Practica 6 – Docker Volumes

Markel Orallo Nogueira

2021

**Crea un volumen (demovol) e inspecciona su metadata. Intenta interpretar la**

**Información.**

**Texto

Descripción generada automáticamente**

Creamos el volumen “demovol” e inspeccionamos los metadatos mediante el comando docker volumen inspect. Nos da la siguiente información:

* CreatedAt: timestamp de creación del volumen.
* Driver: Que driver utiliza el volumen. En este caso usa el driver “local” que es el driver por defecto.
* Labels: Etiquetas que tiene el volumen. En este caso ninguna ya que no hemos especificado ningún etiquetado de dicho volumen.
* Mountpoint: Directorio local del host de docker donde se encuentra el volumen.
* Name: Nombre del volumen
* Scope: “local”. Indica que el volumen solo tiene que ser creado en el host local.

**Pon en marcha un contenedor que monte este volumen y mira el contenido del contenedor en el directorio.**

Ej. **$ docker container run -it -v demovol:/demo centos:7 bash**

**#ls**

**Una captura de pantalla de un celular

Descripción generada automáticamente**

**Encuentra el path donde está el directorio montado e interpreta la información que proporciona:**

**Texto

Descripción generada automáticamente**

El directorio donde encontramos nuestro volumen es /demo (tal y como hemos indicado al ejecutar el comando docker run)

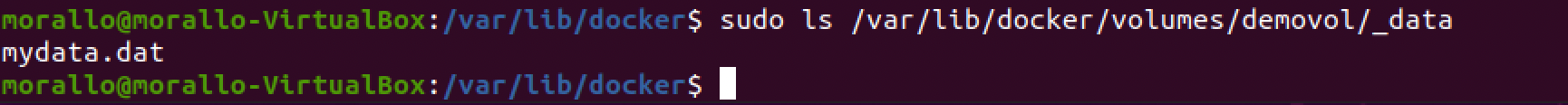
**Escribe un fichero en el volumen que hemos montado**

**Ej. echo 'dummy file' > /demo/mydata.dat**

**Texto

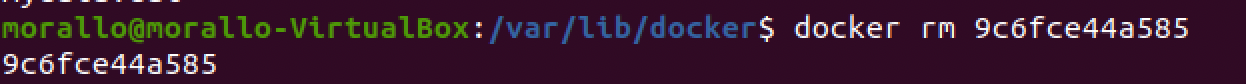
Descripción generada automáticamente**

**Sal del contenedor y lista el contenido del directorio donde hemos montado el volume (Mountpoint). Encuentras el mydata. dat?**

****

Como podemos mydata.dat esta en el mountpoint en el que hemos creado el volumen.

**Borra el contenedor**

****

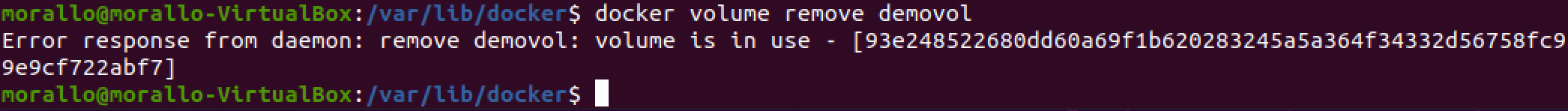
**Pon en marcha otro contenedor y monta el mismo volumen y únete a su shell. Comprueba que el fichero anterior sigue estando en el nuevo contenedor.**

**Texto

Descripción generada automáticamente**

Como podemos ver el fichero existe en el nuevo contenedor. Esto se debe a que los datos persisten en el host de docker al haber almacenado la información en un volumen.

**Intenta borrar el volumen. Te deja? Por qué?**



No deja borrar el volumen porque el ultimo contenedor que hemos creado tiene el volumen en uso. Hasta que el contenedor que esta utilizando el volumen no se elimine no nos dejara borrar el volumen.

**Para usuarios de OS X:**

Docker corre en una maquina virtual a la que podemos acceder a través del comando:

stty -echo -icanon && nc -U ~/Library/Containers/com.docker.docker/Data/debug-shell.sock && stty sane

Se nos abrirá una terminal dentro de la maquina virtual de docker. Alli encontraremos el directorio /var/lib/docker donde se guarda nuestro volumen(Se han seguido los mismos pasos que en Ubuntu para crear el volumen, por no poner dos veces lo mismo omito las capturas).

Texto

Descripción generada automáticamente

Nuestro dummy file:

Texto

Descripción generada automáticamente