

Ciclo Desarrollo de Aplicaciones Web a Distancia

Curso 2024/25

Despliegue de aplicaciones Web

Unidad 3

Configuración y administración de servidores de
aplicaciones

Gabriel Cubillos Rodríguez

2 de diciembre de 2024

Contenido

| | |
|---|----------|
| Enunciado..... | 2 |
| 1. Estructura de directorios de Servlet 2.2..... | 3 |
| 2. Archivo build.xml de ant..... | 4 |
| 3. Wildfly..... | 5 |
| 3.1.- Instalación del JDK..... | 5 |
| 3.2.- Instalación de Wildfly..... | 6 |
| 3.3.- Iniciar el servicio SystemD..... | 8 |
| 3.4.- Autenticación de WildFly..... | 10 |

Índice de figuras

| | |
|--|----|
| Figura 1: Instalación del JDK 8..... | 5 |
| Figura 2: Versiones del JDK instaladas en el sistema..... | 6 |
| Figura 3: Creación del grupo y usuario wildfly..... | 6 |
| Figura 4: Descarga y descompresión del archivo de WildFly..... | 7 |
| Figura 5: Creación del enlace simbólico que apunta a la versión que está en uso..... | 7 |
| Figura 6: Puesta en marcha del servicio systemD..... | 9 |
| Figura 7: Ejecución del script para agregar un usuario en WildFly..... | 10 |
| Figura 8: Página principal de WildFly..... | 11 |
| Figura 9: Modificación en el archivo standalone.xml..... | 11 |
| Figura 10: Consola de gestión del usuario..... | 12 |

Enunciado

1. Una aplicación web puede ser desplegada en diferentes servidores web manteniendo su funcionalidad y sin ningún tipo de modificación en su código debido a la especificación servlet 2.2 , ¿cuál es la estructura de directorios que debe tener?
2. Ant se basa en ficheros XML, normalmente configuramos el trabajo a hacer con nuestra aplicación en un fichero llamado build.xml. Indica alguna de las etiquetas con las que podemos formar el contenido de este archivo.
3. Dispones de una máquina que cuenta con el sistema operativo Ubuntu 18.04 LTS o posterior, recientemente actualizado, en la que está el entorno de red configurado y, además, dispones de conexión a Internet y estás trabajando con la cuenta del usuario root . Indica cada uno de los pasos, y comandos implicados en ellos, para conseguir hacer lo siguiente:
 - * Instalar el JDK 8.
 - * Crear usuario para WildFly.
 - * Descargar e instalar WildFly 19.0.0 Final.
 - * Configurar systemd y el archivo wildfly.conf.
 - * Configurar la autenticación de Wildfly.

1. Estructura de directorios de Servlet 2.2

Una aplicación web puede ser desplegada en diferentes servidores web manteniendo su funcionalidad y sin ningún tipo de modificación en su código debido a la especificación servlet 2.2 , ¿cuál es la estructura de directorios que debe tener?

La estructura de directorios es la siguiente:

- **Directorio principal (raíz):** Contendrá los ficheros estáticos (HTML, imágenes, etc.) y JSPs.
- **Carpeta WEB-INF:** Contiene el fichero "web.xml" (descriptor de la aplicación), encargado de configurar la aplicación.
 - **Subcarpeta classes:** Contiene los ficheros compilados (servlets, beans).
 - **Subcarpeta lib:** Librerías adicionales.
- **Resto de carpetas para ficheros estáticos.**

2. Archivo build.xml de ant

Ant se basa en ficheros XML, normalmente configuramos el trabajo a hacer con nuestra aplicación en un fichero llamado build.xml. Indica alguna de las etiquetas con las que podemos formar el contenido de este archivo.

Etiquetas para el archivo build.xml de Ant:

- **<project>**: Esta etiqueta es el elemento raíz del archivo XML. Solo puede haber una etiqueta <project> en cada archivo build.xml, y se corresponde con la aplicación Java.
- **<target>**: Un target u objetivo es un conjunto de tareas que se quieren aplicar a la aplicación en algún momento. Se puede hacer que unos objetivos dependan de otros, lo que Ant gestionará automáticamente.
- **<task>**: Una task o tarea es un código ejecutable que se aplica a la aplicación. Las tareas pueden contener distintas propiedades, como por ejemplo el classpath. Ant incluye muchas tareas básicas, como la compilación o la eliminación de archivos temporales, pero se puede extender este mecanismo si es necesario.
- **<property>**: Una property o propiedad es un parámetro en forma de par nombre-valor que se necesita para procesar la aplicación. Algunos ejemplos son el nombre del compilador, etc. Ant ya incluye las propiedades más básicas, como BaseDir para el directorio base del proyecto, ant.file para la ruta absoluta del archivo build.xml, y ant.java.version para la versión de la JVM.

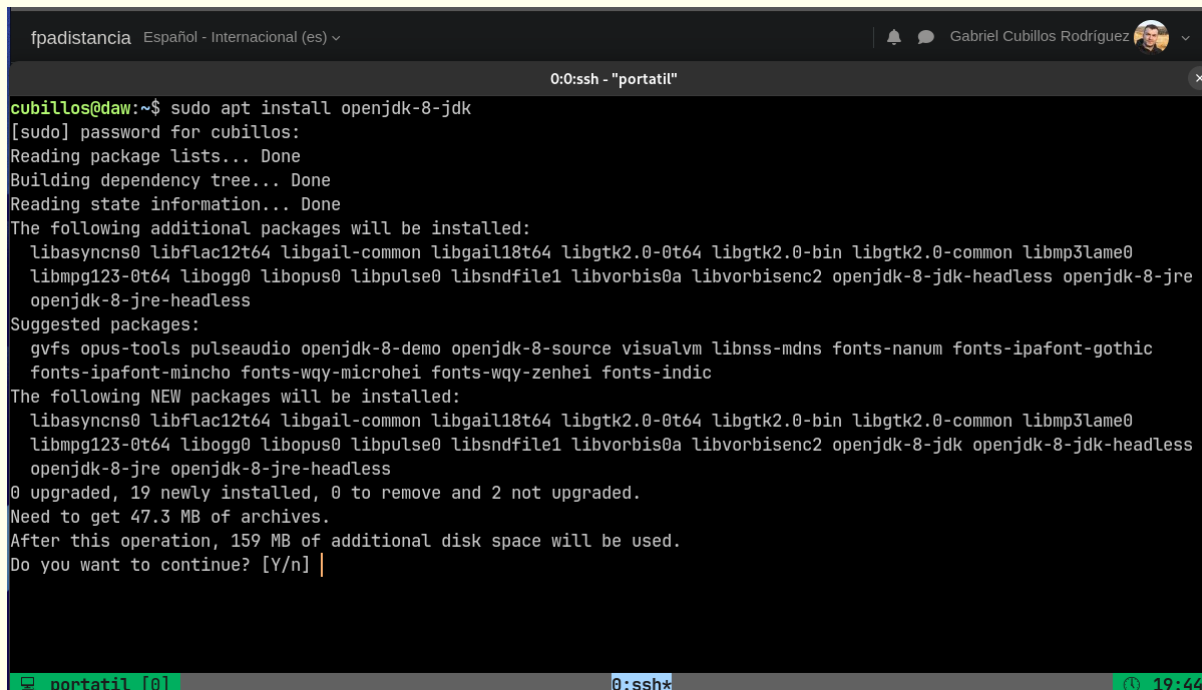
3. Wildfly

Dispones de una máquina que cuenta con el sistema operativo Ubuntu 18.04 LTS o posterior, recientemente actualizado, en la que está el entorno de red configurado y, además, dispones de conexión a Internet y estás trabajando con la cuenta del usuario `root`. Indica cada uno de los pasos, y comandos implicados en ellos, para conseguir hacer lo siguiente:

- * Instalar el JDK 8.
- * Crear usuario para WildFly.
- * Descargar e instalar WildFly 19.0.0 Final.
- * Configurar `systemd` y el archivo `wildfly.conf`.
- * Configurar la autenticación de Wildfly.

3.1.- Instalación del JDK

Accedo al servidor de Ubuntu a través de SSH e instalo la versión 8 del JDK.



```

fpadistancia Español - Internacional (es) v
0:0:ssh - "portatil"

cubillos@daw:~$ sudo apt install openjdk-8-jdk
[sudo] password for cubillos:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libasyns0 libflac12t64 libgail-common libgail18t64 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libmp3lame0
  libmpg123-0t64 libogg0 libopus0 libpulse0 libsndfile1 libvorbis0a libvorbisenc2 openjdk-8-jdk-headless openjdk-8-jre
  openjdk-8-jre-headless
Suggested packages:
  gvfs opus-tools pulseaudio openjdk-8-demo openjdk-8-source visualvm libnss-mdns fonts-nanum fonts-ipafont-gothic
  fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  libasyns0 libflac12t64 libgail-common libgail18t64 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libmp3lame0
  libmpg123-0t64 libogg0 libopus0 libpulse0 libsndfile1 libvorbis0a libvorbisenc2 openjdk-8-jdk openjdk-8-jdk-headless
  openjdk-8-jre openjdk-8-jre-headless
0 upgraded, 19 newly installed, 0 to remove and 2 not upgraded.
Need to get 47.3 MB of archives.
After this operation, 159 MB of additional disk space will be used.
Do you want to continue? [Y/n]
  
```

Figura 1: Instalación del JDK 8

Ejecuto: `sudo apt install openjdk-8-jdk`

3.2.- Instalación de Wildfly

Comprobamos que se ha instalado y que aparece entre las versiones del JDK del sistema.

```

fpadistancia Español - Internacional (es)
0:0:ssh - "portatil"

cubillos@daw:~$ update-alternatives --config java
There are 2 choices for the alternative java (providing /usr/bin/java).

  Selection    Path                                            Priority  Status
  -----
*  0            /usr/lib/jvm/java-21-openjdk-amd64/bin/java    2111     auto mode
   1            /usr/lib/jvm/java-21-openjdk-amd64/bin/java    2111     manual mode
   2            /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java  1081     manual mode

Press <enter> to keep the current choice[*], or type selection number:

```

Figura 2: Versiones del JDK instaladas en el sistema

Creo el usuario y el grupo para WildFly.

```

$ sudo groupadd -r wildfly
$ sudo useradd -r -g wildfly -d /opt/wildfly -s /sbin/nologin wildfly

```

Los modificadores indican que se cree un grupo y usuario de sistema, con la ubicación del directorio \$HOME y sin shell.

```

fpadistancia Español - Internacional (es)
0:0:ssh - "portatil"

cubillos@daw:~$ sudo groupadd -r wildfly
[sudo] password for cubillos:
cubillos@daw:~$ sudo useradd -r -g wildfly -d /opt/wildfly -s /sbin/nologin wildfly
cubillos@daw:~$ id wildfly
uid=999(wildfly) gid=988(wildfly) groups=988(wildfly)
cubillos@daw:~$

```

Figura 3: Creación del grupo y usuario wildfly

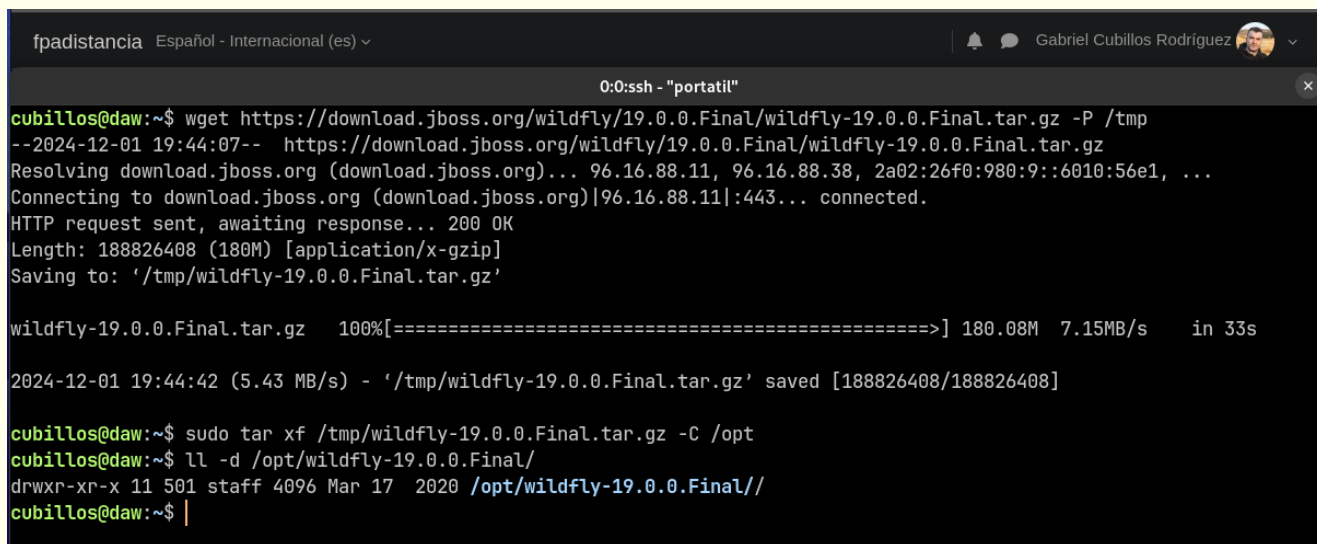
Descargo el programa Wildfly y lo descomprimo en el directorio /opt.

```

$ wget https://download.jboss.org/wildfly/19.0.0.Final/
wildfly-19.0.0.Final.tar.gz -P /tmp
$ sudo tar xf /tmp/wildfly-19.0.0.Final.tar.gz -C /opt

```

Comprobamos que se ha descomprimido correctamente:



```
fpadistancia Español - Internacional (es) v 0:0:ssh - "portatil"
cubillos@daw:~$ wget https://download.jboss.org/wildfly/19.0.0.Final/wildfly-19.0.0.Final.tar.gz -P /tmp
--2024-12-01 19:44:07-- https://download.jboss.org/wildfly/19.0.0.Final/wildfly-19.0.0.Final.tar.gz
Resolving download.jboss.org (download.jboss.org)... 96.16.88.11, 96.16.88.38, 2a02:26f0:980:9::6010:56e1, ...
Connecting to download.jboss.org (download.jboss.org)[96.16.88.11]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 188826408 (180M) [application/x-gzip]
Saving to: '/tmp/wildfly-19.0.0.Final.tar.gz'

wildfly-19.0.0.Final.tar.gz 100%[=====] 180.08M 7.15MB/s in 33s

2024-12-01 19:44:42 (5.43 MB/s) - '/tmp/wildfly-19.0.0.Final.tar.gz' saved [188826408/188826408]

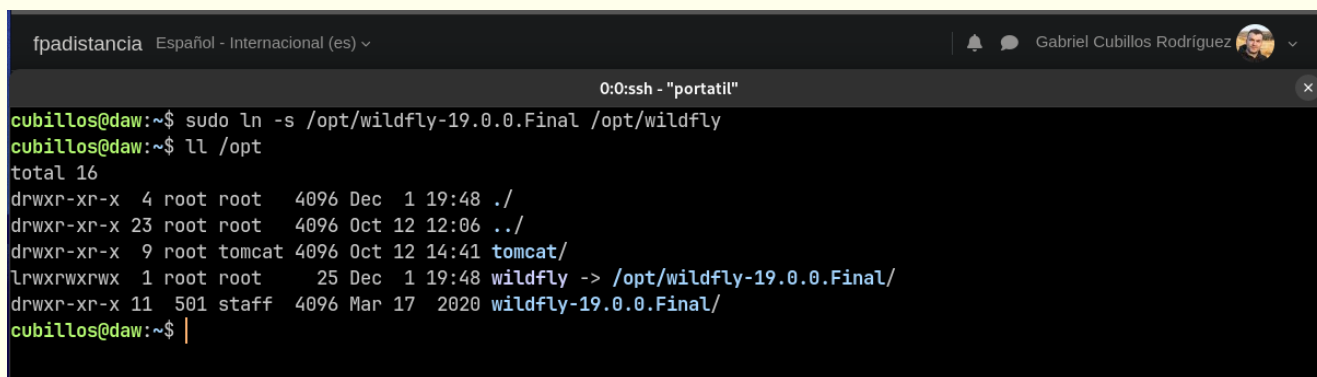
cubillos@daw:~$ sudo tar xf /tmp/wildfly-19.0.0.Final.tar.gz -C /opt
cubillos@daw:~$ ll -d /opt/wildfly-19.0.0.Final/
drwxr-xr-x 11 501 staff 4096 Mar 17 2020 /opt/wildfly-19.0.0.Final/
cubillos@daw:~$
```

Figura 4: Descarga y descompresión del archivo de WildFly

A continuación, creo un enlace simbólico, una carpeta con el nombre genérico del software que apunta a la version instalada:

```
$ sudo ln -s /opt/wildfly-19.0.0.Final /opt/wildfly
```

Observamos el resultado:



```
fpadistancia Español - Internacional (es) v 0:0:ssh - "portatil"
cubillos@daw:~$ sudo ln -s /opt/wildfly-19.0.0.Final /opt/wildfly
cubillos@daw:~$ ll /opt
total 16
drwxr-xr-x 4 root root 4096 Dec 1 19:48 ./
drwxr-xr-x 23 root root 4096 Oct 12 12:06 ../
drwxr-xr-x 9 root tomcat 4096 Oct 12 14:41 tomcat/
lrwxrwxrwx 1 root root 25 Dec 1 19:48 wildfly -> /opt/wildfly-19.0.0.Final/
drwxr-xr-x 11 501 staff 4096 Mar 17 2020 wildfly-19.0.0.Final/
cubillos@daw:~$
```

Figura 5: Creación del enlace simbólico que apunta a la versión que está en uso

Finalmente, se cambia el propietario de la carpeta /opt/wildfly:

```
$ sudo chown -RH wildfly: /opt/wildfly
```


3.3.- Iniciar el servicio SystemD

En primer lugar, creamos el directorio de configuración.

```
sudo mkdir -p /etc/wildfly
```

Copiamos en este directorio los archivos wildfly.conf y launch.sh.

```
$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.conf /  
etc/wildfly/  
$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/launch.sh /opt/  
wildfly/bin/
```

Damos permisos de ejecución a los scripts en el home de WildFly:

```
$ chmod +x /opt/wildfly/bin/*.sh
```

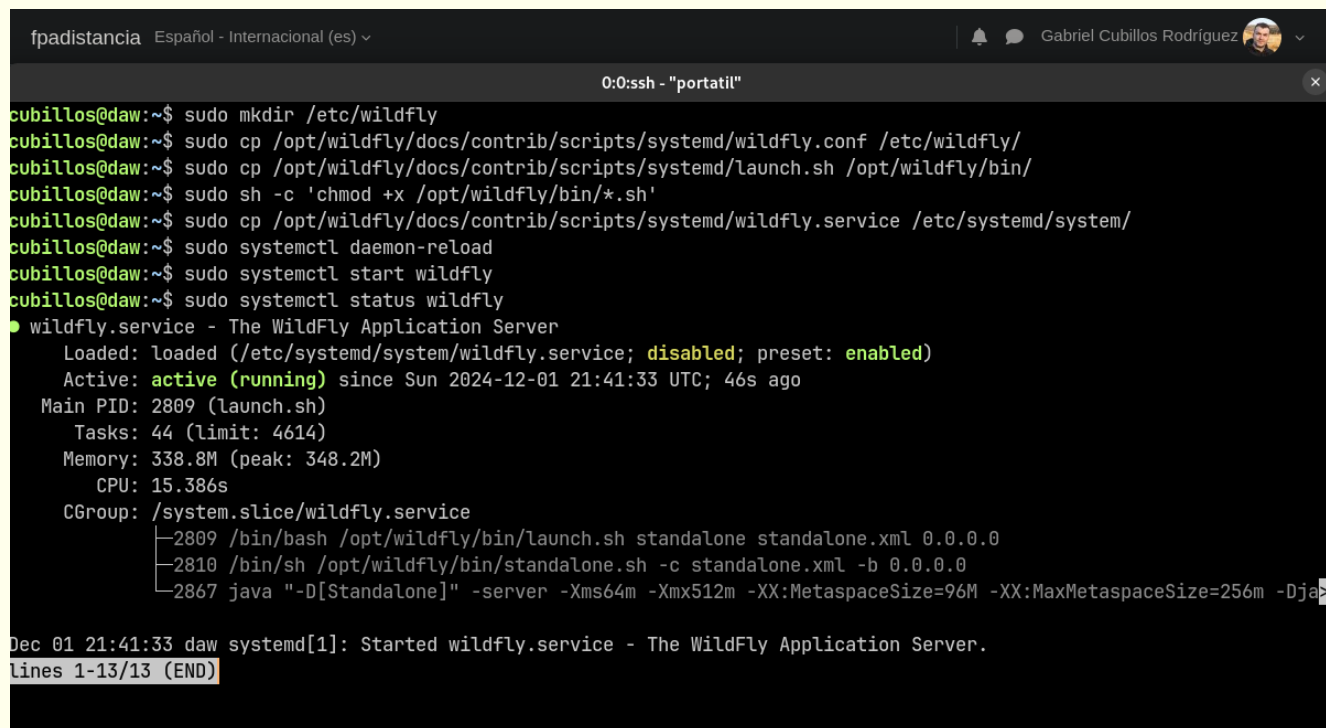
Copiamos el archivo para SystemD en el directorio correspondiente.

```
$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.service /  
etc/systemd/system/
```

Recargamos systemd, iniciamos el servicio y comprobamos que todo ha ido correctamente.

```
$ sudo systemctl daemon-reload  
$ sudo systemctl start wildfly  
$ sudo systemctl status wildfly
```

Así se ve en la terminal el resultado de las operaciones:

A terminal window titled "0:0:ssh - 'portatil'" showing a series of commands to configure and start the WildFly service. The user 'cubillos@daw' runs several 'sudo' commands: 'mkdir /etc/wildfly', 'cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.conf /etc/wildfly/', 'cp /opt/wildfly/docs/contrib/scripts/systemd/launch.sh /opt/wildfly/bin/', 'sh -c 'chmod +x /opt/wildfly/bin/*.sh'', 'cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.service /etc/systemd/system/', 'systemctl daemon-reload', 'systemctl start wildfly', and 'systemctl status wildfly'. The status output shows the service is 'active (running)' since Sun 2024-12-01 21:41:33 UTC. Below the status, a list of processes is shown under the CGroup: /system.slice/wildfly.service, including a bash process (PID 2809), a sh process (PID 2810), and a java process (PID 2867). At the bottom, a system log entry states: 'Dec 01 21:41:33 daw systemd[1]: Started wildfly.service - The WildFly Application Server.' The terminal also shows 'lines 1-13/13 (END)' at the bottom.

```
fpadistancia Español - Internacional (es) v 0:0:ssh - "portatil"
cubillos@daw:~$ sudo mkdir /etc/wildfly
cubillos@daw:~$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.conf /etc/wildfly/
cubillos@daw:~$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/launch.sh /opt/wildfly/bin/
cubillos@daw:~$ sudo sh -c 'chmod +x /opt/wildfly/bin/*.sh'
cubillos@daw:~$ sudo cp /opt/wildfly/docs/contrib/scripts/systemd/wildfly.service /etc/systemd/system/
cubillos@daw:~$ sudo systemctl daemon-reload
cubillos@daw:~$ sudo systemctl start wildfly
cubillos@daw:~$ sudo systemctl status wildfly
● wildfly.service - The WildFly Application Server
   Loaded: loaded (/etc/systemd/system/wildfly.service; disabled; preset: enabled)
   Active: active (running) since Sun 2024-12-01 21:41:33 UTC; 46s ago
     Main PID: 2809 (launch.sh)
        Tasks: 44 (limit: 4614)
      Memory: 338.8M (peak: 348.2M)
         CPU: 15.386s
      CGroup: /system.slice/wildfly.service
              └─2809 /bin/bash /opt/wildfly/bin/launch.sh standalone standalone.xml 0.0.0.0
                 └─2810 /bin/sh /opt/wildfly/bin/standalone.sh -c standalone.xml -b 0.0.0.0
                    └─2867 java "-D[Standalone]" -server -Xms64m -Xmx512m -XX:MetaspaceSize=96M -XX:MaxMetaspaceSize=256m -Dja>
Dec 01 21:41:33 daw systemd[1]: Started wildfly.service - The WildFly Application Server.
lines 1-13/13 (END)
```

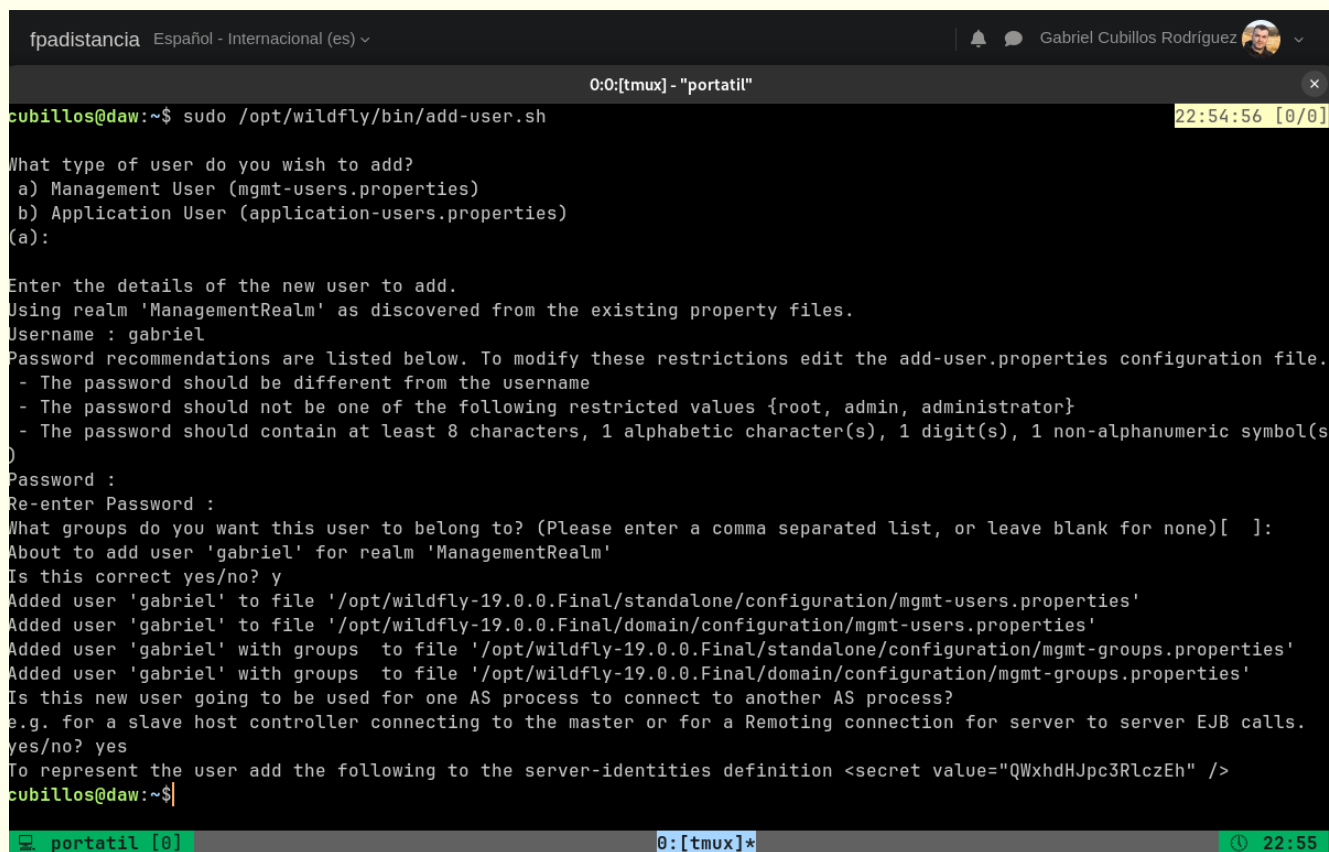
Figura 6: Puesta en marcha del servicio `systemD`

3.4.- Autenticación de WildFly

Para configurar la autenticación, primero crearemos un usuario ejecutando el script correspondiente:

```
$ sudo /opt/wildfly/bin/add-user.sh
```

Agregamos el usuario gabriel, introduciendo los datos que nos solicita el script.



```
fpadistancia Español - Internacional (es) v Gabriel Cubillos Rodríguez
0:0:[tmux] - "portatil"
cubillos@daw:~$ sudo /opt/wildfly/bin/add-user.sh 22:54:56 [0/0]

What type of user do you wish to add?
a) Management User (mgmt-users.properties)
b) Application User (application-users.properties)
(a):

Enter the details of the new user to add.
Using realm 'ManagementRealm' as discovered from the existing property files.
Username : gabriel
Password recommendations are listed below. To modify these restrictions edit the add-user.properties configuration file.
- The password should be different from the username
- The password should not be one of the following restricted values {root, admin, administrator}
- The password should contain at least 8 characters, 1 alphabetic character(s), 1 digit(s), 1 non-alphanumeric symbol(s)
Password :
Re-enter Password :
What groups do you want this user to belong to? (Please enter a comma separated list, or leave blank for none)[ ]:
About to add user 'gabriel' for realm 'ManagementRealm'
Is this correct yes/no? y
Added user 'gabriel' to file '/opt/wildfly-19.0.0.Final/standalone/configuration/mgmt-users.properties'
Added user 'gabriel' to file '/opt/wildfly-19.0.0.Final/domain/configuration/mgmt-users.properties'
Added user 'gabriel' with groups to file '/opt/wildfly-19.0.0.Final/standalone/configuration/mgmt-groups.properties'
Added user 'gabriel' with groups to file '/opt/wildfly-19.0.0.Final/domain/configuration/mgmt-groups.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.
yes/no? yes
To represent the user add the following to the server-identities definition <secret value="QWxhdHJpc3Rlc2Eh" />
cubillos@daw:~$
```

Figura 7: Ejecución del script para agregar un usuario en WildFly

Reiniciamos el servicio para asegurarnos de que se cargan la modificación en la configuración.

```
$ sudo systemctl restart wildfly
```

Después, comprobamos en un navegador que el servidor está funcionando correctamente.



Figura 8: Página principal de WildFly

Para poder acceder a la consola en la interfaz web, debemos modificar el archivo `/opt/wildfly/standalone/configuration/standalone.xml` como se ve en la siguiente captura: cambiar la IP local `127.0.0.1` por la IP `0.0.0.0`, para que acepte la conexión desde el exterior. Luego, debemos reiniciar el servicio. De otra manera, sólo se podrá acceder en local, y siendo que lo lógico es disponer el servicio en un servidor sin entorno gráfico, es necesario realizar esta modificación (y no tiene sentido que no lo muestren en los contenidos del tema, por cierto).

```
<interface name="management">
  <inet-address value="${jboss.bind.address.management:0.0.0.0}"/>
</interface>
```

Figura 9: Modificación en el archivo `standalone.xml`

Finalmente, comprobamos que se puede acceder a la consola del usuario.

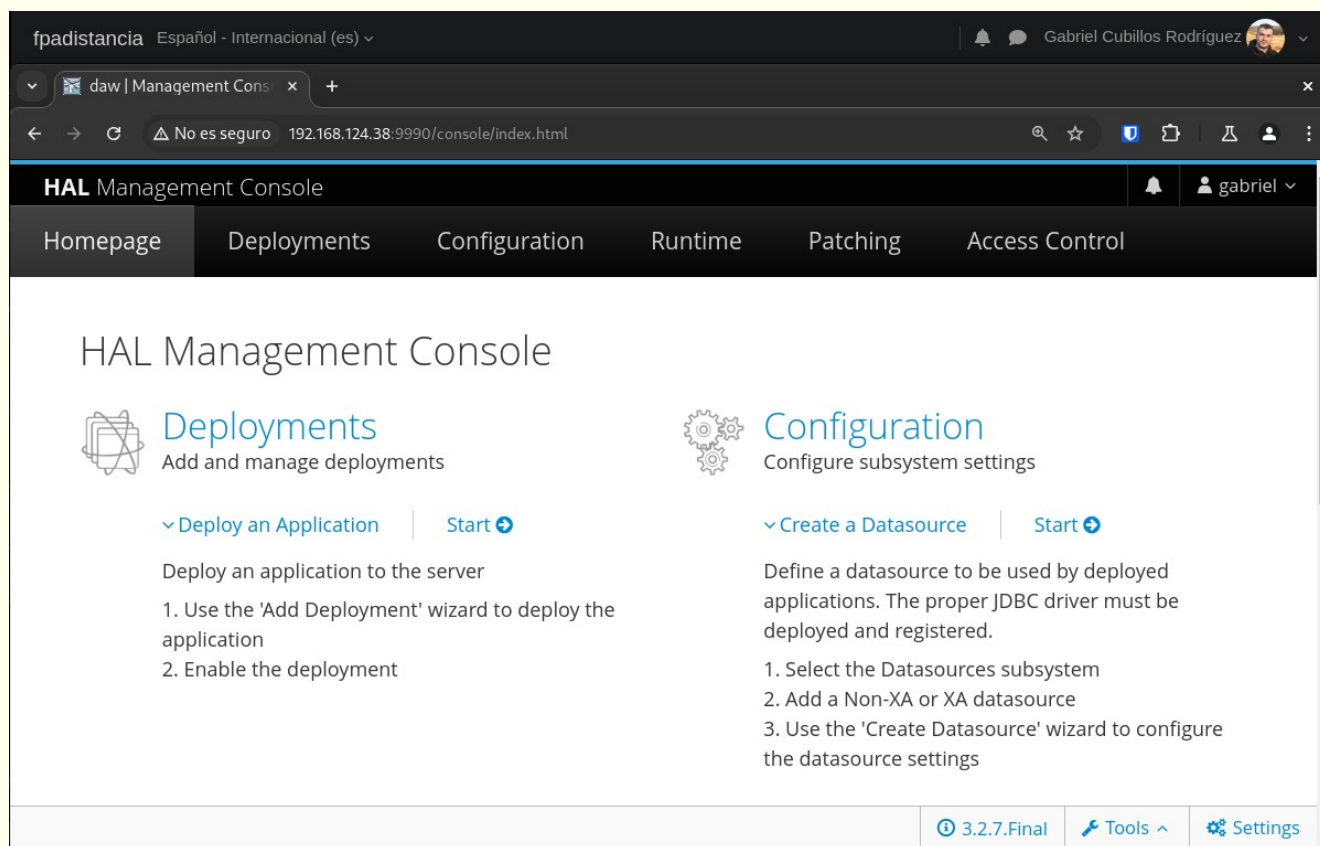


Figura 10: Consola de gestión del usuario