

Ejercicio 2- Redes y Almacenamiento

Tengo que crear una red Bridge `bdnet` . Para ello escribo:

```
begogutierrez@oClienteLinux:~$ docker network create -d bridge bdnet
```

```

begogutierrez@oClienteLinux:~$ docker network create -d bridge bdnet
9fe10aa753c84170ea3512905376b2fde0129af624e25981fb4775ca9b7ad693
begogutierrez@oClienteLinux:~$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
9fe10aa753c8        bdnet               bridge              local
f24ff7c2b020        bridge              bridge              local
27050a182fa4        host                host                local
1ac705bb0ddd        letschat_default    bridge              local
df92beb5dd8a        none                null                local
20fb5d83a127        red1                bridge              local
cf334cdc7cf3        red_guestbook       bridge              local
begogutierrez@oClienteLinux:~$
```

Ahora tengo que crear un contenedor dentro de la red `bdnet` con una imagen `mariadb` . Este contenedor se tiene que ejecutar en segundo plano y será accesible desde el puerto 3306. Definiré la contraseña del usuario `root` y un volumen de datos persistente.

```

begogutierrez@oClienteLinux:~$ docker run --detach --name bbdd -p 3306:3306 -v
/home/$USER/datadir:/var/lib/mysql --env MARIADB_USER=daw --env MARIADB_PASSWORD=laboral1 --
env MARIADB_ROOT_PASSWORD=root --env MARIADB_DATABASE=base1 --network bdnet mariadb:latest

```

```


begogutierrez@oClienteLinux:~$ docker run --detach --name bbdd -p 3306:3306 -v /home/$USER/datad
ir:/var/lib/mysql --env MARIADB_USER=daw --env MARIADB_PASSWORD=laboral1 --env MARIADB_ROOT_PASS
WORD=root --env MARIADB_DATABASE=base1 --network bdnet mariadb:latest
57e6fc42b63dd89b0f7dbd4235e40a982b130b2800327740e38334323ec736e9
begogutierrez@oClienteLinux:~$ docker ps
CONTAINER ID   IMAGE                COMMAND                  CREATED        STATUS        PORTS
NAMES
57e6fc42b63d   mariadb:latest       "docker-entrypoint.s..." 25 seconds ago Up 24 seconds 0.0.0.
0:3306->3306/tcp, :::3306->3306/tcp   bbdd
begogutierrez@oClienteLinux:~$
```

A continuación debo crear un contenedor con el programa `Adminer` que se pueda conectar al contenedor de la base de datos.

En dockerhub encuentro la imagen del Adminer

2_Redes-Almacenamiento


Want faster and simpler Kubernetes development? Test out [Telepresence for Docker](#) today. ×






Explore Pricing Sign In [Register](#)


Ubuntu Software

Explore Official Images [adminer](#)

 **adminer**

 DOCKER OFFICIAL IMAGE ·  100M+ ·  826

Database management in a single PHP file.

[docker pull adminer](#) 

[Overview](#) [Tags](#)

Quick reference

- Maintained by:
[Tim Dusterhus \(of the Docker Community\)](#)
- Where to get help:
[the Docker Community Slack](#), [Server Fault](#), [Unix & Linux](#), or [Stack Overflow](#)

Supported tags and respective Dockerfile links

- [4.8.1-standalone](#), [4-standalone](#), [standalone](#), [4.8.1](#), [4](#), [latest](#)
- [4.8.1-fastcgi](#), [4-fastcgi](#), [fastcgi](#)

Quick reference (cont.)

Creo el contenedor consultando la documentación dentro de dockerhub para Adminer, uso el siguiente comando:

```
begogutierrez@oClienteLinux:~$ docker run --detach -p 8080:8080 -e  
ADMINER_DEFAULT_SERVER=bbdd --network bdnet adminer
```

```
begogutierrez@oClienteLinux:~$ docker run --detach -p 8080:8080 -e ADMINER_DEFAULT_SERVER=bbdd -  
-network bdnet adminer  
9127c0c62ef36c94283323bee62ba901e0c81ce2f0376f6c41fafc81f0d0ce15
```

Muestro el pantallazo de los contenedores creados y en ejecución.

2_Redes-Almacenamiento

```
begogutierrez@oClienteLinux:~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
9127c0c62ef3   adminer        "entrypoint.sh php -..." 16 seconds ago Up 14 seconds 0.0.0.
0:8080->8080/tcp, :::8080->8080/tcp   clever_torvalds
57e6fc42b63d   mariadb:latest "docker-entrypoint.s..." 11 minutes ago Up 11 minutes 0.0.0.
0:3306->3306/tcp, :::3306->3306/tcp   bbdd
begogutierrez@oClienteLinux:~$
```

Aquí podemos ver en el navegador el interfaz del Adminer.

Idioma: Español

Adminer 4.8.1

Login

Motor de base de datos	MySQL
Servidor	bbdd
Usuario	
Contraseña	
Base de datos	

☐ Guardar contraseña

Muestro como me conecto dentro de la base de datos.

Idioma: Español MySQL » bbdd Cerrar sesión

Adminer 4.8.1

Seleccionar Base de datos

[Crear Base de datos](#) [Privilegios](#) [Lista de procesos](#) [Variables](#) [Estado](#)

Versión MySQL: 5.5.5-10.11.2-MariaDB-1:10.11.2+maria~ubu2204 a través de la extensión de PHP MySQLi

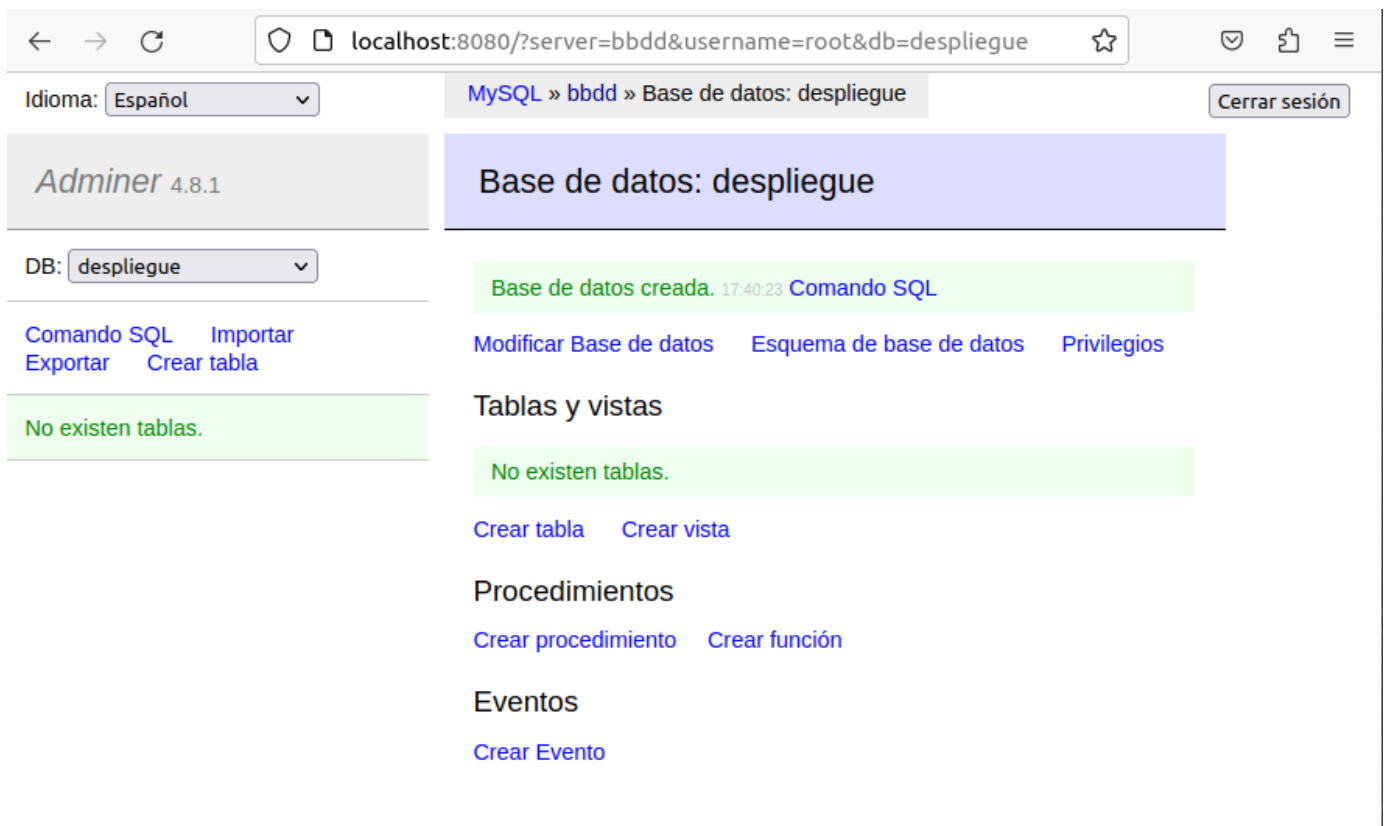
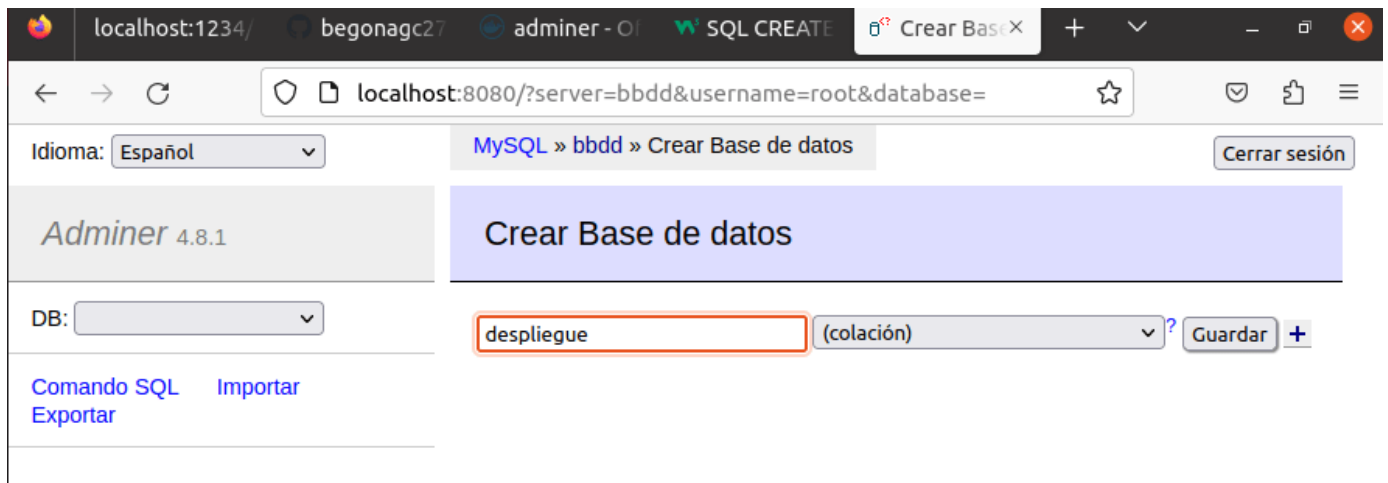
Logueado como: root@172.21.0.3

	Base de datos - Refrescar	Colación	Tablas	Size - Compute
<input type="checkbox"/>	base1	utf8mb4_general_ci	?	?
<input type="checkbox"/>	information_schema	utf8mb3_general_ci	?	?
<input type="checkbox"/>	mysql	utf8mb4_general_ci	?	?
<input type="checkbox"/>	performance_schema	utf8mb3_general_ci	?	?
<input type="checkbox"/>	sys	utf8mb3_general_ci	?	?

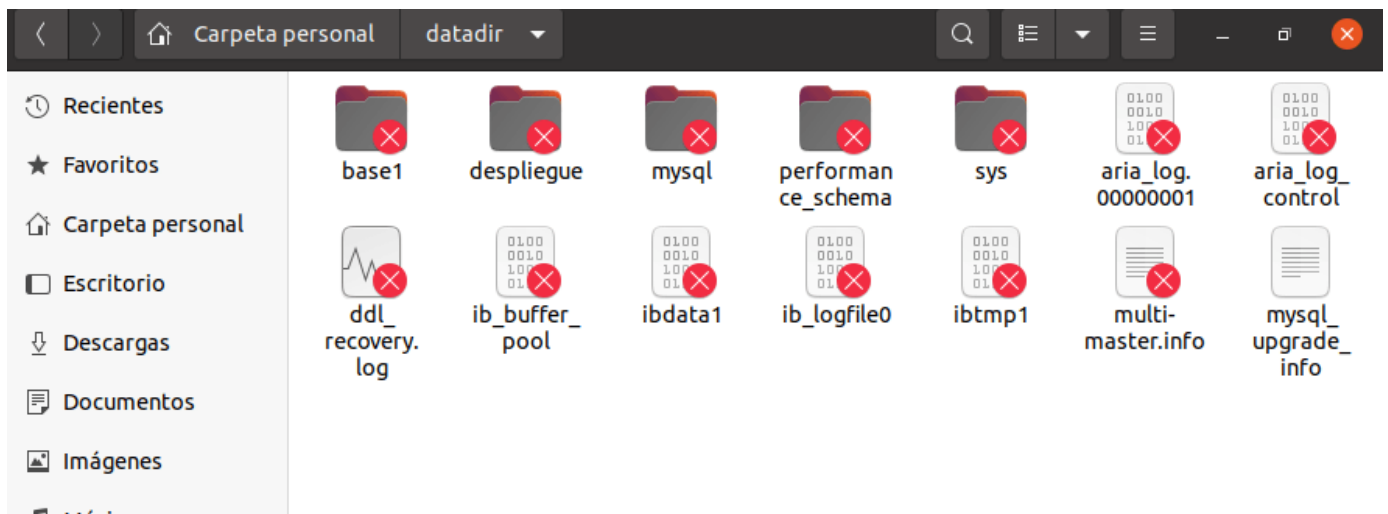
Selected (0)

Muestro como creo una nueva base de datos llamada `despliegue`

2_Redes-Almacenamiento



Ahora muestro los datos generados por el contenedor del servidor de la base de datos.



2_Redes-Almacenamiento

Ahora tengo que borrarlo todo, contenedores, red y volúmenes utilizados. Tengo varios volúmenes creados, puedo verlos con:

```
begogutierrez@oClienteLinux:~$ docker volume ls
```

```
begogutierrez@oClienteLinux:~$ docker volume ls
DRIVER      VOLUME NAME
local       0b03ee85ce965b659f0bcdb99729b7a847c492c11c65c23426c1ab7c9a2ab210
local       04fed56cefebe4fbf4aaeb25c75690932d3a9799d6384a95ce7a61cc209d5e4e
local       7d824332b851ce1fae9a0505d8a8195b80190663ee4b3c5e97e2f4adf5dd09be
local       8e49b270062f05eb97bc2b8e4f145e5a2ea3e90162f6785d9c1ac48d4a0d813c
local       9b83076754ed5f1f839b5f3b7d946cf296a370fe9def165ed2d4844ac672521d
local       11fb4d4babbbba1e5872e6b08997f9a7eb385cb6b31eb8a18a8c3a10d774308dd
local       27f7871af64f8b6ebe906ae0ca54a26dcd4af1795de6642e01544bcd28bd183
local       77e278b21173c2647bf5951e0323e8e2d9012361f34d620df606a5e3192249e2
local       86cdfb4950c906f79cd6f271c7bb01bb82350d2f704b7fad568544114f2607c4
local       761fb3977063bc02100959de3ed43db2c7f7303592359a90ef397621f8ccff
local       47165e3a72ff01e5a50790ae62dbf3b1926f515f0b9f241a97f5cae2f5ea5032
local       3528211b61a0e90e5b461eeaea77257f4d5c48cbcd2f1059cec2d497ac48a67e
local       d525677fa7cce2da6712e2d78ae1456e14fe5faf190943e2877dedcf789572f9
local       dc4c93dfc37fec9831492bfe4536d63c3a63727acf9f8166f5a158a19e129346
local       f77c36893ae2662c5b0765d72a01f66e00613d4fa8c91603e5d6ca8ed257e503
begogutierrez@oClienteLinux:~$
```

```
begogutierrez@oClienteLinux:~$ docker volume ls
begogutierrez@oClienteLinux:~$ docker stop clever_torvalds
begogutierrez@oClienteLinux:~$ docker rm bbdd
begogutierrez@oClienteLinux:~$ docker rm clever_torvalds
```

Puedo ver que no hay ninguno en ejecución:

```
begogutierrez@oClienteLinux:~$ docker ps
```

```
begogutierrez@oClienteLinux:~$ docker stop bbdd
bbdd
begogutierrez@oClienteLinux:~$ docker stop Adminer
Error response from daemon: No such container: Adminer
begogutierrez@oClienteLinux:~$ docker stop adminer
Error response from daemon: No such container: adminer
begogutierrez@oClienteLinux:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
9127c0c62ef3   adminer    "entrypoint.sh php -..." 11 minutes ago Up 11 minutes 0.0.0.0:8080->8080/tcp, ::8080->8080/tcp
clever_torvalds
begogutierrez@oClienteLinux:~$ docker stop clever_torvalds
clever_torvalds
begogutierrez@oClienteLinux:~$ docker rm bbdd
bbdd
begogutierrez@oClienteLinux:~$ docker rm clever_torvalds
clever_torvalds
begogutierrez@oClienteLinux:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
begogutierrez@oClienteLinux:~$
```

Ahora voy a borrar la red

```
begogutierrez@oClienteLinux:~$ docker network rm bdnet
```

En la imagen se puede ver las redes antes y después del borrado de `bdnet`.

2_Redes-Almacenamiento

```
begogutierrez@oClienteLinux:~$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
9fe10aa753c8        bdnet               bridge              local
f24ff7c2b020        bridge             bridge              local
27050a182fa4        host               host                local
1ac705bb0ddd        letschat_default   bridge              local
df92beb5dd8a        none               null                local
20fb5d83a127        red1               bridge              local
cf334cdc7cf3        red_guestbook      bridge              local
begogutierrez@oClienteLinux:~$ docker network rm bdnet
bdnet
begogutierrez@oClienteLinux:~$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f24ff7c2b020        bridge             bridge              local
27050a182fa4        host               host                local
1ac705bb0ddd        letschat_default   bridge              local
df92beb5dd8a        none               null                local
20fb5d83a127        red1               bridge              local
cf334cdc7cf3        red_guestbook      bridge              local
begogutierrez@oClienteLinux:~$
```

Como he usado `bind mounts` el borrado del volúmen persistente se hace desde el sistema operativo y no es necesario ningún comando `docker`.

Borro la carpeta `datadir` creada desde consola.

```
begogutierrez@oClienteLinux:~$ sudo rm -R datadir
```

```
begogutierrez@oClienteLinux:~$ ls
ActividadDocker  Descargas  Foto      libro-git  proyecto-git  recetas-ines  Videos
capitulos       Documentos Imágenes  Música     proyecto-git-remoto  repo1         web
datadir         Escritorio libro      Plantillas Público      snap
begogutierrez@oClienteLinux:~$ sudo rm -R datadir
[sudo] contraseña para begogutierrez:
Lo siento, pruebe otra vez.
[sudo] contraseña para begogutierrez:
begogutierrez@oClienteLinux:~$ ls
ActividadDocker  Documentos  Imágenes  Música     proyecto-git-remoto  repo1  web
capitulos        Escritorio  libro     Plantillas Público      snap
Descargas        Foto       libro-git proyecto-git recetas-ines  Videos
begogutierrez@oClienteLinux:~$
```