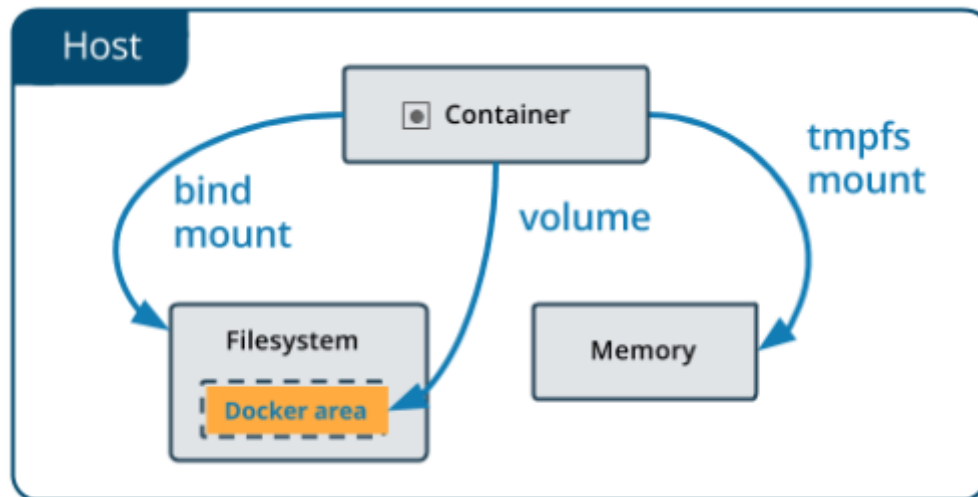


Practica 3 Volúmenes Docker



Practica 3 Volúmenes Docker

Vamos a trabajar con volúmenes docker:

1. Crea un volumen docker que se llame miweb.
2. Crea un contenedor desde la imagen php:7.4-apache donde montes en el directorio/var/www/html (que sabemos que es el DocumentRoot del servidor que nos ofrece esa imagen) el volumen docker que has creado.
3. Utiliza el comando docker cp para copiar un fichero index.html en el directorio/var/www/html.
4. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html.
5. Borra el contenedor.
6. Crea un nuevo contenedor y monta el mismo volumen como en el ejercicio anterior.
7. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html. ¿Seguía existiendo ese fichero?

Vamos a trabajar con bind mount:

1. Crea un directorio en tu host y dentro crea un fichero index.html.
2. Crea un contenedor desde la imagen php:7.4-apache donde montes en el directorio/var/www/html el directorio que has creado por medio de bind mount.
3. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html.

4. Modifica el contenido del fichero index.html en tu host y comprueba que al refrescar la página ofrecida por el contenedor, el contenido ha cambiado.
5. Borra el contenedor.
6. Crea un nuevo contenedor y monta el mismo directorio como en el ejercicio anterior.
6. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html. ¿Se sigue viendo el mismo contenido?.

Vamos a trabajar con volúmenes docker:

1. Crea un volumen docker que se llame miweb.

```
daw@daw-docker:~$ docker volume create miweb  
  
miweb
```

2. Crea un contenedor desde la imagen php:7.4-apache donde montes en el directorio /var/www/html (que sabemos que es el DocumentRoot del servidor que nos ofrece esa imagen) el volumen docker que has creado.

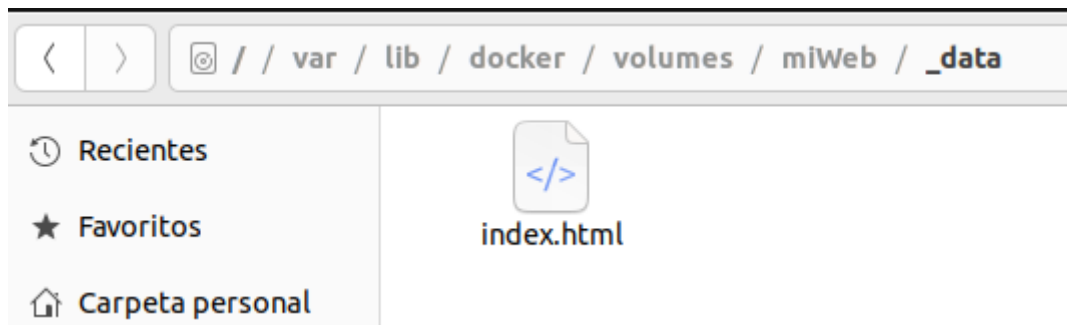
```
daw@daw-docker:~$ docker run -d --name miApachephp -v  
miweb:/var/www/html -p 8080:80 php:7.4-apache
```

```
daw@daw-docker:~$ docker ps
```

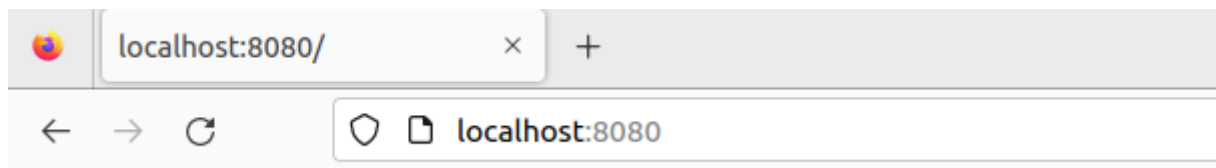
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
69444871ceb2	php:7.4-apache	"docker-php-entrypoi..."	50 seconds ago	Up 49 seconds
	0.0.0.0:8080->80/tcp	:::8080->80/tcp		
		miApachephp		

3. Utiliza el comando docker cp para copiar un fichero index.html en el directorio /var/www/html.

```
daw@daw-docker:~$ docker cp index.html miApachephp:/var/www/html
```



4. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero `index.html`.



5. Borra el contenedor.

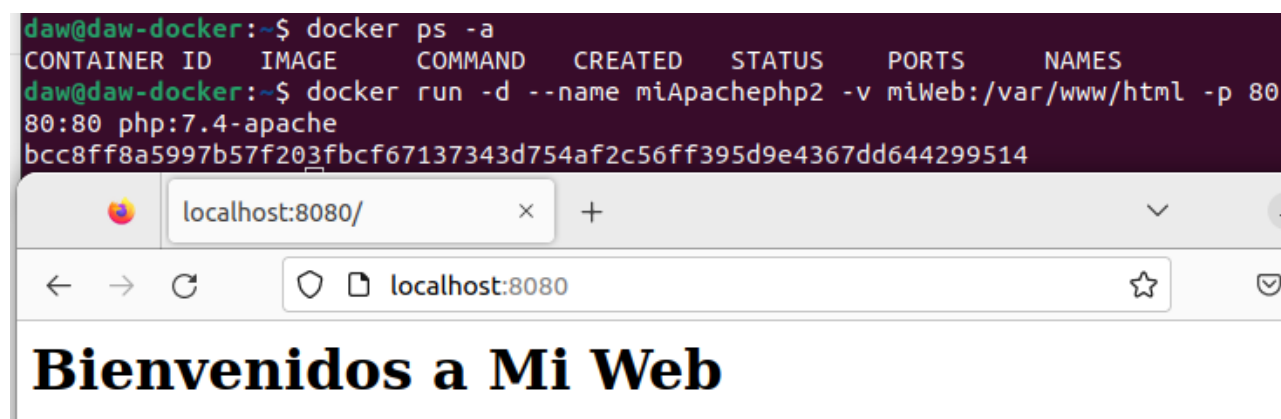
```
daw@daw-docker:~$ docker rm -f miApachephp
miApachephp
daw@daw-docker:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
NAMES					

6. Crea un nuevo contenedor y monta el mismo volumen como en el ejercicio anterior.

```
daw@daw-docker:~$ docker run -d --name miApachephp2 -v
miweb:/var/www/html -p 8080:80 php:7.4-apache
bcc8ff8a5997b57f203fbcf67137343d754af2c56ff395d9e4367dd644299514
```

7. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero `index.html`. ¿Seguía existiendo ese fichero?



Comprobamos que sigue existiendo el `index.html` ya que los volúmenes convierten los datos y hemos montado el contenedor con el volumen `miWeb`.

Vamos a trabajar con bind mount:

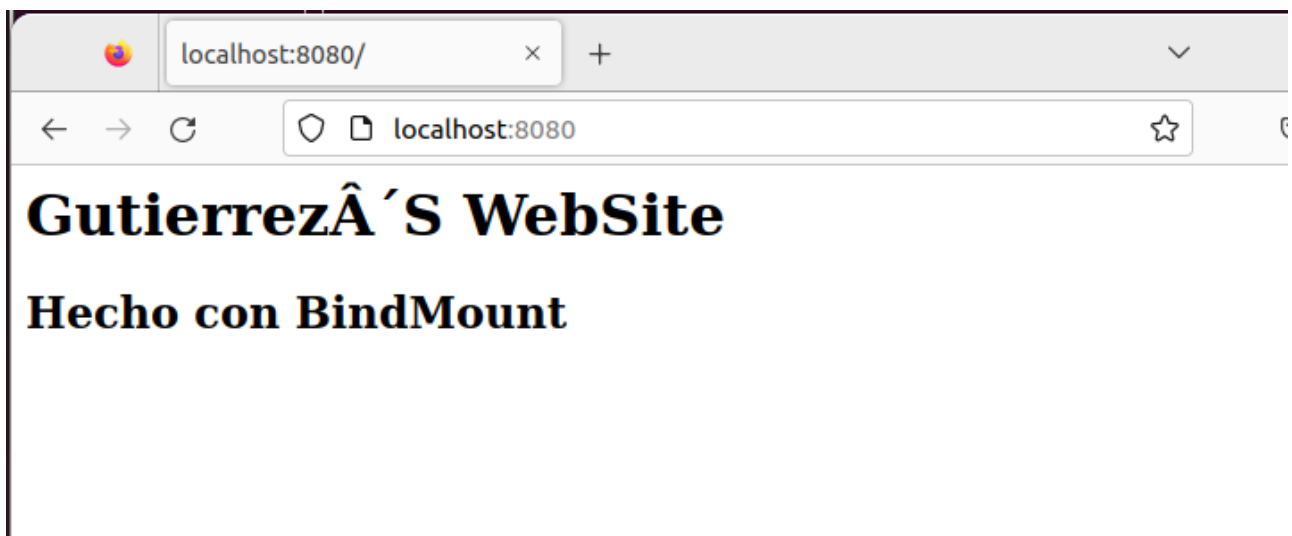
1. Crea un directorio en tu host y dentro crea un fichero `index.html`.

```
daw@daw-docker:~$ mkdir gutierrezweb
daw@daw-docker:~$ cd gutierrezweb/
daw@daw-docker:~/gutierrezweb$ nano index.html
daw@daw-docker:~/gutierrezweb$
```

2. Crea un contenedor desde la imagen `php:7.4-apache` donde montes en el directorio `/var/www/html` el directorio que has creado por medio de bind mount.

```
daw@daw-docker:~$ docker run -d --name miApacheBind -v
/home/daw/gutierrezweb:/usr/local/apache2/htdocs -p 8080:80
httpd:2.4
```

3. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero `index.html`.



4. Modifica el contenido del fichero `index.html` en tu host y comprueba que al refrescar la página ofrecida por el contenedor, el contenido ha cambiado.



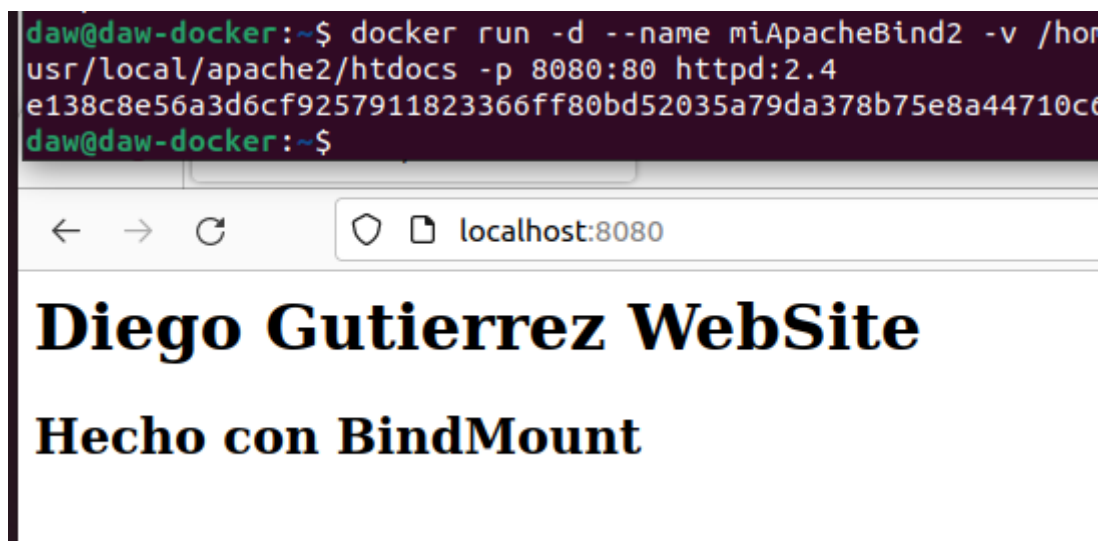
5. Borra el contenedor.

```
daw@daw-docker:~$ docker rm -f miApacheBind
miApacheBind
```

6. Crea un nuevo contenedor y monta el mismo directorio como en el ejercicio anterior.

```
daw@daw-docker:~$ docker run -d --name miApacheBind2 -v
/home/daw/gutierrezweb:/usr/local/apache2/htdocs -p 8080:80
httpd:2.4
```

6. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html. ¿Se sigue viendo el mismo contenido?.



Sigue mostrando el mismo contenido, porque seguimos teniendo la ruta con el index.html en el equipo y hemos montando el contenedor con BindMount.