



CICLO FORMATIVO DE GRADO SUPERIOR - TÉCNICO EN ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN REDES

ADMINISTRACIÓN DE SISTEMAS OPERATIVOS

Docker

Nombre y apellidos:
-Wuke Zhang

PRÁCTICA 2. Ejecución de contenedores en Docker

Durante la primera práctica, realizamos la instalación de Docker para la consola de comandos. Asimismo, realizamos las operaciones básicas para obtener imágenes de contenedores y gestionarlos. Durante esta práctica, aprenderemos a ejecutar contenedores individualmente, teniendo en cuenta todas sus posibilidades.

1. Desplegar un contenedor. (1.25 pts)

Para desplegar un contenedor basado en una imagen, se utiliza el comando *docker run*. Ejecuta el siguiente comando y describe el resultado.

```
root@iso:/home/ubuntu/Desktop# docker run ubuntu
```

```
/mnt/c/Users/34640/Desktop (6.883s)
docker run ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
5a7813e071bf: Download complete
Digest: sha256:72297848456d5d37d1262630108ab308d3e9ec7ed1c3286a32fe09856619a782
Status: Downloaded newer image for ubuntu:latest

/mnt/c/Users/34640/Desktop (0.676s)
docker run ubuntu

✦ Run a different command in the ubuntu container. Ctrl Shift Enter

/mnt/c/Users/34640/Desktop Ctrl I Agent Mode
docker run -it ubuntu bash
```

no pasa nada, bueno si que entrarias a bash pero cosas de las nuevas terminales que ya vienen con bash e ia incluida.

```

/mnt/c/Users/34640/Desktop (0.662s)
docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

```

otro ejemplo es este que lo usas para ver si tienes ya el daemon

Ejecuta el siguiente comando y describe el resultado.

```
root@iso:/home/ubuntu/Desktop# docker run mysql
```

```

/mnt/c/Users/34640/Desktop (31.481s)
docker run mysql
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
43759093d4f6: Download complete
f56a22f949f9: Download complete
893b018337e2: Download complete
df1ba1ac457a: Download complete
277ab5f6ddde: Download complete
2be0d473cadf: Download complete
ce9646b00259: Download complete
d255dceb9ed5: Download complete
23d22e42ea50: Download complete
431b106548a3: Download complete
Digest: sha256:146682692a3aa409eae7b7dc6a30f637c6cb49b6ca901c2cd160becc81127d3b
Status: Downloaded newer image for mysql:latest
2025-02-21 23:40:09+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.
2025-02-21 23:40:10+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2025-02-21 23:40:10+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.
2025-02-21 23:40:10+00:00 [ERROR] [Entrypoint]: Database is uninitialized and password option is not specified
You need to specify one of the following as an environment variable:
- MYSQL_ROOT_PASSWORD
- MYSQL_ALLOW_EMPTY_PASSWORD
- MYSQL_RANDOM_ROOT_PASSWORD

/mnt/c/Users/34640/Desktop gpt-4o 1 block

🔗 Set the MYSQL_ROOT_PASSWORD environment variable for the MySQL container.

📌 Para iniciar un contenedor de MySQL, es necesario establecer una contraseña raíz utilizando la variable de entorno 'MYSQL_ROOT_PASSWORD'. Esto se puede hacer utilizando la opción '-e' en el comando 'docker run'.

✅ docker run -e MYSQL_ROOT_PASSWORD=my-secret-pw -d mysql

```

2. Revisar el estado los contenedores. (1 pts)

Para comprobar el estado de los contenedores existentes, se utiliza el comando *docker ps*. Ejecuta el siguiente comando y describe el resultado.

```
root@iso:/home/ubuntu/Desktop# docker ps
```

```
El contenedor MySQL se ha iniciado correctamente con la variable de entorno 'MYSQL_ROOT_PASSWORD' establecida. El identificador del contenedor es 'bcd6346f914e26af25aaef413b9bce27797da7c5ffafa57cd157b0e0bf57a92c'. Puedes verificar su estado usando el comando 'docker ps' para asegurarte de que se está ejecutando correctamente.
```

```
/mnt/c/Users/34640/Desktop gpt-4o
```

```
docker ps
```

Quieres listar los contenedores en ejecución actualmente para verificar el estado de tu contenedor MySQL. Procederé a ejecutar el comando 'docker ps' para mostrar esta información.

```
✓ docker ps
```

El contenedor MySQL está en ejecución correctamente. Aquí está el detalle del contenedor:

```
- **Container ID:** bcd6346f914e
- **Image:** mysql
- **Command:** "docker-entrypoint.s..."
- **Created:** 20 seconds ago
- **Status:** Up 20 seconds
- **Ports:** 3306/tcp, 33060/tcp
- **Name:** awesome_turing
```

Esto indica que el contenedor está activo y los puertos de MySQL están disponibles para las conexiones. Si necesitas conectarte a la base de datos, puedes hacerlo a través del puerto 3306.

Ahora ejecuta la siguiente variante del comando y describe el resultado. ¿Qué le sucedió al contenedor de Ubuntu que habíamos lanzado?

```
root@iso:/home/ubuntu/Desktop# docker ps -a
```

```
/mnt/c/Users/34640/Desktop gpt-4o
```

```
docker ps -a
```

Quieres listar todos los contenedores, incluidos aquellos que no están actualmente en ejecución. Ejecutaré el comando 'docker ps -a' para obtener esta información.

```
✓ docker ps -a
```

Aquí está la lista de todos los contenedores, incluidos los que están detenidos:

```
- **awesome_turing** (mysql): En ejecución (Up About a minute), expone los puertos 3306/tcp, 33060/tcp.
- **relaxed_payne** (mysql): Detenido (Exited (1)), se detuvo hace 2 minutos.
- **great_babbage** (hello-world): Detenido (Exited (0)), se detuvo hace 3 minutos.
- **serene_goodall** (ubuntu): Detenido (Exited (0)), se detuvo hace 4 minutos.
- **interesting_driscoll** (ubuntu): Detenido (Exited (0)), se detuvo hace 4 minutos.
- **mystifying_euler** (hello-world): Detenido (Exited (0)), se detuvo hace 5 minutos.
- **vibrant_yonath** (hello-world): Detenido (Exited (0)), se detuvo hace 5 minutos.
```

El contenedor 'awesome_turing' que corre con MySQL está activo. Los otros contenedores están detenidos, y algunos pertenecen a imágenes de prueba como 'hello-world' o 'ubuntu'. El contenedor 'relaxed_payne' también fue un intento de ejecutar MySQL, que falló al iniciarse.

3. Inspeccionar imágenes y contenedores. (1.5 pts)

Como hemos comprobado, al desplegar el contenedor de Ubuntu este termina automáticamente, mientras que el de mysql produce un error de ejecución.

Si nos fijamos en la información que devuelve `docker ps -a`, aparece que el comando que se ha ejecutado ha sido "Docker-entrypoint.sh" y "/bin/bash".

```
root@iso:/home/ubuntu/Desktop# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND
c049fb7e9a0e	mysql	"docker-entrypoint.s..."
1b9b6f953153	ubuntu	"/bin/bash"

Ejecuta el siguiente comando y localiza cual es el comando Entrypoint, ¿Para qué puede servir el comando asignado?

```
root@iso:/home/ubuntu/Desktop# docker inspect mysql
```

4. Ejecutar un contenedor interactivo. (1.25 pts)

Al desplegarse la imagen de Ubuntu en un contenedor, esta ejecuta directamente el intérprete de comando de la terminal, como un ordenador en modo comando. Sin embargo, a no tener un usuario activo en la terminal, esta se cierra automáticamente.

Un método para acceder a dicha terminal es utilizar contenedores interactivos. Ejecuta el siguiente comando y describe el resultado.

```
root@iso:/home/ubuntu/Desktop# docker run -it ubuntu
```

(Recuerda que, para salir de una sesión en la terminal, hay que ejecutar el comando *exit*).

```
docker inspect mysql

Estás intentando obtener información detallada sobre el contenedor o la imagen 'mysql'. Sin embargo, 'docker inspect' utilizado con un nombre de imagen como "mysql" devolverá detalles sobre dicha imagen. Si deseas información específica sobre el contenedor 'awesome_turing', necesitas utilizar el nombre del contenedor o el ID. Ejecutaré el comando 'docker inspect mysql' para obtener detalles sobre la imagen.

✓ docker inspect mysql

Aquí están los detalles de la imagen de Docker para MySQL que se inspeccionó:

- **ID de la imagen:** sha256:146682692a3aa409eae7b7dc6a30f637c6cb49b6ca901c2cd160becc81127d3b
- **Etiquetas del repositorio:** mysql:latest
- **Creado en:** 2025-01-22T17:15:22Z
- **Versión de Docker:** 27.4.0
- **Puertos Expuestos:** 3306/tcp, 33060/tcp
- **Comando predeterminado (Cmd):** mysqld
- **Entrypoint:** docker-entrypoint.sh
- **Versiones de MySQL:**
  - **MYSQL_MAJOR:** innovation
  - **MYSQL_VERSION:** 9.2.0-1.el9
  - **MYSQL_SHELL_VERSION:** 9.2.0-1.el9

La información indica que la imagen de MySQL está correctamente configurada para ejecutar 'mysqld' con los puertos estándar expuestos. La estructura de 'config' proporciona valores por defecto, comandos y variables de entorno esenciales para la operación de MySQL.
```

5. Ejecutar comandos al iniciar y ver registros de contenedores. (1 pts)

De la misma forma que podemos acceder a la terminal dentro de la máquina virtual, también podemos indicarle que ejecute un comando en esta nada más iniciarse, por ejemplo, el programa de inicio de una aplicación.

Ejecuta el siguiente código en la terminal y describe que sucede. Luego, inicia una segunda terminal en la máquina virtual y utiliza *docker logs* para ver los registros de ejecución del contenedor (Segunda imagen).

[illegible]

```
root@iso:/home/ubuntu/Desktop# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
f631769e812d   ubuntu   "bash -c 'while true..." 6 seconds ago  Up 5 seconds             distracted_gauss
root@iso:/home/ubuntu/Desktop# docker logs distracted_gauss

~ (0.164s)
docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
0a16cf04771f   ubuntu   "bash -c 'while true..." 28 seconds ago  Up 28 seconds             awesome_payne
bcd6346f914e   mysql    "docker-entrypoint.s..." 5 minutes ago  Up 5 minutes   3306/tcp, 33060/tcp      awesome_turing
```

6. Detener y reiniciar contenedores. (1.25 pts)

Para poder detener el contenedor, puedes ejecutar el comando *docker stop* como ves en la siguiente captura.

```
root@iso:/home/ubuntu/Desktop# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
f631769e812d   ubuntu   "bash -c 'while true..." About a minute ago Up About a minute   distracted_gauss
root@iso:/home/ubuntu/Desktop# docker stop distracted_gauss

~ (10.423s)
docker stop awesome_payne
awesome_payne

~ (0.074s)
docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
bcd6346f914e   mysql    "docker-entrypoint.s..." 6 minutes ago Up 6 minutes   3306/tcp, 33060/tcp   awesome_turing
```

Asimismo, para volver a ponerlo en marcha, utiliza el comando de la siguiente captura. ¿ha acaparado el contenedor la terminal esta vez?

```
root@iso:/home/ubuntu/Desktop# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
f631769e812d   ubuntu   "bash -c 'while true..." 4 minutes ago Exited (137) 2 minutes ago   distracted_gauss
root@iso:/home/ubuntu/Desktop# docker start distracted_gauss
distracted_gauss
```

```
hello world
hello world
hello world
hello world
hello world
hello world
hello world

/mnt/c/Users/34640/Desktop
exit
logout
```

7. Lanzar contenedores en hilos separados. (1 pts)

Como has visto, si ejecutamos un *docker run* directamente, el contenedor se lanzará como parte del hilo que lleva la ejecución de la terminal y no podremos seguir usándola.

```
root@iso:/home/ubuntu/Desktop# docker run -d ubuntu bash -c "while true; do echo hello world; sleep 1; done"
```

De esta forma, se puede seguir ejecutando código en la terminal sin inconvenientes (por ejemplo, el resto de un script).

```
pepe123@WUKE-PC: /mnt/c/Users/34640/Desktop$ docker run -d ubuntu bash -c "while true; do echo hello world; sleep 1; done"
83316c6c4e47242680c6b940e7a65145fc1eaa4c5cb1cfb4568b58b70637a1d7
pepe123@WUKE-PC: /mnt/c/Users/34640/Desktop$
```

8. Eliminar contenedores. (1.75 pts)

Asimismo, para eliminar un contenedor que no vamos a volver a utilizar podemos utilizar el siguiente comando:

```
root@iso:/home/ubuntu/Desktop# docker rm distracted_gauss
```

```
/mnt/c/Users/34640/Desktop (0.065s)
docker rm hopeful_robinson
Error response from daemon: cannot remove container "/hopeful_robinson": container is running: stop the container before removing or force remove

/mnt/c/Users/34640/Desktop (10.449s)
docker stop hopeful_robinson
hopeful_robinson

/mnt/c/Users/34640/Desktop (0.064s)
docker stop hopeful_robinson
hopeful_robinson
```

Como has podido comprobar, no se puede eliminar un contenedor en marcha, como medida de seguridad. Para hacerlo, tienes que primero detener el contenedor con *docker stop* o bien puedes forzarlo como en la siguiente captura. Pruébalo.

```
root@iso:/home/ubuntu/Desktop# docker rm -f distracted_gauss
distracted_gauss
```

```
/mnt/c/Users/34640/Desktop (0.124s)
docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS          NAMES
83316c6c4e47   ubuntu    "bash -c 'while true..." About a minute ago Up About a minute             hopeful_robinson
273566ba0408   mysql     "docker-entrypoint.s..." 11 minutes ago Exited (1) 11 minutes ago             relaxed_payne
a29b76403447   hello-world "/hello"                12 minutes ago Exited (0) 12 minutes ago             great_babbage
410dd821c299   ubuntu    "/bin/bash"              13 minutes ago Exited (0) 13 minutes ago             serene_goodall
b226a3068833   ubuntu    "/bin/bash"              13 minutes ago Exited (0) 13 minutes ago             interesting_driscoll
d7e532c8788c   hello-world "/hello"                14 minutes ago Exited (0) 14 minutes ago             mystifying_euler
97dad5adc0b1   hello-world "/hello"                14 minutes ago Exited (0) 14 minutes ago             vibrant_yonath

/mnt/c/Users/34640/Desktop (0.115s)
docker rm relaxed_payne great_babbage serene_goodall interesting_driscoll mystifying_euler vibrant_yonath
relaxed_payne
great_babbage
serene_goodall
interesting_driscoll
mystifying_euler
vibrant_yonath

/mnt/c/Users/34640/Desktop (0.074s)
docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS          NAMES
83316c6c4e47   ubuntu    "bash -c 'while true..." 2 minutes ago Up 2 minutes             hopeful_robinson
```

Otro dato a aclarar es que el nombre de los contenedores son random si no lo nombramos manualmente.