

Ejercicio 2 - Docker Compose

Adrián García de la Cera

Ejercicio 2 - Docker Compose

Enunciado

Desarrollo

Creación archivo

Inicio de servicio

Pruebas con filebrowser

Finalización del servicio

Enunciado

Explorar la imagen de la aplicación FileBrowser en este repositorio en GitHub: <https://hub.docker.com/r/hurlenko/filebrowser>

Escribir un fichero compose.yaml para desplegarla. Los datos se pueden guardar utilizando volúmenes o utilizando bind-mount.

Entregar, al menos, las siguientes capturas de pantalla y los comandos y/o operaciones con Docker Desktop empleados para resolver el ejercicio:

- Captura de pantalla y documento donde se vea el fichero docker-compose.yaml que has creado.
- Captura de pantalla donde se vean los volúmenes/carpetas donde se han almacenado los datos.
- Captura de pantalla donde se vea la aplicación funcionando, sube algún fichero, cambia el lenguaje a español...
- Explicar brevemente cómo funciona esta aplicación y qué hace.

Desarrollo

El desarrollo del ejercicio se va a realizar en una máquina virtual y lo dividiremos en partes, en la primera crearemos el fichero compose.yaml, y explicaremos sus partes, en la segunda iniciaremos el servicio accediendo a la web y mostrando su funcionamiento realizando cambio de idioma, etc y por último eliminaremos el contenedor.

Para entrar en contexto, filebrowser es un explorador de archivos web que como su web indica:

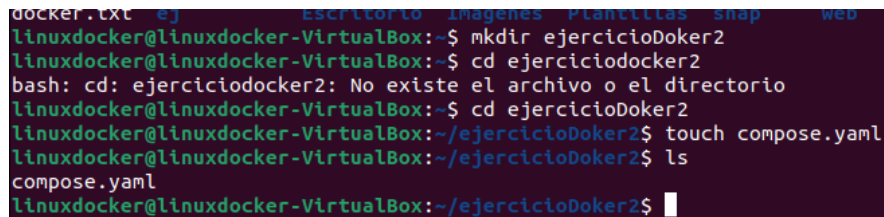
"filebrowser proporciona una interfaz de gestión de archivos dentro de un directorio especificado y se puede utilizar para cargar, eliminar, previsualizar, renombrar y editar sus archivos. Permite la creación de múltiples usuarios y cada usuario puede tener su propio directorio. Se puede utilizar como una aplicación independiente o como middleware."

Creación archivo

Lo primero que vamos a hacer es crear el directorio donde alojar el archivo compose.yaml y el propio archivo con los siguientes comandos:

Creamos el directorio, accedemos al mismo y creamos el archivo compose.yaml:

```
mkdir ejercicioDocker2
cd ejercicioDocker2
touch compose.yaml
```



```
docker.txt  Ej  Escritorio  Imágenes  Plantillas  Snap  Web
linuxdocker@linuxdocker-VirtualBox:~$ mkdir ejercicioDocker2
linuxdocker@linuxdocker-VirtualBox:~$ cd ejerciciodocker2
bash: cd: ejerciciodocker2: No existe el archivo o el directorio
linuxdocker@linuxdocker-VirtualBox:~$ cd ejercicioDocker2
linuxdocker@linuxdocker-VirtualBox:~/ejercicioDocker2$ touch compose.yaml
linuxdocker@linuxdocker-VirtualBox:~/ejercicioDocker2$ ls
compose.yaml
linuxdocker@linuxdocker-VirtualBox:~/ejercicioDocker2$
```

Una vez creado el archivo accedemos al mismo

```
nano compose.yaml
```

Una vez dentro completamos con el siguiente código:

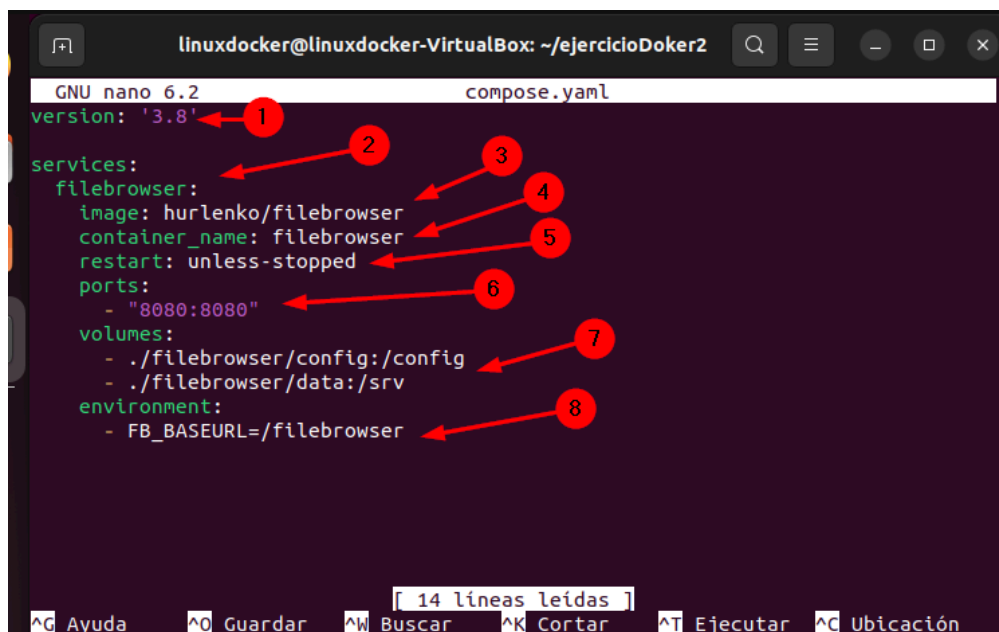
```
version: '3.8'

services:
  filebrowser:
    image: hurlenko/filebrowser
    container_name: filebrowser
    restart: unless-stopped
    ports:
      - "8080:8080"
    volumes:
      - ./filebrowser/config:/config
      - ./filebrowser/data:/srv
    environment:
      - FB_BASEURL=/filebrowser
```

El archivo compose.yaml se divide en varias partes:

1. La versión en este caso 3.8
2. Se definen los servicios (contenedores) que se van a desplegar en este caso filebrowser.
3. Se especifica la imagen del contenedor a utilizar.
4. Asignamos un nombre al contenedor.
5. Definimos la política de reinicio, para que se reinicie automáticamente si se detiene excepto si el usuario lo detiene manualmente.
6. Mapeamos el puerto 8080:8080 para acceder a la interfaz.

- Montamos los volúmenes de configuración y de archivos, para que cuando se elimine o reinicie el servicio no se pierda la información.
- Configuramos la URL base la aplicación.

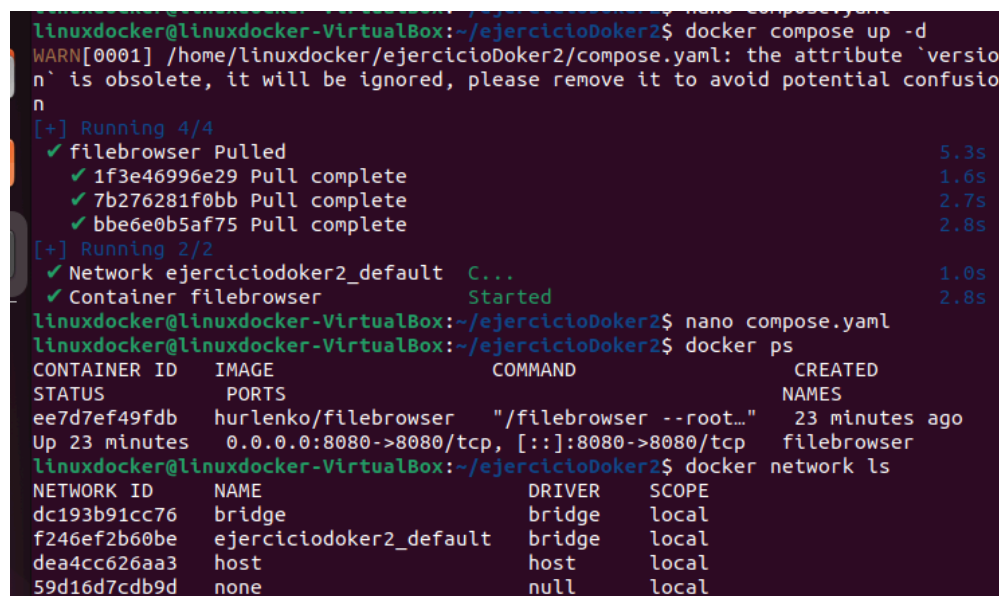


Una vez configurado el archivo compose.yaml y guardado ejecutamos el servicio en segundo plano.

Inicio de servicio

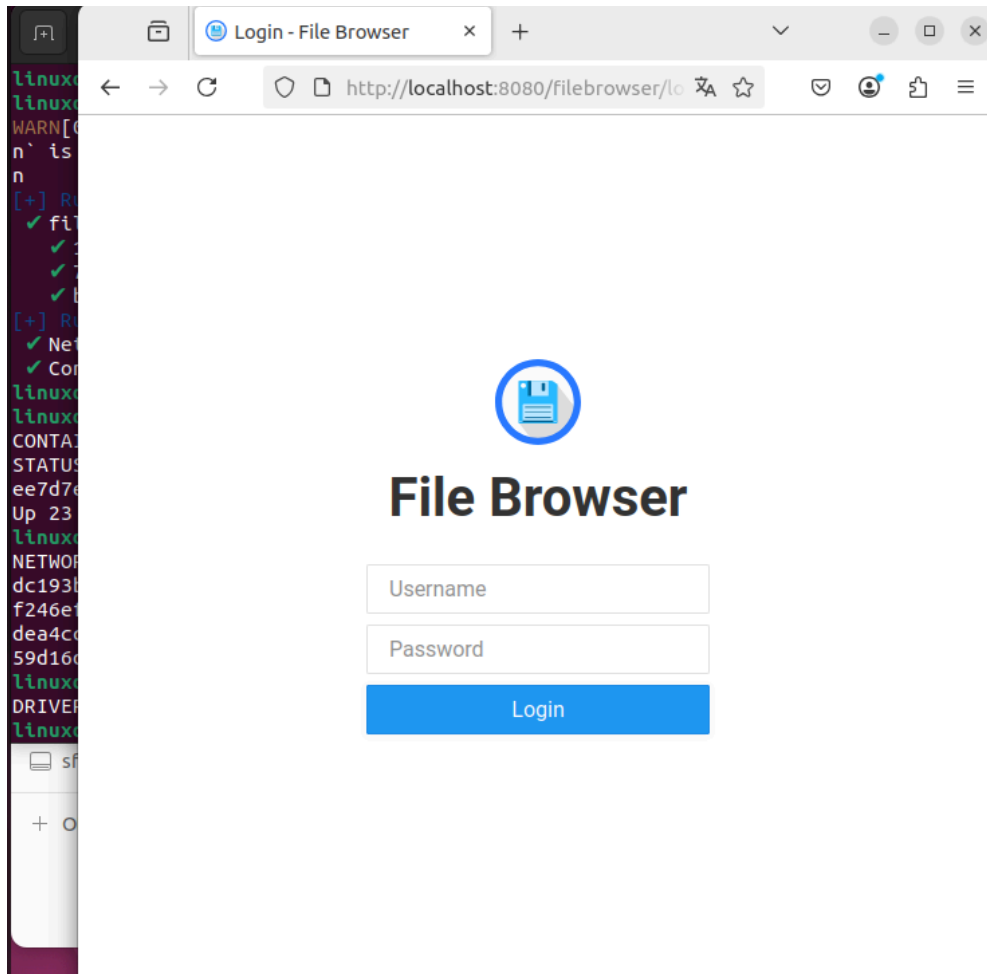
```
docker compose up -d
```

Esto descargará la imagen, creará la red, creará y arrancará el contenedor y montará los volúmenes.



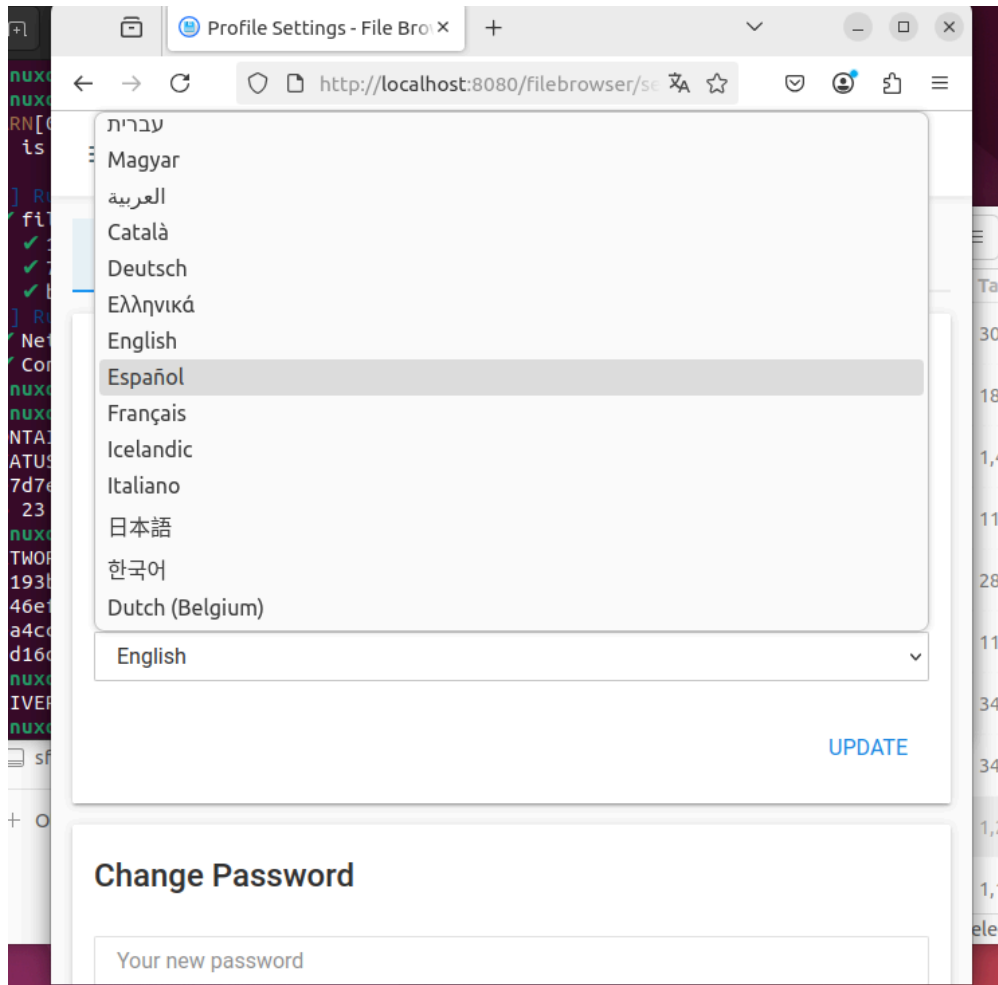
Ya creado el archivo compose.yaml y montado comprobamos si accedemos correctamente al servicio.

```
http://localhost:8080
```

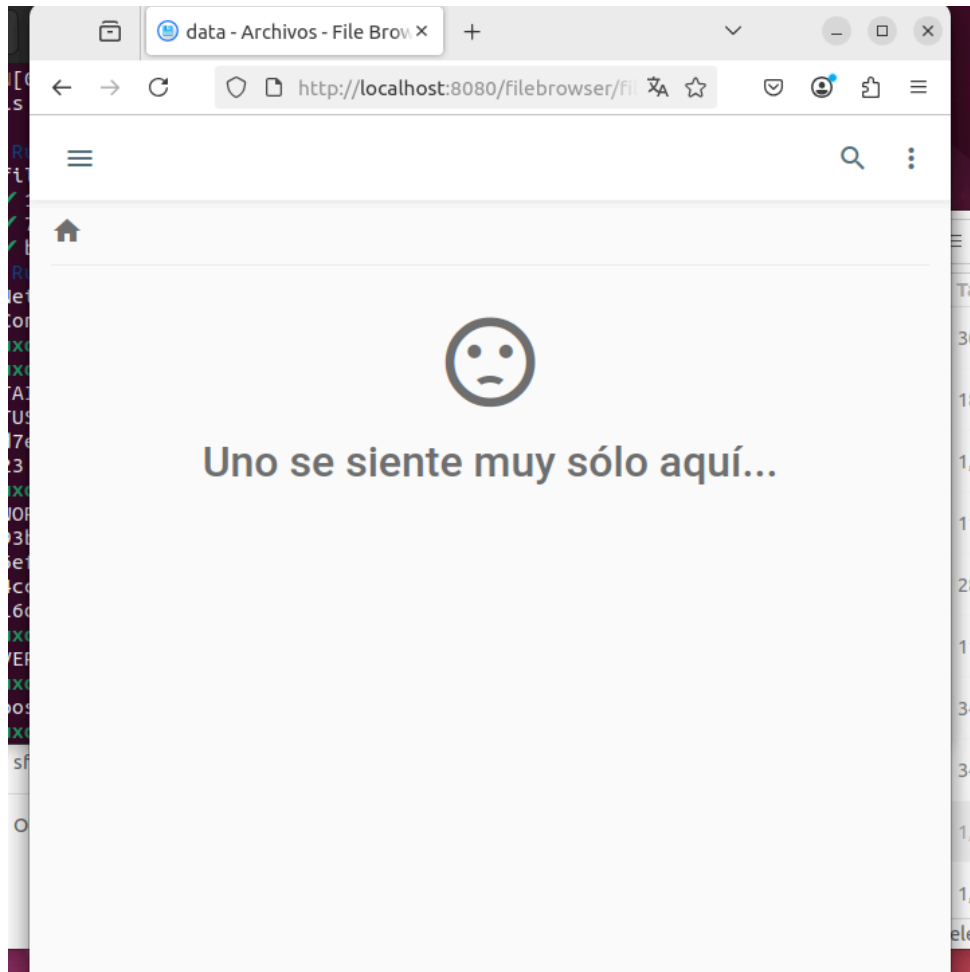


Pruebas con filebrowser

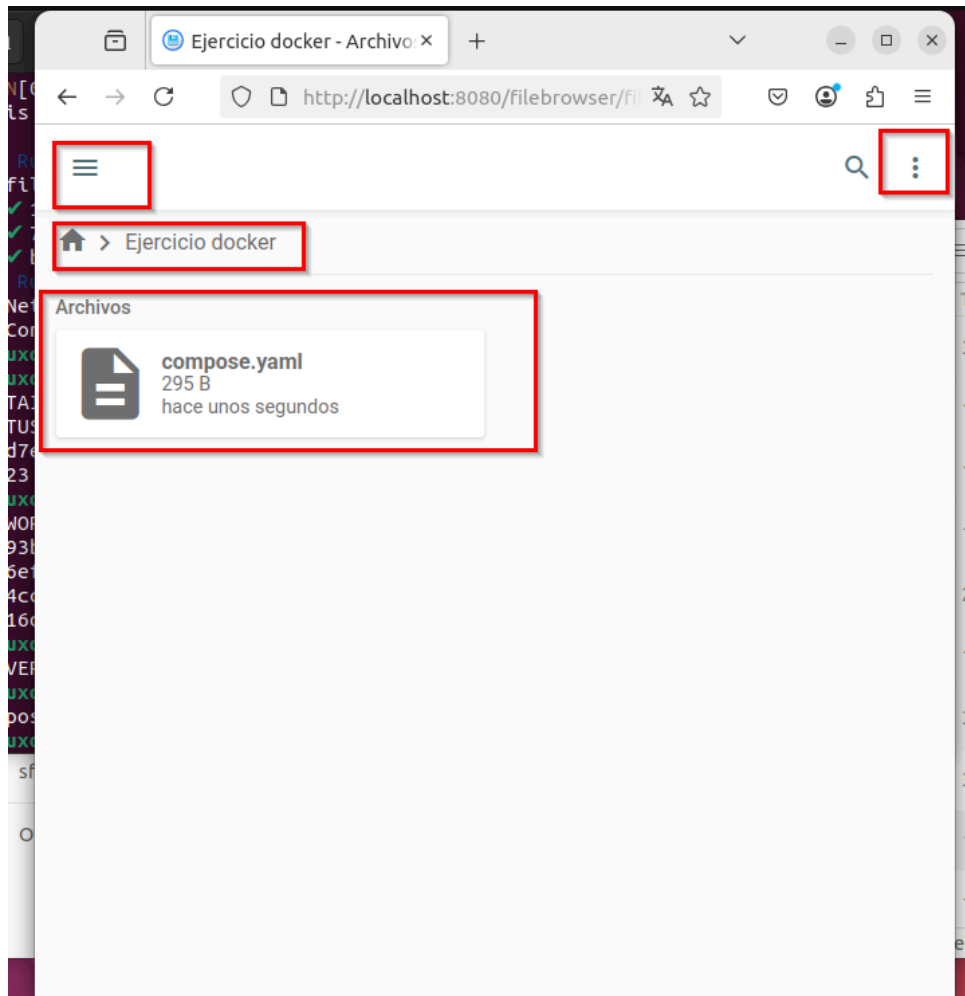
Nos logamos con las credenciales que se indican en el enunciado (user: admin password: admin) y procedemos a cambiar el idioma en Settings



Como vemos a continuación, de momento no tenemos ningún archivo subido a filebrowser:



Vamos a crear dentro de filebrowser una carpeta llamada "Ejercicio docker" , para ello, en la esquina superior derecha hacemos clic en las 3 barras para que nos despliegue el menú y seleccionamos nueva carpeta, lo que nos abrirá un cuadro para indicar el nombre de la misma, una vez creada directamente accede a la misma, ahora en la parte superior derecha damos a los 3 puntos y seleccionamos subir, nos abrirá el gestor de archivos de linux y seleccionamos el fichero a subir en nuestro caso hemos seleccionado el propio archivo compose.yaml.



Finalización del servicio

Para finalizar vamos a detener el servicio y eliminar usamos el comando

```
docker compose down
```

En la siguiente imagen vemos como tras detener el servicio, como se han usado volúmenes a persistido la información:

```
linuxdocker@linuxdocker-VirtualBox: ~/ejercicioDocker2$ tree
.
├── compose.yaml
└── filebrowser
    ├── config
    │   └── filebrowser.db
    └── data

3 directories, 2 files
```