

Introducción a Docker y Kubernetes

UD 03. Caso práctico 01

- Práctica de comandos en contenedor Docker



Autor: Sergi García Barea

Actualizado Septiembre 2025

Licencia




Reconocimiento – NoComercial - CompartirIgual (BY-NC-SA): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 **Importante**

 **Atención**

 **Interesante**

1. Introducción	3
2. Preparando el contenedor	3
3. Solucionando el ejercicio	4
4. Script del profesor para comprobar la práctica	5
5. Bibliografía	6

UD03. CASO PRÁCTICO 01

1. INTRODUCCIÓN

En este caso práctico, partimos de un supuesto donde un profesor ha mandado una práctica de comandos de Linux a sus alumnos: deben crear carpetas numeradas del 01 al 10 en “/root”.

Los alumnos, para familiarizarse con Docker, deben realizar la práctica dentro de un contenedor y asimismo, el profesor preparará un script que recibirá el identificador o nombre del script y escribirá “PRÁCTICA OK” si la práctica está hecha correctamente, “PRÁCTICA INCORRECTA” en caso contrario.

2. PREPARANDO EL CONTENEDOR

Crearemos un contenedor con la imagen base "ubuntu". Además, al crearlo, dejaremos lista una "shell" para instalar los programas pertinentes. Esto podemos hacerlo con la orden:

```
docker run -d --name ejercicio ubuntu tail -f /dev/null
```

Tras esto, el contenedor estará en marcha en segundo plano y lanza en el contenedor el comando “tail -f /dev/null”, que es un comando cuya ejecución no acaba nunca (para que mientras no paremos el contenedor, no se pare).

Con esto, podemos tanto parar el contenedor como ponerlo en marcha con los comandos “docker start” y “docker stop”.

```
docker stop ejercicio
docker start ejercicio
```

Con el comando “docker ps” puedes ver los contenedores en marcha del sistema

```
docker ps
```

y con el comando “docker ps -a” puedes ver todos los contenedores (tanto parados como en marcha).

```
docker ps -a
```

Una vez el contenedor está en marcha, podemos acceder a una Shell dentro del contenedor con el comando:

```
docker exec -it ejercicio bash
```

En algunos contenedores a veces no está disponible la Sh

```
docker exec -it ejercicio sh
```

Un ejemplo de esta ejecución:

```

alumno@alumno-virtualbox:~/Desktop$ docker run -d --name ejercicio ubuntu tail -f /dev/null
f95e77f187cb36eba9fd70642aa6bf6e5afb6fed84132f2cb59971c78f88edb8
alumno@alumno-virtualbox:~/Desktop$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
f95e77f187cb   ubuntu   "tail -f /dev/null"     3 seconds ago Up 2 seconds             ejercicio
alumno@alumno-virtualbox:~/Desktop$ docker stop f9
f9
alumno@alumno-virtualbox:~/Desktop$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
alumno@alumno-virtualbox:~/Desktop$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
f95e77f187cb   ubuntu   "tail -f /dev/null"     24 seconds ago Exited (137) 6 seconds ago             ejercicio
alumno@alumno-virtualbox:~/Desktop$ docker start f9
f9
alumno@alumno-virtualbox:~/Desktop$ docker exec -it f9 bash
root@f95e77f187cb:/#

```

3. SOLUCIONANDO EL EJERCICIO

La solución del ejercicio es tan simple como crear 10 carpetas manualmente (del 00 al 10).

Estas soluciones deben ejecutarse dentro del contenedor. (Es una simulación de lo que respondería el alumnado)

Algunas posibles soluciones algo más en ShellScript y Python 3:

Solución en ShellScript

```

#!/bin/bash

#Bucle que va del 1 al 10
for i in {1..10}
do
    #Si $i es menor que 10, metemos 0 a la izquierda
    if [ $i -lt 10 ]
    then
        mkdir 0$i
    else
        mkdir $i
    fi
done

```

Solución en Python 3 (nota, si la máquina no tiene Python 3, instalarlo con ***“apt install python3”***)

```

#!/usr/bin/python3

#Importamos bibliotecas de sistema
import os, sys

#Bucle que recorre 10 elementos

for x in range(1,11,1):
    #Si es menor que diez, metemos 0 a la izquierda

```

```
if x<10:
    os.mkdir("0"+str(x))
else:
    os.mkdir(str(x))
```

4. SCRIPT DEL PROFESOR PARA COMPROBAR LA PRÁCTICA

El profesor ha preparado un pequeño script para corregir la práctica. Este script se ejecuta en la máquina anfitriona, da por hecho que el contenedor ya está lanzado y recibe como parámetro el identificador o nombre del contenedor.

Este script se ejecuta FUERA del contenedor, en la máquina real.

Dicho script es el siguiente:

```
#!/bin/bash

#Bucle del 1 al 10
for i in {1..10}
do
    # Si es menor que 0, al ejecutar docker exec metemos 0 a la izq
    # el comando test -d comprueba si existe un directorio
    # $1 es el identificador o nombre del contenedor
    if [ $i -lt 10 ]
    then
        docker exec -it $1 test -d /root/0$i
    else
        docker exec -it $1 test -d /root/$i
    fi

    # La variable $? devuelve 0 si la orden se ha ejecutado
    correctamente,
    # o distinto de cero en caso contrario.
    # Al enlazar la terminal en docker exec, el valor de $? es
    # el valor de la ejecución de la orden en el contenedor
    if [ $? -ne 0 ]
    then
        echo "PRÁCTICA INCORRECTA"
        echo "ERROR EN PRUEBA ${i}"
        exit
    fi
done
echo "PRÁCTICA OK"
```

Un ejemplo de ejecución:

```
sergi@ubuntu:~$ docker start ejercicio
ejercicio
sergi@ubuntu:~$ ./probar.sh ejercicio
PRÁCTICA INCORRECTA
ERROR EN PRUEBA 5
sergi@ubuntu:~$
```

5. BIBLIOGRAFÍA

- [1] Docker Docs <https://docs.docker.com/>
- [2] Python for SysAdmins <https://github.com/tuladhar/Python-for-SysAdmin>