

### 3. Docker Compose Avanzado

Para gestionar múltiples contenedores con redes, volúmenes y dependencias de forma eficiente vamos a crear un archivo docker-compose.yml que defina varios servicios relacionados: una aplicación PHP, una base de datos y un servidor Nginx, incluyendo junto a estos volúmenes para persistencia de datos. Vamos a seguir los siguientes pasos:

#### 1. Creamos nuestro archivo docker-compose.yml en la versión 3.8

```
Sprint-1 > Ejercicio-3 > 🚀 docker-compose.yml
1  version: "3.8"
   ▶ Run All Services
2  services:
   ▶ Run Service
3      nginx:
4          image: nginx:latest
5          container_name: nginx
6          ports:
7              - "8080:80"
8          depends_on:
9              - php
10         networks:
11             - network
12
13         ▶ Run Service
14         php:
15             image: php:8.1-apache
16             container_name: php
17             networks:
18                 - network
19
20         ▶ Run Service
21         mysql:
22             image: mysql:8.0
23             container_name: mysql
24             volumes:
25                 - mysql-data:/var/lib/mysql
26             environment:
27                 MYSQL_ROOT_PASSWORD: rootpass
28                 MYSQL_DATABASE: mydb
29                 MYSQL_USER: myuser
30                 MYSQL_PASSWORD: password
31             networks:
32                 - network
33
34     volumes:
35         mysql-data:
36
37     networks:
38         network:
39             name: network
```

Vamos a ir explicando paso a paso el código del archivo:

- Vamos a ejecutar 3 servicios: nginx, php y mysql.

- **Nginx:**

- Usa la imagen más reciente de nginx.
- El nombre del contenedor es nginx.
- Se expone el puerto 80 del contenedor al puerto 8080 de la máquina local.
- Depende del servicio php, es decir, el contenedor php se levanta antes que el de nginx.
- Esta conectada a una red llamada network que permite que todos los contenedores conectados a esa misma red se comuniquen entre sí.

- **PHP:**

- Usa la imagen en la versión 8.1 de PHP junto al servidor web Apache.
- El nombre del contenedor es php.
- Esta conectada a una red llamada network que permite que todos los contenedores conectados a esa misma red se comuniquen entre sí.

- **MySQL:**

- Usa la imagen en la versión 8.0 de MySQL.
- El nombre del contenedor es mysql.
- Usa un volumen persistente llamado mysql-data montado en la ruta /var/lib/mysql
- Define variables de entorno para configurar el usuario y la base de datos.
- Esta conectada a una red llamada network que permite que todos los contenedores conectados a esa misma red se comuniquen entre sí.
- Definimos el volumen mysql-data.
- Definimos la red con el nombre network, dónde se conectarán todos los contenedores entre sí.

## 2. Comprobamos que los servicios estén funcionando

Tras crear el archivo docker-compose.yml, vamos a comprobar que los servicios se puedan lanzar con el comando:

```
docker compose up -d
```

```
anabel@ubuntu:~/Sprint-1/Ejercicio-3$ docker compose up -d
WARN[0000] /home/anabel/Sprint-1/Ejercicio-3/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 4/4
✔ Network network Created                                0.3s
✔ Container mysql Started                               1.1s
✔ Container php Started                                 1.0s
✔ Container nginx Started                               1.5s
anabel@ubuntu:~/Sprint-1/Ejercicio-3$
```

Tras lanzar los servicios, vamos a verificar que los contenedores estén en ejecución con el siguiente comando:

```
docker compose ps
```

```
anabel@ubuntu:~/Sprint-1/Ejercicio-3$ docker compose ps
WARN[0000] /home/anabel/Sprint-1/Ejercicio-3/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
NAME      IMAGE          COMMAND                  SERVICE  CREATED      STATUS      PORTS
mysql     mysql:8.0     "docker-entrypoint.s..." mysql    2 minutes ago Up 2 minutes 3306/tcp, 33060/tcp
nginx     nginx:latest   "/docker-entrypoint..." nginx    2 minutes ago Up About a minute 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp
php       php:8.1-apache "docker-php-entrypoi..." php      2 minutes ago Up 2 minutes 80/tcp
anabel@ubuntu:~/Sprint-1/Ejercicio-3$
```

Vamos ahora a ver con docker compose logs los registros que generan los contenedores, y así poder verificar que éstos funcionen. Para ello ejecutamos los siguientes comandos:

## docker compose logs nginx

```
anabel@ubuntu:~/Sprint-1/Ejercicio-3$ docker compose logs nginx
WARN[0000] /home/anabel/Sprint-1/Ejercicio-3/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
nginx | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
nginx | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
nginx | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
nginx | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
nginx | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
nginx | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
nginx | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
nginx | /docker-entrypoint.sh: Configuration complete; ready for start up
nginx | 2025/06/17 09:10:04 [notice] #1: using the "epoll" event method
nginx | 2025/06/17 09:10:04 [notice] #1: nginx/1.27.5
nginx | 2025/06/17 09:10:04 [notice] #1: built by gcc 12.2.0 (Debian 12.2.0-14)
nginx | 2025/06/17 09:10:04 [notice] #1: OS: Linux 6.11.0-26-generic
nginx | 2025/06/17 09:10:04 [notice] #1: getrlimit(RLIMIT_NOFILE): 1024:524288
nginx | 2025/06/17 09:10:04 [notice] #1: start worker processes
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 29
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 30
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 31
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 32
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 33
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 34
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 35
nginx | 2025/06/17 09:10:04 [notice] #1: start worker process 36
anabel@ubuntu:~/Sprint-1/Ejercicio-3$
```

```
#1: start worker processes
#1: start worker process 29
#1: start worker process 30
#1: start worker process 31
#1: start worker process 32
#1: start worker process 33
#1: start worker process 34
#1: start worker process 35
#1: start worker process 36
```

→ Podemos observar que Nginx se ejecutó de manera correcta.

## docker compose logs nginx

```
anabel@ubuntu:~/Sprint-1/Ejercicio-3$ docker compose logs php
WARN[0000] /home/anabel/Sprint-1/Ejercicio-3/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.20.0.3. Set the 'ServerName' directive globally to suppress thi
s message
php | AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.20.0.3. Set the 'ServerName' directive globally to suppress thi
s message
php | [Tue Jun 17 09:10:03.930105 2025] [mpm_prefork:notice] [pid 1:tid 1] AH00163: Apache/2.4.62 (Debian) PHP/8.1.32 configured -- resuming normal operations
php | [Tue Jun 17 09:10:03.930160 2025] [core:notice] [pid 1:tid 1] AH00094: Command line: 'apache2 -D FOREGROUND'
anabel@ubuntu:~/Sprint-1/Ejercicio-3$
```

```
Apache/2.4.62 (Debian) PHP/8.1.32 configured -
d line: 'apache2 -D FOREGROUND'
```

→ Podemos ver que PHP está usando Apache

## docker compose logs mysql

```
anabel@ubuntu:~/Sprint-1/Ejercicio-3$ docker compose logs mysql
WARN[0000] /home/anabel/Sprint-1/Ejercicio-3/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
mysql | 2025-06-17 09:10:03+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.42-1.el9 started.
mysql | 2025-06-17 09:10:04+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
mysql | 2025-06-17 09:10:04+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.42-1.el9 started.
mysql | '/var/lib/mysql/mysql.sock' -> '/var/run/mysqld/mysqld.sock'
mysql | 2025-06-17T09:10:05.142888Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and will be removed in a future release. Please use SE
T GLOBAL host cache size=0 instead.
mysql | 2025-06-17T09:10:05.144822Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.42) starting as process 1
mysql | 2025-06-17T09:10:05.180273Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
mysql | 2025-06-17T09:10:05.764922Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
mysql | 2025-06-17T09:10:06.212542Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
mysql | 2025-06-17T09:10:06.212663Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this chann
el.
mysql | 2025-06-17T09:10:06.219020Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all
OS users. Consider choosing a different directory.
mysql | 2025-06-17T09:10:06.288672Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, socket: /var/run/mysqld/mysqlx.sock
mysql | 2025-06-17T09:10:06.288963Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.42' socket: '/var/run/mysqld/mysqld.sock' p
ort: 3306 MySQL Community Server - GPL.
anabel@ubuntu:~/Sprint-1/Ejercicio-3$
```

```
Entrypoint script for MySQL Server 8.0.42-1.el9 started.  
Switching to dedicated user 'mysql'  
Entrypoint script for MySQL Server 8.0.42-1.el9 started.  
[mysql@localhost ~]$
```

→ Podemos ver que el servidor MySQL se ha iniciado de manera correcta.

Finalmente, vemos que todo funciona correctamente mediante los logs.