



CFGS ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED



BIENVENIDOS!

Manuel Domínguez.



mftienda@gmail.com



@adoti



wnkedin.com/in/mftienda



<https://github.com/mftienda>



Ud.- Docker compose

Índice

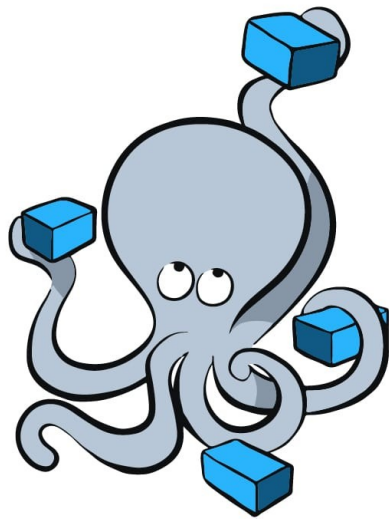
1.- Introducción

2.- Volúmenes



1.- Introducción

Escenarios multi-contenedor



docker
Compose



2.- Volúmenes

```
version: "3.2"
services:
  web:
    image: nginx:alpine
    volumes:
      - type: volume
        source: mydata
        target: /data
        volume:
          nocopy: true
      - type: bind
        source: ./static
        target: /opt/app/static
    ports:
      - 80:80
  db:
    image: postgres:latest
    volumes:
      - "/var/run/postgres/postgres.sock:/var/run/postgres/postgres.sock"
      - "dbdata:/var/lib/postgresql/data"

volumes:
  mydata:
  dbdata:
```



2.- Volúmenes

Ejemplos de Volúmenes:

Dentro del **contenedor web**, se ha definido **un volumen tipo volume**:

Se llama: mydata → Se guardará en /var/lib/docker/volume

Se vincula con el directorio del contenedor: /data

Se ha añadido una directiva: nocopy: true, para que no copie el directorio.

Se ha definido un **directorio compartido (Type bind)**

Directorio host: ./static

Directorio del contenedor: /opt/appr/static

```
web:
  image: nginx:alpine
  volumes:
    - type: volume
      source: mydata
      target: /data
      volume:
        nocopy: true
    - type: bind
      source: ./static
      target: /opt/appr/static
```



2.- Volúmenes

Ejemplos de Volúmenes:

```
db:
  image: postgres:latest
  volumes:
    - "/var/run/postgres/postgres.sock:/var/run/postgres/postgres.sock"
    - "dbdata:/var/lib/postgresql/data"
```

Otra forma definir los volúmenes:

Directorio compartido, en el primer caso. Indica el directorio del host y directorio de contenedor

Volumen compartido, en el 2º caso. Nombre del volumen, lo enlaza con un directorio del contenedor.



2.- Volúmenes

Ejemplos de Volúmenes:

Hay un apartado que pone volumes, que está fuera de servicios.

Sirve para declarar los volúmenes que se han utilizado.

```
services:
  web:
    image: nginx:alpine
    volumes:
      - type: volume
        source: mydata
        target: /data
        volume:
          nocopy: true
      - type: bind
        source: ./static
        target: /opt/app/s1
    ports:
      - 80:80
  db:
    image: postgres:latest
    volumes:
      - "/var/run/postgres,"
      - "dbdata:/var/lib/p"

volumes:
  mydata:
  dbdata:
```




2.- Volúmenes

Procedimiento:

Despliega los contenedores contenidos en el docker-compose.

- 1.- Crea previamente el directorio static, dentro del proyecto.
- 2.- #docker-compose up
- 3.- # docker volume ls → Vemos los volúmenes creados
- 4.-# docker-compose config --services → Vemos los contenedores.
- 5.- Nos conectamos al contenedor web: # docker-compose exec web sh
- 6.- Comprobamos el directorio compartido: /opt/app/static



2.- Volúmenes

Procedimiento:

Despliega los contenedores contenidos en el docker-compose.

Lo eliminamos todo: contenedores, volúmenes, redes, etc

```
/opt/app/static # exit
root@debian:~/volumen# docker-compose down
Stopping volumen_web_1 ... done
Removing volumen_web_1 ... done
Removing volumen_db_1 ... done
Removing network volumen_default
root@debian:~/volumen#
```



Bueno, los volúmenes se quedan huérfanos.



Resumen



Sugerencias/mejoras del tema



Sugerencias /mejoras del tema



Dudas

