

Alumno/a: José Manuel González Peña

Práctica 9: Instalación Apache Tomcat 8

Objetivos:

Instalación de un servidor Apache con soporte para jsp y servlets mediante el uso de Tomcat. Tomcat es un contenedor de servlets para aplicaciones Java.

Usaremos Tomcat versión 8.0 . Entre los principales cambios en esta versión, puede resaltar el soporte para Servlet 3.0 y JavaServer Pages 2.2.

En esta práctica comenzaremos por instalar Tomcat y comprobar su correcto funcionamiento. En futuras prácticas lo integraremos con el servidor web Apache.

Una vez instalado usaremos la herramienta de gestión de Tomcat (manager)

Fuentes:

-[Tutorial instalación Tomcat en Linux](#)

- Descarga de la última versión de Java SE: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

- Otras versiones de Java: <http://www.oracle.com/technetwork/java/javase/archive-139210.html>

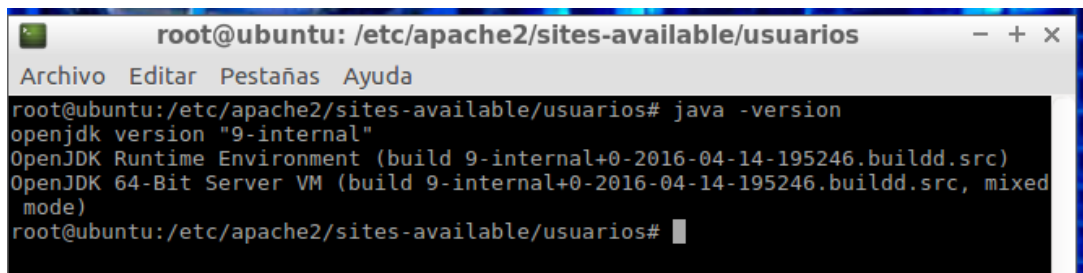
- Ayuda instalación de Java SE: http://www.java.com/es/download/help/index_installing.xml

Requerimientos:

Tener instalado Java .

1. Comprobarlo.

\$ java -version



```
root@ubuntu: /etc/apache2/sites-available/usuarios
Archivo Editar Pestañas Ayuda
root@ubuntu:/etc/apache2/sites-available/usuarios# java -version
openjdk version "9-internal"
OpenJDK Runtime Environment (build 9-internal+0-2016-04-14-195246.buildd.src)
OpenJDK 64-Bit Server VM (build 9-internal+0-2016-04-14-195246.buildd.src, mixed mode)
root@ubuntu:/etc/apache2/sites-available/usuarios#
```

2. Si no lo tienes instalado, instálalo.

a) Comprueba que tienes los requisitos hardware suficientes para instalar:

http://www.java.com/es/download/help/index_installing.xml

b) Analiza la Java Standard Edition (Java SE) u OpenJDK que más se ajuste a tus necesidades:

Version OpenJDK y Java SE:

OpenJDK: es la versión libre JDK la cual es mantenida por la comunidad, esta esta basada en las especificaciones para cada una de las versiones de la JRE descritas por SUN.

Sun Java SE: mantenido por Sun Microsystems, el I kit de desarrollo es JDK

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Which Java Standard Edition package do I need?

- **JDK:** (Java Development Kit). For Java Developers. Includes a complete JRE plus tools for developing, debugging, and monitoring Java applications.
<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>
- **Server JRE:** (Server Java Runtime Environment) For deploying Java applications on servers. Includes tools for JVM monitoring and tools commonly required for server applications, but does not include browser integration (the Java plug-in), auto-update, nor an installer.
- **JRE:** (Java Runtime Environment). Covers most end-users needs. Contains everything required to run Java applications on your system. <http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html>

Nota: JRE es la versión sin herramientas de desarrollo como son compilers o debuggers y es suficiente para esta practica.

```
sudo apt-get install openjdk-X-jre  
(donde X es la versión)
```

Si tambien me interesa el kit de desarrollo:

```
sudo apt-get install default-jdk
```

Procedimiento:

1. Instalación de Tomcat

a) Lo primero que debemos hacer es descargar el paquete "**apache-tomcat-8.0.0.tar.gz**" desde el siguiente link:

```
wget http://tomcat.apache.org/download-80.cgi
```

Lee <http://apache.rediris.es/tomcat/tomcat-8/v8.5.9/README.html>

y decide qué distribución binaria (no source) te interesa instalar.

b) Mover el fichero descargado a un directorio adecuado (se recomienda /usr/local o /usr/share)

c) Ahora lo descomprimos con el siguiente comando:

```
tar xvzf apache-tomcat-8.0.0.tar.gz
```

Esto crea un directorio con los siguientes subdirectorios alla donde :

- *bin* (contiene los binarios del servidor),
- *conf* (contiene los archivos de configuración)
- *webapps* (contiene las aplicaciones desplegadas en el servidor)

d) Asigna permisos de ejecución a todos los archivos .sh que hay bajo el directorio bin. Puedes usar el comando **chmod +x *.sh**

```
root@ubuntu:/apache-tomcat-8.5.9/bin# chmod +x *.sh  
root@ubuntu:/apache-tomcat-8.5.9/bin# ls  
bootstrap.jar          daemon.sh              startup.sh  
catalina.bat           digest.bat            tomcat-juli.jar  
catalina.sh            digest.sh             tomcat-native.tar.gz  
catalina-tasks.xml     setclasspath.bat     tool-wrapper.bat  
commons-daemon.jar     setclasspath.sh      tool-wrapper.sh  
commons-daemon-native.tar.gz shutdown.bat          version.bat  
configtest.bat         shutdown.sh           version.sh  
configtest.sh          startup.bat  
root@ubuntu:/apache-tomcat-8.5.9/bin#
```

e) Finalmente puedes eliminar el `.tar.gz` file

```
root@ubuntu:/# rm apache-tomcat-8.5.9.tar.gz
root@ubuntu:/# ls
apache-tomcat-8.5.9  etc          lib          mnt          run          sys          vmlinuz
bin                  home         lib64        opt          sbin         tmp          vmlinuz.old
boot                initrd.img   lost+found   proc         snap         usr
dev                 initrd.img.old media         root         srv          var
root@ubuntu:/#
```

2. Iniciar el servidor Tomcat:

Ejecutando el script `startup.sh` de la carpeta `bin` de tomcat:

```
mc@mc-PC: $ cd /usr/share/tomcat8/bin
```

```
mc@mc-PC: $ ./startup.sh
```

Nota: comprueba que tienes permisos de ejecución. Si da error de inexistencia del directorio logs (donde se crea el fichero de log `catalina.out`), crearlo a mano bajo `tomcat8`.

a) ¿Qué información te muestra al levantar el servicio? Captura pantalla y descríbela. Ya veras que te crea las variables de entorno `$CATALINA_HOME`, `$JRE_HOME`, etc.

```
root@ubuntu:/apache-tomcat-8.5.9# cd bin
root@ubuntu:/apache-tomcat-8.5.9/bin# ./startup.sh
Using CATALINA_BASE:   /apache-tomcat-8.5.9
Using CATALINA_HOME:   /apache-tomcat-8.5.9
Using CATALINA_TMPDIR: /apache-tomcat-8.5.9/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /apache-tomcat-8.5.9/bin/bootstrap.jar:/apache-tomcat-8.5.9/bin/tomcat-juli.jar
Tomcat started.
root@ubuntu:/apache-tomcat-8.5.9/bin#
```

b) ¿Qué script se utiliza para parar el servidor?

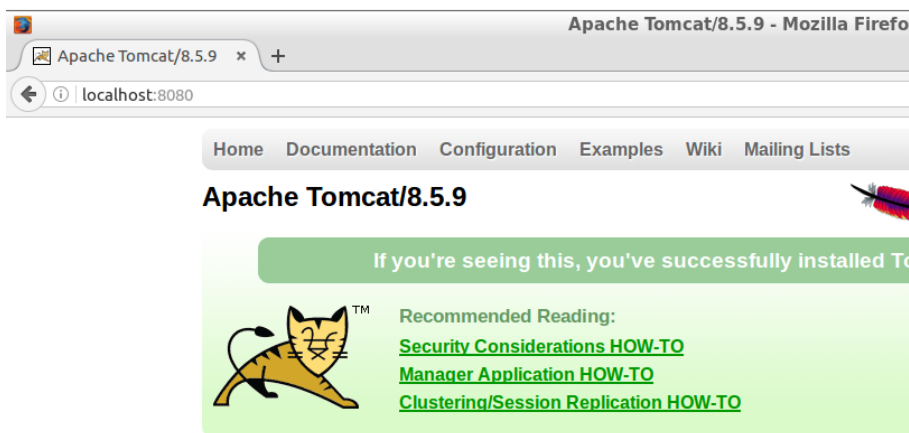
```
root@ubuntu:/apache-tomcat-8.5.9/bin# ls
bootstrap.jar      daemon.sh          startup.sh
catalina.bat       digest.bat         tomcat-juli.jar
catalina.sh        digest.sh          tomcat-native.tar.gz
catalina-tasks.xml setclasspath.bat  tool-wrapper.bat
commons-daemon.jar setclasspath.sh   tool-wrapper.sh
commons-daemon-native.tar.gz shutdown.bat       version.bat
configtest.bat     shutdown.sh        version.sh
configtest.sh      startup.bat
root@ubuntu:/apache-tomcat-8.5.9/bin#
```

c) Comprobar que funciona

Visualiza el archivo `conf/server.xml` y descubre cuál es el puerto por defecto para Tomcat.

Para comprobar que la instalación ha funcionado correctamente hay que abrir un navegador web y conectarse a la dirección <http://127.0.0.1:puertopordefecto>

```
<!-- Define a non-SSL/HTTPS HTTP/1.1 Connector on port 8080 -->
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443" />
<!-- A "Connector" using the shared thread pool -->
<!--
<Connector executor="tomcatThreadPool"
  port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  -->
```



d) Ficheros de log

Si inició el contenedor con `start` el log lo puede ver con el comando `tail` de la siguiente forma:

```
tail -f $CATALINA_HOME/logs/catalina.out
```

Nota: Si da error al crear el archivo de log `catalina.out` comprueba que el directorio de logs tiene permisos de escritura

```
root@ubuntu:/apache-tomcat-8.5.9# tail -f logs/catalina.out
11-Jan-2017 08:52:53.246 INFORMACIÓN [localhost-startStop-1] org.apache.catalina
.startup.HostConfig.deployDirectory Deployment of web application directory /apa
che-tomcat-8.5.9/webapps/manager has finished in 29 ms
11-Jan-2017 08:52:53.247 INFORMACIÓN [localhost-startStop-1] org.apache.catalina
.startup.HostConfig.deployDirectory Despliegue del directorio /apache-tomcat-8.5
.9/webapps/examples de la aplicación web
11-Jan-2017 08:52:53.550 INFORMACIÓN [main] org.apache.coyote.AbstractProtocol.s
tart Starting ProtocolHandler [ajp-nio-8009]
11-Jan-2017 08:52:53.551 INFORMACIÓN [main] org.apache.catalina.startup.Catalina
.start Server startup in 738 ms
^C
```

3. Aplicación Manager de Tomcat.

a) Para desplegar o desplegar aplicaciones es muy cómodo utilizar la aplicación manager de tomcat. Para poder acceder a dicha aplicación se necesita un usuario y un password que formen parte del rol "*manager-gui*", la configuración de dichas credenciales se encuentra en el archivo `conf/tomcat-users.xml` y su contenido debe ser el siguiente:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
<user username="user" password="changeme" roles="manager-gui"/>
</tomcat-users>
```

b) Crea un usuario que sea el administrador (llámalo `admin+[tu primer apellido]` y clave la misma).

Se puede escoger cualquier nombre de usuario; lo único importante de este archivo es que el usuario que se elija como administrador pertenezca al rol "*manager-gui*".

```
them. You will also need to set the passwords to something appropri
-->
<role rolename="manager-gui"/>
<user username="josema" password="josema1" roles="manager-gui"/>
</tomcat-users>
```

A diferencia de las versiones anteriores donde el administrador debía poseer rol "*manager*" ahora este debe ser "*manager-gui*" para operar sobre la administración web de tomcat7.

c) Entrar en la administración:

hacemos click en Tomcat Manager en el menu o directamente en la URL:

<http://127.0.0.1:8080/manager/html>

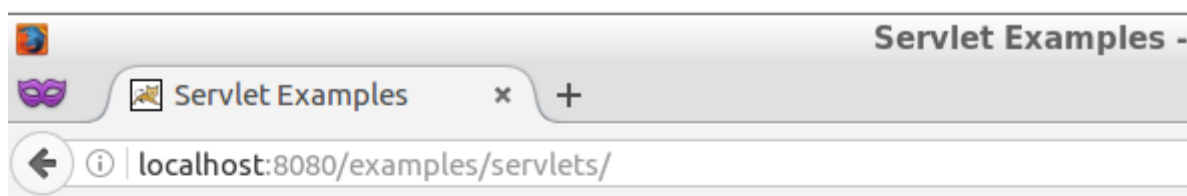
Aquí nos pedirá los datos de usuarios que anteriormente registramos en tomcat-users.xml.



Trayectoria	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado	Welcome to Tomcat	true	0	<input type="button" value="Arrancar"/> <input type="button" value="Parar"/> <input type="button" value="Expirar sesiones"/>

d) Probar "examples" para cerciorarse que todo funciona ok, estos se encuentran en la seccion "Miscellaneous" del menu lateral o en la URL:

<http://127.0.0.1:8080/examples/>



Servlet Examples with Code

This is a collection of examples which demonstrate some of the more frequently used Servlet API methods. It is assumed that you have a basic understanding of Servlets and how they are deployed.

These examples will only work when viewed via an http URL. They will not work if you view them via a file URL. They will not provide with this Tomcat release regarding how to configure and start the server.

Wherever you see a form, enter some data and see how the servlet reacts to it. This will help you to see exactly what your browser is sending the server.