



CFGs ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED



BIENVENIDOS!

Manuel Domínguez.



mftienda@gmail.com



@mafradoti



www.linkedin.com/in/mftienda



<https://github.com/mftienda>



Ud.- Comunicar contenedores

Índice

- 1.- Introducción.
- 2.- Volúmenes.
- 3.- Información de los volúmenes.
- 4.- Borrar volúmenes.
- 5.- Crear volúmenes.
- 6.- Crear un directorio compartido en el host.
- 7.- Crear un volumen independiente.



1.- Introducción

¿Podemos disponer de un almacenamiento compartido entre contenedores y el host?





2.- Volúmenes

Hasta ahora cuando borramos un contenedor también se borraba los datos.

Si tenemos datos valiosos los perdemos.

Tampoco podemos compartir los datos entre contenedores y el host.

Estos problemas se resuelven con los volúmenes.

Los **volúmenes** son la forma con la que Docker nos permite guardar los datos del contenedor. Esos datos podemos guardarlos en: Host remotos, NAS, Cloud, etc.



2.- Volúmenes

Volúmenes



- ☐ Con los volúmenes puedo
 - ☐ Usar un almacenamiento persistente para el contenedor
 - ☐ Compartir almacenamiento entre el **HQST** y un contenedor
 - ☐ Compartir almacenamiento entre distintos contenedores

Dentro del contenedor existe una carpeta llamada /datos. Esa carpeta realmente es un enlace a un espacio de almacenamiento que hemos llamado volume1 que está fuera del contenedor.

¿Qué ocurre si se borra el contenedor?



3.- Información de los volúmenes

¿Dónde se guardan los volúmenes?

/var/lib/docker/volumes/ID/_data

```
root@debian:~# cd /var/lib/docker/  
root@debian:/var/lib/docker# ls  
builder    containers  network    plugins    swarm      trust  
buildkit   image       overlay2   runtimes   tmp        volumes
```

Estos directorios no se deben tocar. Los maneja Docker.

#docker volume ls → Vemos los volúmenes

#docker volume inspect ID → Más información del volumen



4.- Borrar volúmenes

Borrar un volume

`#docker volume rm ID`

Borrar todos los volúmenes que no se usen

`#docker volume prune`



5.- Creación de volúmenes.

CREACIÓN DE UN VOLUMEN

docker run -it -v /datos --name mi-ubuntu ubuntu bash

-v → Volumen

/datos → Es la carpeta que se va a crear dentro del contenedor.

Como no hemos indicado donde se guarda. **Por defecto**, en el host, se guarda en:
/var/lib/docker/volumes → data

Dentro del contenedor podemos ver que se ha creado /datos

```
root@e06ad840b4a3:/# cd /
root@e06ad840b4a3:/# ls
bin      datos    etc      lib      lib64    media    opt      root    sbin    sys    usr
boot    dev      home    lib32    libx32   mnt      proc     run     srv     tmp    var
```



5.- Creación de volúmenes.

CREACIÓN DE UN VOLUMEN

Comprobación:

- 1.- Creamos en **el contenedor** un archivo dentro de /datos → prueba.txt
- 2.- Dentro del **host**, lo podemos ver en: /var/lib/docker/volumes/xxxxx/_data

Es bidireccional.

- 3.- En el **host**, creamos un archivo llamado ejercicio.txt
- 4.- Podremos verlo en **contenedor**.
- 5.- Podemos parar el contenedor y volver arrancarlo: **#docker start -i mi-ubuntu** , y el volumen sigue estando ahí.



6.- Crear un directorio compartido en el host.

CREACIÓN DE UN DIRECTORIO COMPARTIDO EN EL HOST

1.- Nos creamos un directorio en el host: **#mkdir /root/datos**

2.- Creamos el contenedor:

docker run -it -v /root/datos:/datos --name mi-ubuntu ubuntu bash

/root/datos → Es un directorio del host

/datos → Es un directorio del contenedor.

3.- Comprobaciones: Creamos archivos en el host y en el contenedor.

4.- Si vemos los volúmenes: **#docker volume ls** → No aparece. Porque realmente estamos compartiendo un directorio. Estamos haciendo un punto de montaje. Pero a todos los efectos funciona como un volumen. **#docker inspect mi-ubuntu** → bind (montaje)



7.-Crear un volumen independiente.

Crear un volumen independiente

```
# docker volume create volumen-1
```

```
# docker volume ls
```

```
# docker volume inspect volumen-1
```

Asociamos el volumen a un contenedor

```
# docker run -it --name mi-ubuntu -v volumen-1:/datos ubuntu bash
```



7.-Crear un volumen independiente.

Comprobaciones:

- 1.- Creamos un archivo en /datos→ prueba1.txt
- 2.- # ls /var/lib/docker/volumes/volumen-1/_data/
- 3.- #docker inspect mi-ubuntu → **Mounts**

Crear un volumen en modo solo lectura

#docker run -it --name mi-ubuntu-4 -v volumen-1:/datos:ro ubuntu bash

Comprobaciones: Intentamos crear un archivo.



7.-Crear un volumen independiente.

CONCLUSIONES:

Los volúmenes independientes son accesibles aunque no exista ningún contenedor asociado a ese volumen.

Lo mejor es trabajar con volúmenes independiente o directorios creados por nosotros.



Resumen

1.- ¿Qué es un volumen en Docker?



Sugerencias/mejoras del tema



Sugerencias /mejoras del tema

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |



Dudas





Referencias

- ❑ Albert Coronado
- ❑ Sergio de Apasoft