



PERSISTENCIA DATOS DOCKER

Javier Martínez Gallego

Contenido

- #Patrón general..... 2
- # Borrar un volumen 2
- #Borrar todos los volúmenes no usados..... 2
- #Listar volúmenes 3
- #Inspeccionar volumen 3
- # Aplicando a contenedores..... 5

#Patrón general

```
docker volume create [options] [volume]
```

Podemos crear un volumen con este comando, en options podemos configurar cosas como el driver, nombre, etc.

```
# Creación de un volumen llamado datos (driver local sin opciones)
> docker volume create data

# Creación de un volumen data especificando el driver local
> docker volume create -d local data

# Creación de un volumen llamando web añadiendo varios metadatos
> docker volume create --label servicio=http --label server=apache Web
```

Borrar un volumen

```
docker rm "volumen"
```

#Borrar todos los volúmenes no usados

```
docker prune
```

```
# Borrar un volumen por nombre
```

```
> docker volume rm nombre_volumen
```

```
# Borrar un volumen por ID
```

```
> docker volume rm
```

```
a5175dc955cfcf7f118f72dd37291592a69915f82a49f62f83666ddc81f67441
```

```
# Borrar dos volúmenes de una sola vez
```

```
> docker volume rm nombre_volumen1 nombre_volumen2
```

```
# Forzar el borrado de un volumen -f o --force
```

```
> docker volume rm -f nombre_volumen
```

#Listar volúmenes

Con docker volume ls vemos todos los volúmenes que tenemos

```
# Listar los volúmenes creados en el sistema
```

```
> docker volume ls
```

#Inspeccionar volumen

Con docker inspect vemos todos los datos de un volumen

```
> docker volume inspect  
a5175dc955cfcf7f118f72dd37291592a69915f82a49f62f83666ddc81f67441
```

Obtendremos una salida similar a la siguiente:

```
{  
  "CreatedAt": "2020-11-05T00:37:51+01:00",  
  "Driver": "local",  
  "Labels": null,  
  "Mountpoint": "/var/lib/docker/volumes/jenkins_home/_data",  
  "Name": "jenkins_home",  
  "Options": null,  
  "Scope": "local"  
}
```

Y la información que nos muestra es:

- La fecha de creación del volumen.
- El tipo del driver.
- Etiquetas asociadas.
- El punto de montaje.
- El nombre del volumen.
- Las opciones asociadas al driver.
- Y el ámbito del volumen.

Aplicando a contenedores

BIND MOUNT (flag -v): La carpeta web del usuario será el directorio raíz del servidor apache. Se crea si no existe

```
> docker run --name apache -v /home/usuario/web:/usr/local/apache2/htdocs  
-p 80:80 httpd
```

BIND MOUNT (flag --mount): La carpeta web del usuario será el directorio raíz del servidor apache. Se crea si no existe

```
> docker run --name apache -p 80:80 --mount  
type=bind,src=/home/usuario/web,dst=/usr/local/apache2/htdocs httpd
```

VOLUME (flag --mount). Mapear el volumen previamente creado y que se llama Data en la carpeta raíz del servidor apache

```
> docker run --name apache -p 80:80 --mount  
type=volume,src=Data,dst=/usr/local/apache2/htdocs httpd
```