

Instalación de servicios en Wildfly Linux



Jesús Joel Meneses Meneses
2º DAW A
DPL---Despliegue de Aplicaciones Web

Índice

1. Requisitos previos para la instalación

- ## 1.1 Instalación de Java, Maven y WildFly ##

2. Pasos para realizar la instalación de los servicios rest/ws en wildfly

- 2.1 Descarga de los ejemplos
-

. 2.2 Instalación de Servicio Hello-RS

- ## 2.3 Instalación de Servicio Hello-WS ##
-

1. Requisitos previos a la instalación

Tener un S.O. Ubuntu 20.04 con un cuenta de superusuario no root. También es necesario disponer de Java instalado y Maven.

1.1 Instalación de Java, Maven y WildFly

Prioritariamente tenemos que tener instalado Java y Maven en nuestro equipo, para asegurarnos ejecutaremos los siguientes comandos que les muestro a continuación:

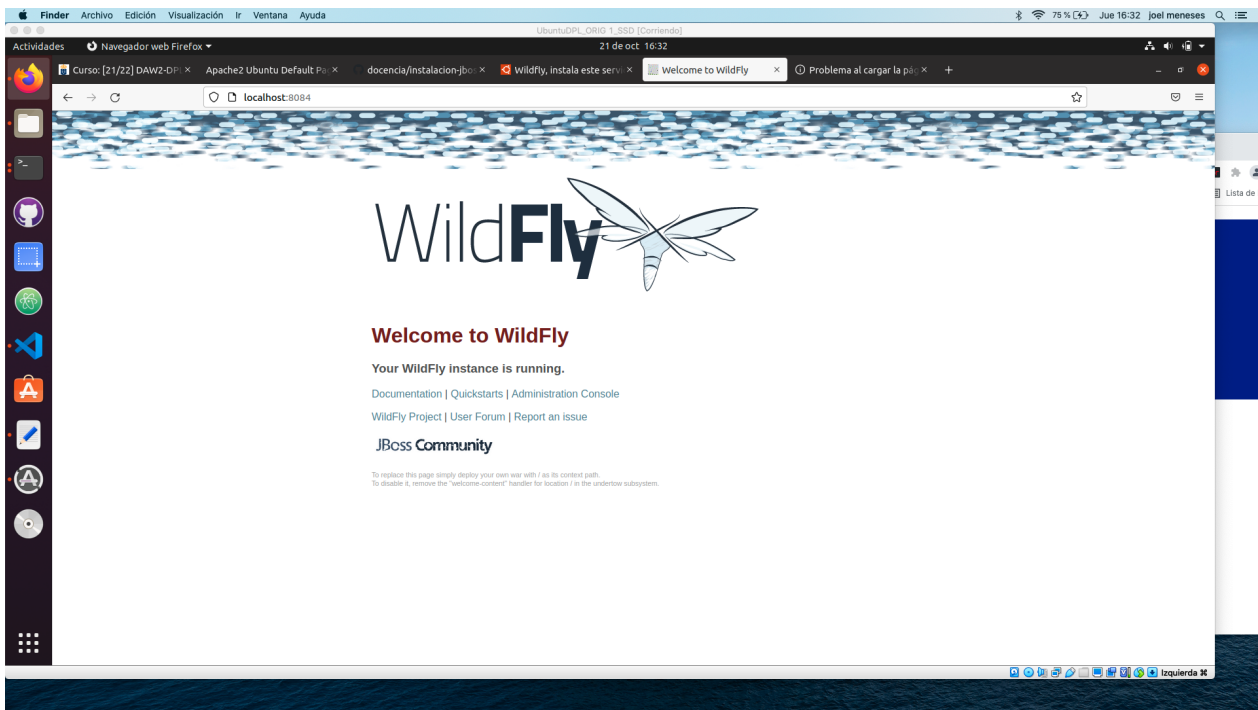
```
java --version
```

```
joel@joel-VirtualBox:~$ java --version
openjdk 11.0.11 2021-04-20
OpenJDK Runtime Environment (build 11.0.11+9-Ubuntu-0ubuntu2.20.04)
OpenJDK 64-Bit Server VM (build 11.0.11+9-Ubuntu-0ubuntu2.20.04, mixed mode, sha
ring)
joel@joel-VirtualBox:~$
```

```
mvn -version
```

```
joel@joel-VirtualBox:~$ mvn -version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.11, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd
64
Default locale: es_ES, platform encoding: UTF-8
OS name: "linux", version: "5.11.0-37-generic", arch: "amd64", family: "unix"
joel@joel-VirtualBox:~$
```

Además tener desplegado el servidor de aplicaciones WildFly, como se muestra a continuación...



2. Pasos para realizar la instalación de los servicios res/ws en WildFly

2.1 Descarga de los ejemplos

Abriremos una terminal, nos dirigimos a la carpeta tmp, y ahí, ponemos la carpeta que vamos a descargar a continuación

```
cd /tmp
```

```
joel@joel-VirtualBox:~$ cd /tmp
```

En mi git tenéis disponible la carpeta de los proyectos para descargar o clonar , en esta [url](#).

2.2 Instalación de Servicio Hello-RS

Accederemos al directorio helloworld-rs

```
joel@joel-VirtualBox:/tmp$ cd quickstart-main
joel@joel-VirtualBox:/tmp/quickstart-main$ cd helloworld-rs
```

y desde la terminal y ejecutamos el siguiente comando..

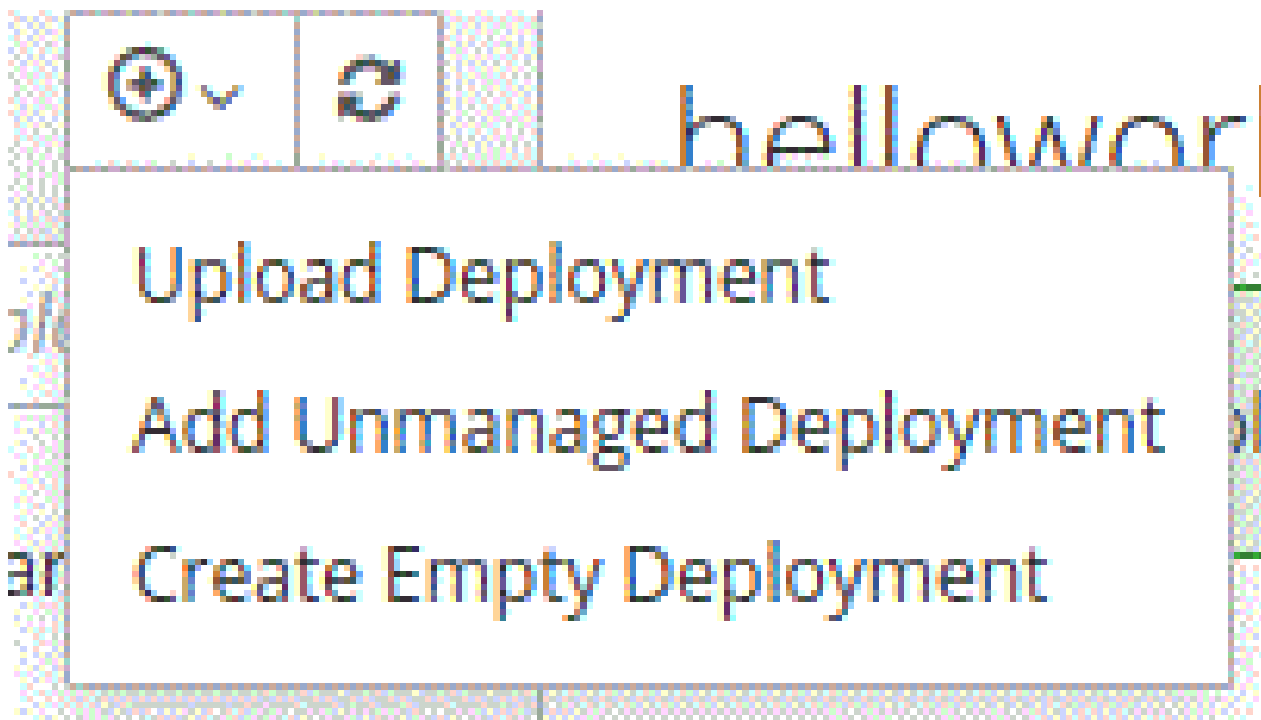
```
mvn clean install
```

```
joel@joel-VirtualBox:/tmp/quickstart-main/helloworld-rs$ mvn clean install
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
Downloading from jboss-public-maven-repository: https://repository.jboss.org/nexus/content/groups/public/org/wildfly/quickstarts/wildfly-quickstart-parent/2/wildfly-quickstart-parent-2.pom
Downloaded from jboss-public-maven-repository: https://repository.jboss.org/nexus/content/groups/public/org/wildfly/quickstarts/wildfly-quickstart-parent/2/wildfly-quickstart-parent-2.pom (18 kB at 11 kB/s)
Downloading from jboss-public-maven-repository: https://repository.jboss.org/nexus/content/groups/public/org/jboss/...
```

Finalizara con un mensaje como el siguiente

```
[INFO] Webapp assembled in [43 msecs]
[INFO] Building war: /tmp/quickstart-main/helloworld-rs/target/helloworld-rs.war
[INFO]
[INFO] --- maven-source-plugin:3.1.0:jar-no-fork (attach-sources) @ helloworld-rs ---
[INFO] Building jar: /tmp/quickstart-main/helloworld-rs/target/helloworld-rs-sources.jar
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ helloworld-rs ---
[INFO] Installing /tmp/quickstart-main/helloworld-rs/target/helloworld-rs.war to /home/joel/.m2/repository/org/wildfly/quickstarts/helloworld-rs/26.0.0.Beta1-SNAPSHOT/helloworld-rs-26.0.0.Beta1-SNAPSHOT.war
[INFO] Installing /tmp/quickstart-main/helloworld-rs/pom.xml to /home/joel/.m2/repository/org/wildfly/quickstarts/helloworld-rs/26.0.0.Beta1-SNAPSHOT/helloworld-rs-26.0.0.Beta1-SNAPSHOT.pom
[INFO] Installing /tmp/quickstart-main/helloworld-rs/target/helloworld-rs-sources.jar to /home/joel/.m2/repository/org/wildfly/quickstarts/helloworld-rs/26.0.0.Beta1-SNAPSHOT/helloworld-rs-26.0.0.Beta1-SNAPSHOT-sources.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 03:49 min
[INFO] Finished at: 2021-10-25T22:25:07+01:00
[INFO] -----
joel@joel-VirtualBox: /tmp/quickstart-main/helloworld-rs$
```

Finalizada la compilación del proyecto acudiremos a wildfly, al apartado **deployments**, vamos a la pestaña + y luego elegimos la primera opción



Seleccionamos la carpeta donde se encuentra nuestro proyecto. En concreto la ruta, donde se encuentra el archivo helloworld-rs.war

Add Deployment

Upload Deployment

Specify Names

1

2

helloworld-rs.war

Posteriormente pulsamos en el botón siguiente

En la siguiente ventana, dejamos las opciones como están y continuamos

Upload Deployment

1

Specify Names

2

[Help](#)

Name *

Runtime Name

Enabled

☒ ON

Required fields are marked with *

Finalizará con un mensaje similar al siguiente

Add Deployment


X

Upload Deployment

Specify Names

1

2



Upload successful

helloworld-rs.war has been successfully uploaded to the content repository.

View Deployment

Cancel

< Back

Close

Una vez desplegado el Servicio, abrimos el navegador y en la url ponemos nuestra **ip:puerto/helloworld-rs**, y verificamos que se muestra como en las ventanas siguientes.

← → ↻

localhost:8084/helloworld-rs/

☆

🔒 🔽 ☰

Choose between *JSON* or *XML*:

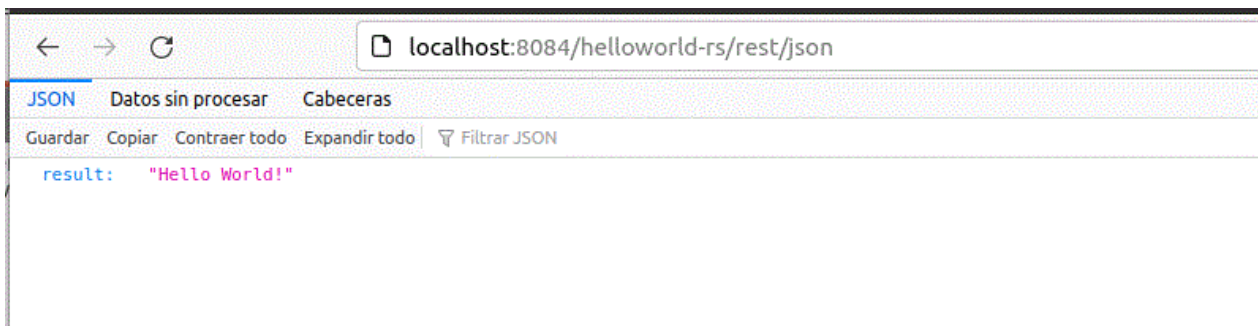
- [JSON](#)
- [XML](#)

← → ↻

localhost:8084/helloworld-rs/rest/xml

Este fichero XML no parece tener ninguna información de estilo asociada. Se muestra debajo e

```
-<xml>  
  <result>Hello World!</result>  
</xml>
```



2.3 Instalación de Servicio Hello-WS

Ahora vamos a instalar el siguiente servicio, que será como en los pasos anteriores

Accederemos al directorio hello-ws, para posteriormente ejecutar el siguiente comando

```
mvn clean install
```

```
joel@joel-VirtualBox:/tmp/quickstart-main/helloworld-ws$ mvn clean install
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/
lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.Protection
Domain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.wildfly.quickstarts:helloworld-ws >-----
[INFO] Building Quickstart: helloworld-ws 26.0.0.Beta1-SNAPSHOT
[INFO] -----[ war ]-----
Downloading from jboss-public-maven-repository: https://repository.jboss.org/nexus/content/groups/public/junit/junit
4.13.1/junit-4.13.1.pom
Downloading from redhat-ga-maven-repository: https://maven.repository.redhat.com/ga/junit/junit/4.13.1/junit-4.13.1
```

Hasta finalizar con un mensaje similar al siguiente

```
[INFO] --- maven-source-plugin:2.4.1-jdk8:attach-sources (attach-sources) @ helloworld-ws ---
[INFO] Building jar: /tmp/quickstart-main/helloworld-ws/target/helloworld-ws-sources.jar
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ helloworld-ws ---
[INFO] Installing /tmp/quickstart-main/helloworld-ws/target/helloworld-ws.war to /home/joel/.m2/repository/org/wild
fly/quickstarts/helloworld-ws/26.0.0.Beta1-SNAPSHOT/helloworld-ws-26.0.0.Beta1-SNAPSHOT.war
[INFO] Installing /tmp/quickstart-main/helloworld-ws/pom.xml to /home/joel/.m2/repository/org/wildfly/quickstarts/h
elloworld-ws/26.0.0.Beta1-SNAPSHOT/helloworld-ws-26.0.0.Beta1-SNAPSHOT.pom
[INFO] Installing /tmp/quickstart-main/helloworld-ws/target/helloworld-ws-sources.jar to /home/joel/.m2/repository/
org/wildfly/quickstarts/helloworld-ws/26.0.0.Beta1-SNAPSHOT/helloworld-ws-26.0.0.Beta1-SNAPSHOT-sources.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 26.846 s
[INFO] Finished at: 2021-10-25T22:45:58+01:00
[INFO]
joel@joel-VirtualBox:/tmp/quickstart-main/helloworld-ws$
```

Hacemos los mismos pasos para desplegar el servicio que en el paso anterior

Una vez desplegado el servicio, lo verificaremos en nuestro servidor , introduciendo en la url lo siguiente **ip:puerto/helloworld-ws**

Nos mostrará un mensaje similar al siguiente



helloworld-ws Quickstart

The *helloworld-ws* quickstart demonstrates the use of **JAX-WS** in Red Hat JBoss Enterprise Application Platform as a simple Hello World application.

There is no user interface for this quickstart. Instead, you can verify the Web Service is running and deployed correctly by accessing the following URL:

[HelloWorldService?wsdl](#)

This URL will display the WSDL definition for the deployed Web Service endpoint.

Enlace github: [enlace](#)