:~/servidorWeb1$
```


-El contenido del index.html es el siguiente:

index.html:

```
<!DOCTYPE html>
<html>
<head>
<title>Servidor web 1</title>
</head>
<body>
<h1>Bienvenido al servidor web 1</h1>
</body>
</html>
```

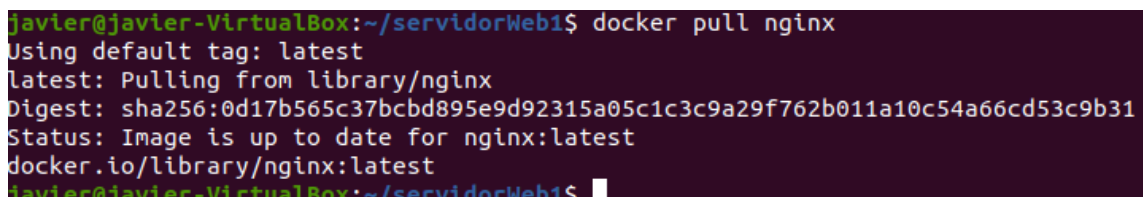
A screenshot of a terminal window titled 'javier@javier-VirtualBox: ~/servidorWeb1'. It shows the GNU nano 4.8 text editor editing the file 'html/index.html'. The editor contains the following HTML code: <!DOCTYPE html>, <html>, <head>, <title>Servidor web 1</title>, </head>, <body>, <h1>Bienvenido al servidor web 1</h1>, </body>, and </html>. The cursor is at the end of the last line. At the bottom of the terminal, there is a status bar showing '[9 líneas escritas]' and a list of keyboard shortcuts: ^G Ver ayuda, ^O Guardar, ^W Buscar, ^K Cortar Text, ^J Justificar, ^D Posición, ^X Salir, ^R Leer fich., ^U Reemplazar, ^U Pegar, ^T Ortografía, and ^_ Ir a línea.

- Acude a la documentación de la imagen *nginx* en docker-hub y responde a las siguientes cuestiones:

- ¿Qué aplicaciones/ usos tiene en general el servidor nginx y en particular la imagen nginx que acabas de descargar?
- Indica la ruta (ubicación) de los siguientes archivos en el contenedor:
 - Directorio ubicación sitio web por defecto (index.html): /var/html/www/
 - Fichero de ubicación de nginx (nginx.conf): [/etc/nginx/](#)
 - Fichero de configuración del sitio web por defecto (default.conf): [/etc/nginx/conf.d/](#)

- Descarga la imagen nginx

Pista: docker p.....

A screenshot of a terminal window showing the command 'docker pull nginx' being executed. The output is: 'Using default tag: latest', 'latest: Pulling from library/nginx', 'Digest: sha256:0d17b565c37bcbd895e9d92315a05c1c3c9a29f762b011a10c54a66cd53c9b31', 'Status: Image is up to date for nginx:latest', and 'docker.io/library/nginx:latest'. The prompt 'javier@javier-VirtualBox: ~/servidorWeb1\$' is visible at the bottom.

- Muestra las imágenes que tienes tu registro local

Pista: docker im.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
SIZE
mi_servidor_web     latest             01a1d7f0ea15       11 days ago
472MB
javiersanchezgomez/mi_servidor_web v1                 01a1d7f0ea15       11 days ago
472MB
nginx               latest             605c77e624dd       3 weeks ago
141MB
joomla              latest             de6fac0e496d       4 weeks ago
548MB
php                 8.0-apache        4a8710f05615       4 weeks ago
472MB
mysql              8.0                3218b38490ce       5 weeks ago
516MB
phpmyadmin/phpmyadmin latest             2e5141bbcbfb       7 months ago
474MB
panubo/vsftpd       latest             dafd22f924fa       2 years ago
107MB
javier@javier-VirtualBox:~/servidorWeb1$
```

-Haz un inspect de la imagen de nginx para ver qué puertos (ExposedPorts) expone.

Pista: docker ins.....

```
"ContainerConfig": {
  "Hostname": "ca3e48389f71",
  "Domainname": "",
  "User": "",
  "AttachStdin": false,
  "AttachStdout": false,
  "AttachStderr": false,
  "ExposedPorts": {
    "80/tcp": {}
  },
  "Volumes": {}
}
```

- Crea un contenedor demonio, llamado *servidorWeb1*, accesible con el puerto 8081, usando la imagen oficial de **nginx**. (Expón el puerto 8081 de tu host al puerto 80 del contenedor) y monta el directorio del sitio web por defecto (/usr/share/...) en el directorio html creado anteriormente

Pista: docker run -d -p.....- -name.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker run -d -p 8081:80 --name servidorWeb1 -v /home/javier/servidorWeb1/html:/usr/share/nginx/html nginx
db207cb66b70439c0381fc385a44e858f89691c9152c3dae5e82cd64f1098cd7
javier@javier-VirtualBox:~/servidorWeb1$
```

-Muestra la lista de contenedores y comprueba que *servidorWeb1* se está ejecutando

Pista: docker p.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS       PORTS                               NAMES
4f892e736eef  nginx     "/docker-entrypoint..." About a minute ago Up About a minute 0.0.0.0:8081->80/tcp, :::8081->80/tcp servidorWeb1
f4cd97edb602   joomla    "/entrypoint.sh apac..." 11 days ago   Up About an hour 0.0.0.0:8200->80/tcp, :::8200->80/tcp joomla
87b0015e73b6   phpmyadmin/phpmyadmin "/docker-entrypoint..." 11 days ago   Up About an hour 0.0.0.0:8080->80/tcp, :::8080->80/tcp joomlapma
fb50f7ad9250   mysql:8.0 "docker-entrypoint.s..." 11 days ago   Up About an hour 3306/tcp, 33060/tcp             joomladb
javier@javier-VirtualBox:~/servidorWeb1$
```

-Inicia en el contenedor *servidorWeb1* una terminal interactiva (bash) y muestra en el mismo su dirección IPv4, puerta de enlace y muestra el sitio web (curl)

Pista: docker exec.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker exec -it servidorWeb1 bash
root@4f892e736eef:/# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
16: eth0@if17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
        valid_lft forever preferred_lft forever
root@4f892e736eef:/#
```

-Accede al navegador web desde el host (máquina virtual) y comprueba que el servidor web 1 está funcionando.

Pista: <http://localhost>.....



SERVIDOR FTP EN CONTENEDOR

-Borra el contenedor *servidorFtp1*

Pista: docker r.....

- Ejecuta un nuevo contenedor demonio, de nombre *servidorFtp2*, a partir de la imagen *panubo/vsftpd* de forma que lleves el mantenimiento del *servidorWeb1*. Utiliza el **usuario:** asirXXX y **contraseña:** usuario@1

Pista: `docker run -d - --name ¿? -p 20:20 -p 21:21 -p 4559-4564:4559-4564 -v ¿?:/srv -e FTP_USER=¿? -e FTP_PASSWORD=¿? -e FTP_CHOWN_ROOT=true panubo/vsftpd`

```
javier@javier-VirtualBox:~$ docker run -d -p 20:20 -p 21:21 -p 4559-4564:4559-4564 --name servidorFtp2 -v /home/javier/servidorWeb1/html:/srv -e FTP_USER=asir222 -e FTP_PASSWORD=usuario@1 -e FTP_CHOWN_ROOT=true panubo/vsftpd
b27023bd714c0f99649e56d1ca8beb2881e1ca62db4de76cd8111dc9ace8dd55
```

Nota: No te olvides acceder al contenedor, *servidorFtp2*, y establecer como usuario y grupo propietario del directorio raíz, /srv, a ftp: ftp al igual que hiciste en servidorFtp1

```
javier@javier-VirtualBox:~$ docker exec -it servidorFtp2 bash
root@eccde9b9af12:/# chown -R ftp:ftp /srv
root@eccde9b9af12:/#
```

-Muestra la lista de contenedores y comprueba que *servidorFtp2* se está ejecutando

Pista: docker p.....

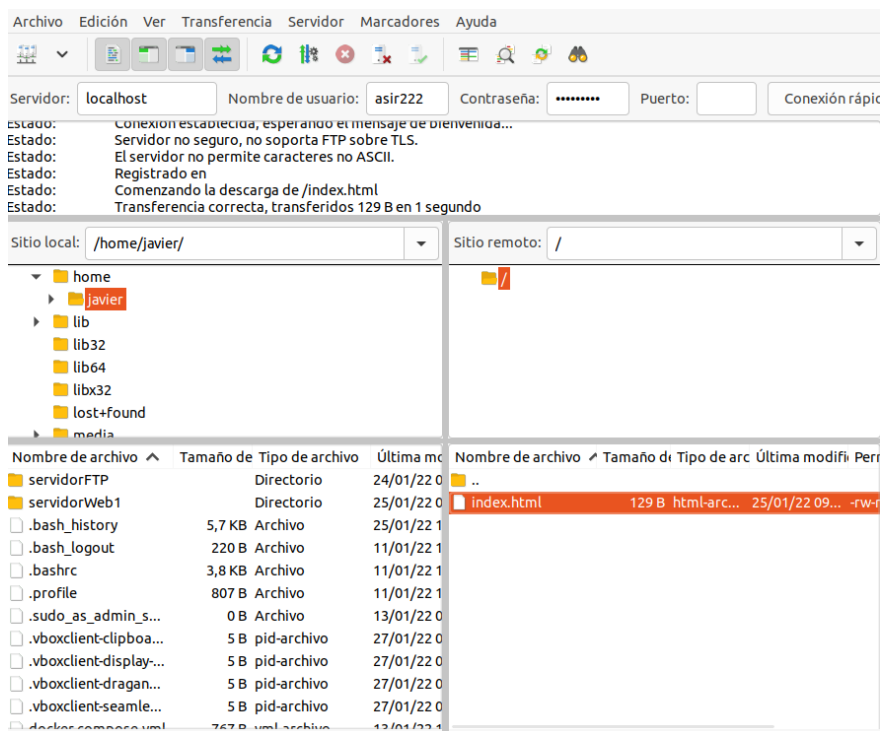
```
javier@javier-VirtualBox:~$ docker ps | grep servidorFtp
eccde9b9af12  panubo/vsftpd  "/entry.sh vsftpd"  About a minute ago  Up About a minute  0.0.0.0:20->20/tcp, :::20->20/tcp, 0.0.0.0:4559-4564->4559-4564/tcp, :::4559-4564->4559-4564/tcp, 21/tcp  servidorFtp2
javier@javier-VirtualBox:~$
```

-Inicia en el contenedor *servidorFtp2* una terminal interactiva (bash) y haz un ping al contenedor *servidorWeb1*. (ping....).

Pista:docker exec.....

```
javier@javier-VirtualBox:~$ docker exec -it servidorFtp2 bash
root@eccde9b9af12:/# ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.066 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.063 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.039 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.043 ms
64 bytes from 172.17.0.2: icmp_seq=5 ttl=64 time=0.040 ms
64 bytes from 172.17.0.2: icmp_seq=6 ttl=64 time=0.039 ms
^C
--- 172.17.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5103ms
rtt min/avg/max/mdev = 0.039/0.048/0.066/0.012 ms
root@eccde9b9af12:/#
```


-Utiliza el contenedor *servidorFtp2* para descargar el index.html del *servidorweb1*, modificarlo y subirlo al *servidorWeb1*.



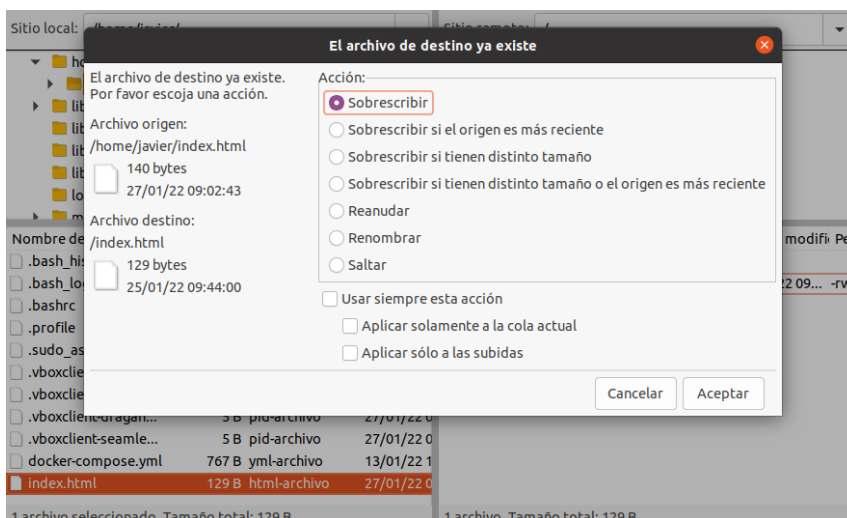
index.html (modificado):

```
<!DOCTYPE html>
<html>
<head>
<title>Servidor web 1</title>
```

```
</head>
<body>
<h1>Bienvenido al servidor web 1 DEFINITIVO</h1>
</body>
</html>
```

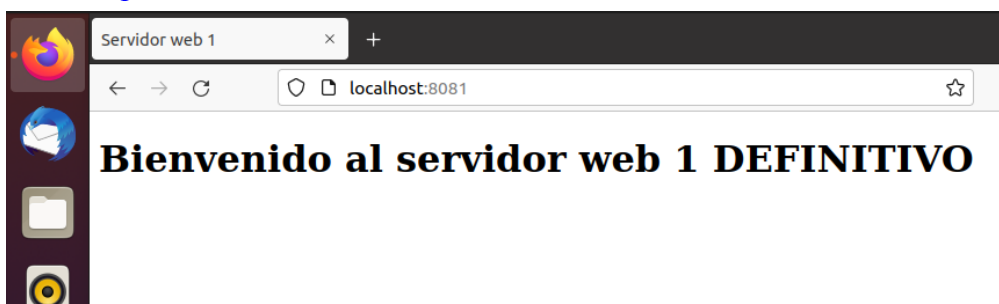


```
GNU nano 4.8 index.html Modificado
<!DOCTYPE html>
<html>
<head>
<title>Servidor web 1</title>
</head>
<body>
<h1>Bienvenido al servidor web 1 DEFINITIVO</h1>
</body>
</html>
```



-Accede al navegador web del host (máquina virtual) y comprueba que el servidor web 1 esta funcionando y modificado.

Pista: <http://localhost>.....



3ª PARTE

Objetivo. Vamos a generar los contenedores *servidorWeb1* y *servidorFtp2* con **docker-compose** y repetir los pasos de la 2ª parte, es decir, EN VEZ DE GENERAR LOS CONTENEDORES CON **DOCKER RUN** como en la 1ª y 2ª parte LOS VAMOS A CONSTRUIR CON LA HERRAMIENTA **DOCKER-COMPOSE** configurando los escenarios en **docker-compose.yml**. Esta segunda manera es la forma habitual de arrancar los contenedores.

-Borra los contenedores *servidorWeb1* y *servidorFtp2*

Pista: docker r.....

```
javier@javier-VirtualBox:~$ docker stop servidorFtp2
servidorFtp2
javier@javier-VirtualBox:~$ docker rm servidorFtp2
servidorFtp2
javier@javier-VirtualBox:~$ docker stop servidorWeb1
servidorWeb1
javier@javier-VirtualBox:~$ docker rm servidorWeb1
servidorWeb1
javier@javier-VirtualBox:~$
```

-Instala docker-compose.

```
javier@javier-VirtualBox:~$ sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left  Speed
100 664    100 664    0     0  3928      0 --:--:-- --:--:-- --:--:-- 3952
100 12.1M  100 12.1M    0     0 8656k      0 0:00:01 0:00:01 --:--:-- 10.0M
javier@javier-VirtualBox:~$ sudo chmod +x /usr/local/bin/docker-compose
javier@javier-VirtualBox:~$
```

- Verifica que docker-compose está instalado

Pista: docker-compose --....

```
javier@javier-VirtualBox:~$ docker-compose --version
docker-compose version 1.29.2, build 5becea4c
javier@javier-VirtualBox:~$
```

SERVIDOR WEB EN CONTENEDOR ESCENARIO CON DOCKER-COMPOSE

-Vamos a crea el escenario del servidor web 1. Ubícate en el directorio **\$HOME/servidorWeb1**

-Edita el archivo **docker-compose.yml**

-Copia las siguientes líneas en docker-compose.yml:

version: "3.9"

services:

web:

container_name: *servidorWeb1*

image: nginx:latest


```

volumes:
- ./html:/usr/share/nginx/html
ports:
- 8081:80
networks:
  contenedores:
    ipv4_address: 192.168.20.101
networks:
  contenedores:
    name: red_contenedores
  ipam:
    config:
      - subnet: 192.168.10.0/24

```

The screenshot shows a terminal window titled 'javier@javier-VirtualBox: ~/servidorWeb1'. The editor is GNU nano 4.8, editing 'docker-compose.yml'. The file content is as follows:

```

version: "3.9"
services:
  web:
    container_name: servidorWeb1
    image: nginx:latest
    volumes:
      - ./html:/usr/share/nginx/html
    ports:
      - 8081:80
    networks:
      contenedores:
        ipv4_address: 192.168.10.101
networks:
  contenedores:
    name: red_contenedores
    ipam:
      config:
        - subnet: 192.168.10.0/24

```

The bottom of the terminal shows nano editor shortcuts: Ver ayuda, Guardar, Buscar, Cortar, Text, Justificar, Posición, Salir, Leer fich, Reemplazar, Pegar, Ortografía, Ir a línea.

-Explica el significado de cada una de las líneas anteriores del archivo docker-compose.yml

-Levanta el escenario con docker-compose en modo demonio (detach)

Pista: docker-compose u.....

The terminal shows the command 'docker-compose up -d' being executed. The output is 'Recreating servidorWeb1 ... done'. The prompt returns to 'javier@javier-VirtualBox:~/servidorWeb1\$'.

-Muestra los contenedores en ejecución con docker-compose

Pista: docker-compose p.....

The terminal shows the command 'docker-compose ps'. The output is a table with columns: Name, Command, State, and Ports.

Name	Command	State	Ports
servidorWeb1	/docker-entrypoint.sh nginx	Up	0.0.0.0:8081->80/tcp, :::8081->80/tcp
...			

The prompt returns to 'javier@javier-VirtualBox:~/servidorWeb1\$'.

-Muestra un listado de los contenedores con docker

Pista: docker p.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS      PORTS
d467d3174b93   nginx:latest   "/docker-entrypoint...." About a minute ago Up A
bout a minute   0.0.0.0:8081->80/tcp, :::8081->80/tcp
servidorWeb1
javier@javier-VirtualBox:~/servidorWeb1$
```

Muestra un listado de las redes

Pista: docker ne.....

```
javier@javier-VirtualBox:~/servidorWeb1$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
fba3b52ea595        bridge              bridge              local
2421371fde9d        host                host                local
d5b81c103f6b        javier_default      bridge              local
f4ec5778b3df        none                null                local
68226e72ae8e        red_contenedores    bridge              local
javier@javier-VirtualBox:~/servidorWeb1$
```

-Accede al navegador web del host y comprueba que el servidor web 1 esta funcionando.

Pista: <http://localhost>.....



SERVIDOR FTP EN CONTENEDOR ESCENARIO CON DOCKER-COMPOSE

-Vamos a crea el escenario del servidor ftp 2. Ubícate en el directorio **\$HOME/servidorFtp**

```
javier@javier-VirtualBox:~/servidorWeb1$ cd ../servidorFTP/
javier@javier-VirtualBox:~/servidorFTP$
```

-Edita el archivo Dockerfile

-Copia las siguientes líneas

FROM panubo/vsftpd

RUN apt update && apt install -y nano

RUN chown -R ftp:ftp /srv

EXPOSE 21

A screenshot of a terminal window with the nano text editor open. The title bar says 'javier@javier-VirtualBox: ~/servidorFTP'. The editor shows the following content: 'GNU nano 4.8 Dockerfile', 'FROM panubo/vsftpd', 'RUN apt update && apt install -y nano', 'RUN chown -R ftp:ftp /srv', and 'EXPOSE 21'.

-Explica el significado de cada una de las líneas anteriores. ¿Qué estamos haciendo en el Dockerfile? ¿Te suena de algo?

- Genera una imagen de nombre *tu_usuario_docker_hub/vsftpd*

Pista: `docker build -t.....`

```
javier@javier-VirtualBox:~/servidorFTP$ docker build -t javiersanchezgomez/vsftpd .
Sending build context to Docker daemon 3.072kB
Step 1/4 : FROM panubo/vsftpd
----> dafd22f924fa
Step 2/4 : RUN apt update && apt install -y nano
----> Running in bf55b0864eb2
```

- Haz un listado de las imágenes y comprueba que se ha creado la imagen: *tu_usuario_docker_hub/vsftpd*

Pista: `docker i.....`

```
javier@javier-VirtualBox:~/servidorFTP$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED
SIZE			
javiersanchezgomez/vsftpd	latest	204e10902f9c	6 seconds ago
127MB			
javiersanchezgomez/mi_servidor_web	v1	01a1d7f0ea15	13 days ago
472MB			
mi_servidor_web	latest	01a1d7f0ea15	13 days ago
472MB			
nginx	latest	605c77e624dd	4 weeks ago
141MB			
joomla	latest	de6fac0e496d	5 weeks ago
548MB			
php	8.0-apache	4a8710f05615	5 weeks ago
472MB			
mysql	8.0	3218b38490ce	5 weeks ago
516MB			
phpmyadmin/phpmyadmin	latest	2e5141bbcbfb	7 months ago
474MB			
panubo/vsftpd	latest	dafd22f924fa	2 years ago
107MB			

-Edita el archivo `docker-compose.yml`

-Copia las siguientes líneas en `docker-compose.yml`

version: "3.9"

services:

vsftpd:

image: *tu_usuario_docker_hub/vsftpd*

container_name: ***servidorFtp2***

environment:

- FTP_USER= asirXXX

- FTP_PASSWORD=usuario@1

- FTP_CHOWN_ROOT=true

volumes:

- ../servidorWeb1/html:/srv

ports:

- 0.0.0.0:21:21

- 0.0.0.0:20:20

- 4559-4564:4559-4564

```

networks:
  contenedores:
    ipv4_address: 192.168.20.100
networks:
  contenedores:
    external: true
    name: red_contenedores

```

```

GNU nano 4.8 docker-compose.yml
version: "3.9"
services:
  vsftpd:
    image: javiersanchezgomez/vsftpd
    container_name: servidorFtp2
    environment:
      - FTP_USER= asir222
      - FTP_PASSWORD=usuario@1
      - FTP_CHOWN_ROOT=true
    volumes:
      - ../servidorWeb1/html:/srv
    ports:
      - 0.0.0.0:21:21
      - 0.0.0.0:20:20
      - 4559-4564:4559-4564
    networks:
      contenedores:
        ipv4_address: 192.168.10.100
networks:

```

- Explica el significado de cada una de las líneas anteriores
- Levanta el escenario con docker-compose

Pista: docker-compose u.....

```

javier@javier-VirtualBox:~/servidorFTP$ docker-compose up -d
Recreating servidorFtp2 ... done
javier@javier-VirtualBox:~/servidorFTP$

```

- Muestra los contenedores en ejecución con docker-compose

Pista: docker-compose p.....

```

javier@javier-VirtualBox:~/servidorFTP$ docker-compose ps

```

Name	Command	State	Ports
servidorFtp2	/entry.sh vsftpd	Up	0.0.0.0:20->20/tcp, 0.0.0.0:21->21/tcp, 0.0.0.0:4559->4559/tcp, ::4559->4559/tcp, 0.0.0.0:4560->4560/tcp, ::4560->4560/tcp, 0.0.0.0:4561->4561/tcp, ::4561->4561/tcp, 0.0.0.0:4562->4562/tcp, ::4562->4562/tcp, 0.0.0.0:4563->4563/tcp, ::4563->4563/tcp, 0.0.0.0:4564->4564/tcp, ::4564->4564/tcp

```

javier@javier-VirtualBox:~/servidorFTP$

```

- Muestra los contenedores con docker

Pista: docker p.....

```
javier@javier-VirtualBox:~/servidorFTP$ docker ps
```

CONTAINER ID	IMAGE	PORTS	COMMAND	CREATED
9e455ef28db7	javersanchezgomez/vsftpd		"/entry.sh vsftpd"	2 minutes ago
Up 2 minutes	0.0.0.0:20-21->20-21/tcp, 0.0.0.0:4559-4564->4559-4564/tcp, :::4559-4564->4559-4564/tcp	servidorFtp2		
d467d3174b93	nginx:latest		"/docker-entrypoint..."	17 minutes ago
Up 17 minutes	0.0.0.0:8081->80/tcp, :::8081->80/tcp	servidorWeb1		

```
javier@javier-VirtualBox:~/servidorFTP$
```

-Muestra un listado de las redes

Pista: docker n.....

```
javier@javier-VirtualBox:~/servidorFTP$ docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
fba3b52ea595	bridge	bridge	local
2421371fde9d	host	host	local
d5b81c103f6b	javier_default	bridge	local
f4ec5778b3df	none	null	local
68226e72ae8e	red_contenedores	bridge	local

```
javier@javier-VirtualBox:~/servidorFTP$
```

- Conéctate al servidor ftp

Pista: ftp ip_servidor_ftp (No utilices localhost)

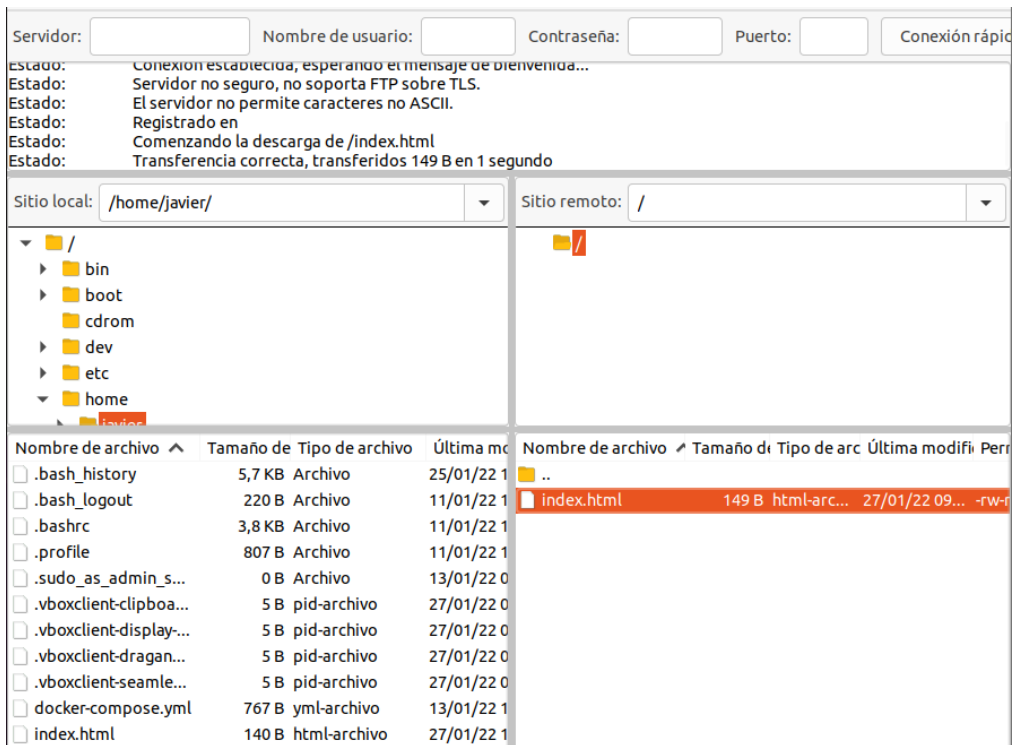
```
javier@javier-VirtualBox:~/servidorFTP$ ftp 192.168.222.99
Connected to 192.168.222.99.
220 (vsFTPd 3.0.3)
Name (192.168.222.99:javier): asir222
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
```

- Muestra el contenido de directorio raíz

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp      149 Jan 27 08:02 index.html
226 Directory send OK.
ftp>
```

Nota: Si no puedes mostrarlo, activa el modo pasivo en el cliente

- Descarga el index.html



- Modifica el index.html

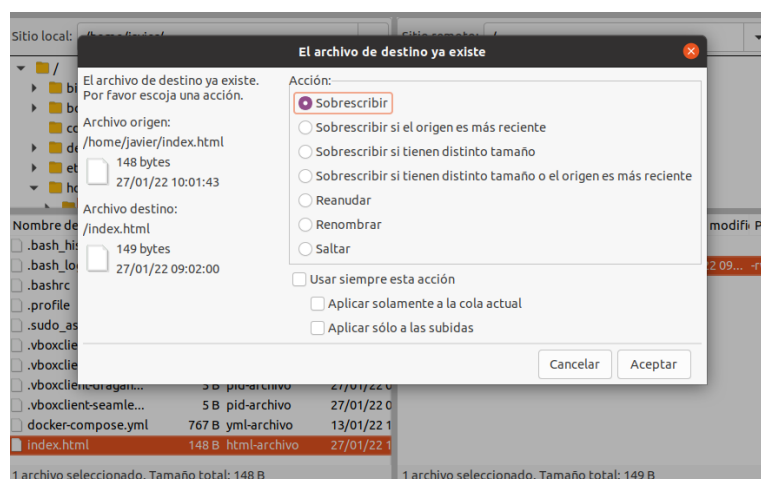
index.html (modificado de nuevo):

```
<!DOCTYPE html>
<html>
<head>
<title>Servidor web 1</title>
</head>
<body>
<h1>Bienvenido al servidor web 1 REQUETE DEFINITIVO</h1>
</body>
</html>
```



```
javier@javier-VirtualBox: ~
GNU nano 4.8 index.html Modificado
<!DOCTYPE html>
<html>
<head>
<title>Servidor web 1</title>
</head>
<body>
<h1>Bienvenido al servidor web 1 REQUETE DEFINITIVO</h1>
</body>
</html>
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Text ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^T Ortografía ^I Ir a línea
```

- Sube el index.html



Nota: En el caso de que no te deje subir el archivo index.html, accede al contenedor, *servidorFtp2*, y establecer como usuario y grupo propietario del directorio raíz *./srv*, a *ftp:ftp* al igual que hiciste en la 1ª y 2ª parte

- Establece las reglas DNAT adecuada en el router mikrotik para que el profesor pueda acceder a tus sitios FTP (*servidorFTP2*) y WEB (*servidorWeb1*). Cuando hayas configurado de manera adecuada el servicio avisa al profesor.

Firewall										
Filter Rules NAT Mangle Raw Service Ports Connections Address Lists Layer7 Protocols										
<div> <div> <div>+</div> <div>-</div> <div>✓</div> <div>✗</div> <div>📄</div> <div>🔍</div> </div> <div> <div>Reset Counters</div> <div>Reset All Counters</div> </div> <div> <div>Find</div> <div>all</div> <div>▼</div> </div> </div>										
#	Action	Chain	Src. Address	Dst. Address	Proto...	Src. Port	Dst. Port	In. Inter...	Out. Int...	li ▼
0	mas...	srcnat							wan	
1	dst...	dstnat			6 (tcp)		3389	wan		
2	dst...	dstnat			6 (tcp)		81	wan		
3	dst...	dstnat			6 (tcp)		443	wan		
4	dst...	dstnat			6 (tcp)		8080	wan		
5	dst...	dstnat			6 (tcp)		5000	wan		
6	dst...	dstnat			6 (tcp)		10000	wan		
7	dst...	dstnat			6 (tcp)		22	wan		
8	dst...	dstnat			6 (tcp)		4559-4564	wan		
9	dst...	dstnat			6 (tcp)		20-21	wan		
10	dst...	dstnat			6 (tcp)		8081	wan		
<div> <div>◀</div> <div></div> <div>▶</div> </div>										
11 items (3 selected)										

- Una vez haya realizado las comprobaciones adecuadas el profesor, ubícate en los directorios respectivos `·$HOME/servidorWeb1` y `$HOME/servidorFtp` y realiza las siguientes acciones:

Detén ambos escenarios.

Pista: `docker-compose s....`

- Borra ambos escenarios

Pista: `docker-compose r....`

PRÁCTICA FINALIZADA