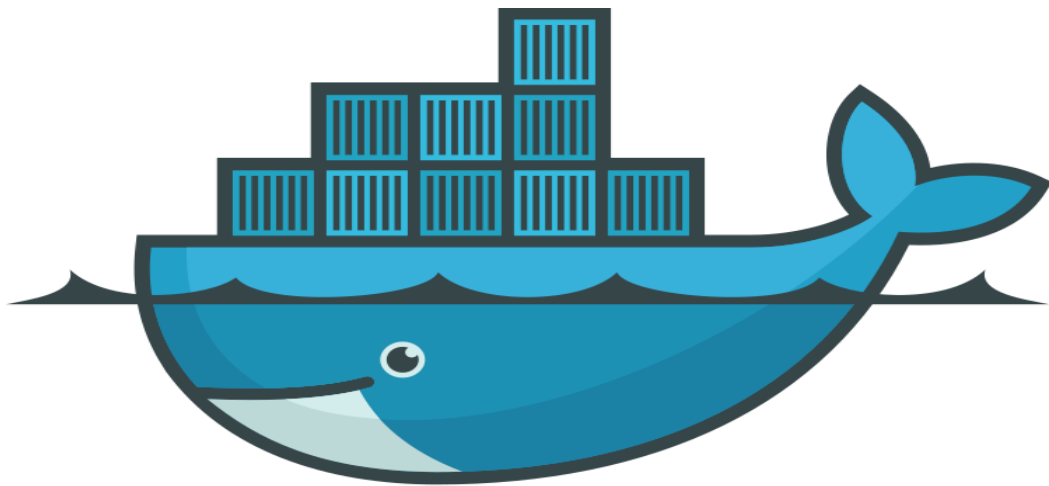


Tarea 12: Dockerizacion de Wildfly



docker

Índice

1. Introducción
2. Creación del fichero dockerfile para Wildfly
3. Uso de la nueva imagen

1. Introducción

Como hemos visto, podemos construir nuestro entorno customizado gracias a Docker.

El primer ejemplo de personalización de una imagen es a través de la imagen previa que hemos descargado de wildfly, añadiendo una pequeña personalización a este entorno.

2. Creación del fichero dockerfile para Wildfly

Creamos el directorio de trabajo:

```
usuario@ubuntu-20:~$ mkdir wildfly-config
usuario@ubuntu-20:~$ cd wildfly-config
usuario@ubuntu-20:~/wildfly-config$
```

Creamos el fichero de configuración y Añadimos la configuración de nuestra imagen:

```
usuario@ubuntu-20:~$ sudo docker run -d -p 5000:8080 --name "servidor-des" jboss/wildfly
Unable to find image 'jboss/wildfly:latest' locally
latest: Pulling from jboss/wildfly
f87ff222252e: Pull complete
13776e8da872: Pull complete
0b43aea4eeb1: Pull complete
8116b2f7ca5a: Pull complete
f26d32e28c29: Pull complete
Digest: sha256:35320abafdec6d360559b411aff466514d5741c3c527221445f48246350fdfe5
Status: Downloaded newer image for jboss/wildfly:latest
77c9bc39c41d6cdd101af898b0d215b2e93f1778184760536814e6a8ff6d3080
```

Construcción de la imagen:

```
usuario@ubuntu-20:~/wildfly-config$ sudo docker build -q --rm --tag=jboss/wildfly:25.0.0.Final-config .
[sudo] contraseña para usuario:
sha256:2a88339acee31954e73491d47747b3c608b6fb66559ca0196c09ccb12efde9a6
```

Verificando que existe la imagen dentro de docker:

```
usuario@ubuntu-20:~/wildfly-config$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
jboss/wildfly	25.0.0.Final-config	2a88339acee3	58 seconds ago	736MB
jboss/wildfly	25.0.0.Final	856694040847	4 weeks ago	736MB
jboss/wildfly	latest	856694040847	4 weeks ago	736MB
hello-world	latest	feb5d9fea6a5	5 weeks ago	13.3kB

3. Uso de la nueva imagen

Vamos a probar la imagen recién creada. Para ello ejecutamos el comando:

```
usuario@ubuntu-20:~/wildfly-config$ sudo docker run -d -p 8080:8080 -p 9990:9990 -p 8009:8009 --name servidor-wilfly-config -it jboss/wildfly:25.0.0.Final-config
```

Una vez realizada esta acción debes ejecutar el siguiente comando:

```
usuario@ubuntu-20:~/wildfly-config$ sudo docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED
c9856fac210d	jboss/wildfly:25.0.0.Final-config	"/opt/jboss/wildfly/..."	2 minutes ago
ff97c70eaf3d	jboss/wildfly:25.0.0.Final-config	"/opt/jboss/wildfly/..."	11 minutes ago

Y por último acceder a la consola de wildfly, y verificar que se puede entrar en la consola de administración con el usuario admin.

HAL Management Console

Deployments
Add and manage deployments

Deploy an application to the server

1. Use the 'Add Deployment' wizard to deploy the application
2. Enable the deployment

Configuration
Configure subsystem settings

Define a datasource to be used by deployed applications. The proper JDBC driver must be deployed and registered.

1. Select the Datasources subsystem
2. Add a Non-XA or XA datasource
3. Use the 'Create Datasource' wizard to configure the datasource settings

Runtime
Monitor server status

View runtime information such as server status, JVM status, and server log files.

1. Select the server
2. View log files or JVM usage

Access Control
Manage user and group permissions for management operations

Assign roles to users or groups to determine access to system resources.

1. Add a new user or group
2. Assign one or more roles to that user or group

Patching
Manage WildFly Full patches

Apply a WildFly Full patch to the server.

1. Download the patch file to the local machine
2. Use the 'Apply Patch' wizard to select and apply the patch

