



Actividad 2. Tarea Individual. Configuración del Servidor

PILAR BERMEJO 2 DAW

<https://github.com/Pilupbp/despliegueAppWeb>

Queremos preparar nuestro servidor Linux para poder desplegar una aplicación web. Para ello tenemos que verificar que están instalados:

1. Java
2. Apache
3. Tomcat
4. openSSH
5. MariaDB

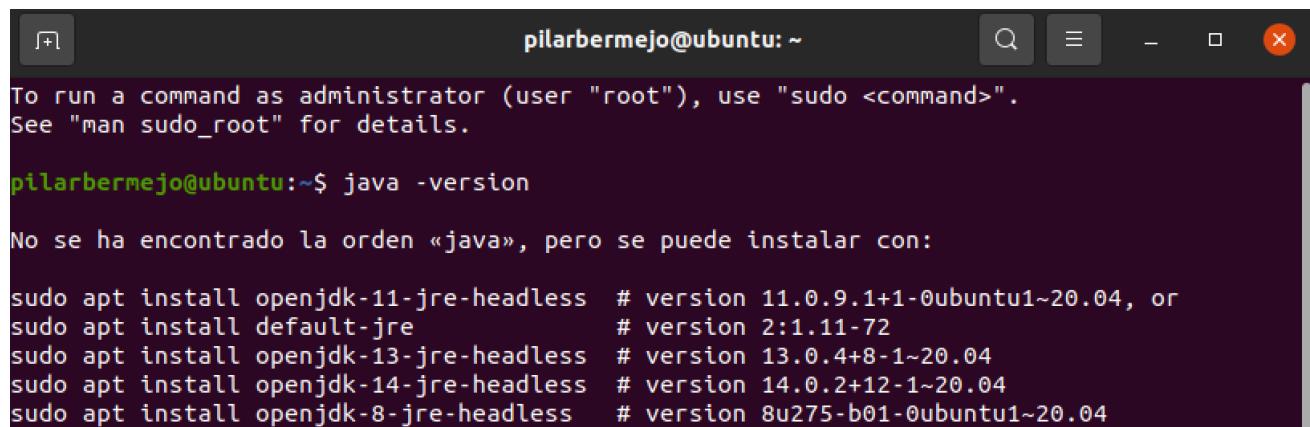
Así mismo, queremos asegurarnos de que los servidores están bien configurados y son accesibles antes de desplegar la aplicación. Por ello debemos configurar y comprobar que los puertos asociados a Apache, Tomcat y MariaDB están abiertos en el Firewall y son accesibles desde el exterior.

Para toda la actividad se valorará el orden y la claridad de la documentación, así como la facilidad de uso.
Para la entrega, es necesaria la creación y subida a la plataforma de un pequeño documento formal sobre la actividad (portada, explicación, etc.) y un Manual de instalación que describa y permita realizar todos los pasos para la instalación y configuración de las aplicaciones.
 Nótese que más adelante se pedirá que se realicen tareas con un repositorio GIT que contenga la documentación de esta actividad.

Hemos instalado VM Ware creando una máquina virtual de Ubuntu
Lo primero que hacemos es en nuestro Ubuntu abrir el terminal.

Para comprobar si tenemos **JAVA** instalado, realizamos el comando:

\$ Java -version



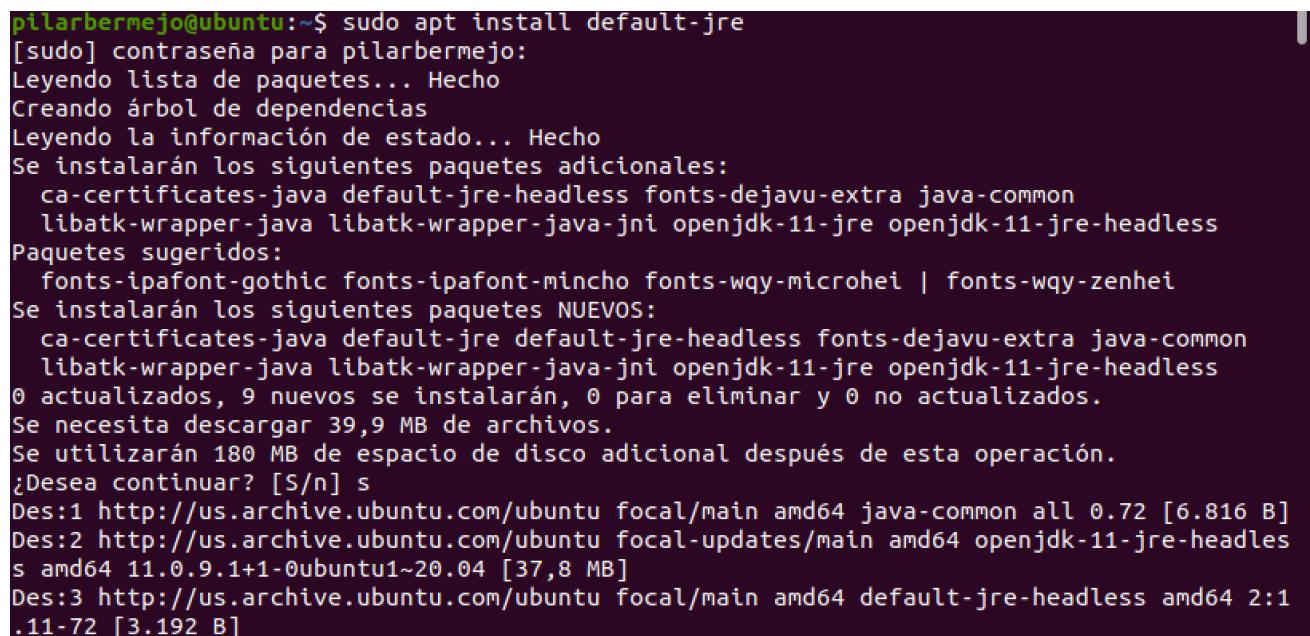
```
pilarbermejo@ubuntu:~$ java -version
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

pilarbermejo@ubuntu:~$ java -version
No se ha encontrado la orden «java», pero se puede instalar con:
sudo apt install openjdk-11-jre-headless # version 11.0.9.1+1-0ubuntu1~20.04, or
sudo apt install default-jre           # version 2:1.11-72
sudo apt install openjdk-13-jre-headless # version 13.0.4+8-1~20.04
sudo apt install openjdk-14-jre-headless # version 14.0.2+12-1~20.04
sudo apt install openjdk-8-jre-headless # version 8u275-b01-0ubuntu1~20.04
```

Como podemos comprobar, no lo tenemos instalado.

Procedemos a instalarlo con el comando que nos indica el terminal.

\$ sudo apt install default-jre

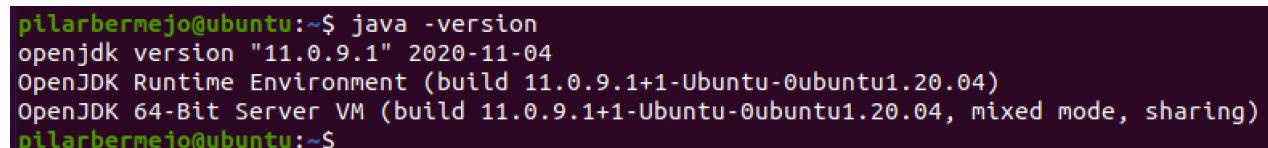


```
pilarbermejo@ubuntu:~$ sudo apt install default-jre
[sudo] contraseña para pilarbermejo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ca-certificates-java default-jre-headless fonts-dejavu-extra java-common
  libatk-wrapper-java libatk-wrapper-java-jni openjdk-11-jre openjdk-11-jre-headless
Paquetes sugeridos:
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei
Se instalarán los siguientes paquetes NUEVOS:
  ca-certificates-java default-jre default-jre-headless fonts-dejavu-extra java-common
  libatk-wrapper-java libatk-wrapper-java-jni openjdk-11-jre openjdk-11-jre-headless
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 39,9 MB de archivos.
Se utilizarán 180 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 java-common all 0.72 [6.816 B]
Des:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 openjdk-11-jre-headless amd64 11.0.9.1+1-0ubuntu1~20.04 [37,8 MB]
Des:3 http://us.archive.ubuntu.com/ubuntu focal/main amd64 default-jre-headless amd64 2:1.11-72 [3.192 B]
```

Con el que instalaremos la versión por defecto.

Una vez realizado volvemos a comprobar la instalación con el comando

\$ Java -version



```
pilarbermejo@ubuntu:~$ java -version
openjdk version "11.0.9.1" 2020-11-04
OpenJDK Runtime Environment (build 11.0.9.1+1-Ubuntu-0ubuntu1.20.04)
OpenJDK 64-Bit Server VM (build 11.0.9.1+1-Ubuntu-0ubuntu1.20.04, mixed mode, sharing)
pilarbermejo@ubuntu:~$
```

El siguiente paso es **APACHE**, este está disponible en los repositorios de software predeterminados de Ubuntu, lo que permite instalarlo con las herramientas convencionales de administración de paquetes.

Instalamos el paquete apache2 con el comando siguiente:

```
$ sudo apt install apache2
```

```
pilarbermejo@ubuntu:~$ sudo apt install apache2
[sudo] contraseña para pilarbermejo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1.819 kB de archivos.
Se utilizarán 7.935 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libapr1 amd64 1.6.5-1
ubuntu1 [91,4 kB]
Des:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1 amd64 1.6
.1-4ubuntu2 [84,7 kB]
Des:3 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1-dbd-sqlit
e3 amd64 1.6.1-4ubuntu2 [10,5 kB]
Des:4 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libaprutil1-ldap amd6
```

Antes de probar Apache, es necesario modificar los ajustes de firewall para permitir el acceso externo a los puertos web predeterminados.

Durante la instalación, Apache se registra con UFW para proporcionar algunos perfiles de aplicación que pueden utilizarse para habilitar o deshabilitar el acceso a Apache a través del firewall.
Enumeramos los perfiles de aplicación ufw escribiendo lo siguiente:

```
$ sudo ufw app list
```

```
pilarbermejo@ubuntu:~$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
```

Existen tres perfiles disponibles para Apache:

- **Apache**: este perfil abre solo el puerto 80 (tráfico web normal no cifrado)
- **Apache Full**: este perfil abre el puerto 80 (tráfico web normal no cifrado) y el puerto 443 (tráfico TLS/SSL cifrado)
- **Apache Secure**: este perfil abre solo el puerto 443 (tráfico TLS/SSL cifrado)

Vamos a habilitar el perfil más restrictivo que permitirá el tráfico en el puerto 80.

```
$ sudo ufw allow 'Apache'
```

Verificamos con **\$ sudo ufw status**

```

pilarbermejo@ubuntu:~$ sudo ufw allow 'Apache'
Reglas actualizadas
Reglas actualizadas (v6)
pilarbermejo@ubuntu:~$ sudo ufw status
Estado: inactivo
pilarbermejo@ubuntu:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
    Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese>
      Active: active (running) since Mon 2021-02-08 09:33:26 CET; 10min ago
        Docs: https://httpd.apache.org/docs/2.4/
       Main PID: 2785 (apache2)
          Tasks: 55 (limit: 4620)
         Memory: 6.2M
        CGroup: /system.slice/apache2.service
                  ├─2785 /usr/sbin/apache2 -k start
                  ├─2786 /usr/sbin/apache2 -k start
                  └─2787 /usr/sbin/apache2 -k start

feb 08 09:33:25 ubuntu systemd[1]: Starting The Apache HTTP Server...
feb 08 09:33:25 ubuntu apachectl[2784]: AH00558: apache2: Could not reliably de>
feb 08 09:33:26 ubuntu systemd[1]: Started The Apache HTTP Server.
lines 1-15/15 (END)

```

El siguiente paso es **APACHE TOMCAT** es un servidor web contenedor de Servlets que utilizamos para presentar aplicaciones Java.

Lo instalamos con el comando:

```
$ sudo apt install tomcat9
```

```

pilarbermejo@ubuntu:~$ sudo apt install tomcat9
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libeclipse-jdt-core-java libtcnative-1 libtomcat9-java tomcat9-common
Paquetes sugeridos:
  tomcat9-admin tomcat9-docs tomcat9-examples tomcat9-user
Se instalarán los siguientes paquetes NUEVOS:
  libeclipse-jdt-core-java libtcnative-1 libtomcat9-java tomcat9
  tomcat9-common
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 12,3 MB de archivos.
Se utilizarán 14,6 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libeclipse-jdt-co

```

Una vez instalado comprobamos el estado:

```
$ systemctl status tomcat9
```

```

pilarbermejo@ubuntu:~$ systemctl status tomcat9
● tomcat9.service - Apache Tomcat 9 Web Application Server
    Loaded: loaded (/lib/systemd/system/tomcat9.service; enabled; vendor prese>
      Active: active (running) since Mon 2021-02-08 10:08:27 CET; 9min ago
        Docs: https://tomcat.apache.org/tomcat-9.0-doc/index.html
       Main PID: 6584 (java)
          Tasks: 35 (limit: 4620)
         Memory: 147.4M
        CGroup: /system.slice/tomcat9.service
                  └─6584 /usr/lib/jvm/default-java/bin/java -Djava.util.logging.conf

feb 08 10:08:30 ubuntu tomcat9[6584]: OpenSSL inicializado correctamente [OpenS>
feb 08 10:08:31 ubuntu tomcat9[6584]: Inicializando el manejador de protocolo [>
feb 08 10:08:31 ubuntu tomcat9[6584]: Server initialization in [1.765] millisec>
feb 08 10:08:31 ubuntu tomcat9[6584]: Arrancando servicio [Catalina]
feb 08 10:08:31 ubuntu tomcat9[6584]: Starting Servlet engine: [Apache Tomcat/9]>
feb 08 10:08:31 ubuntu tomcat9[6584]: Desplegado el directorio [/var/lib/tomca>
feb 08 10:08:33 ubuntu tomcat9[6584]: Al menos un JAR, que se ha explorado busc>
feb 08 10:08:33 ubuntu tomcat9[6584]: Deployment of web application directory [>
feb 08 10:08:33 ubuntu tomcat9[6584]: Starting ProtocolHandler ["http-nio-8080"]>
feb 08 10:08:33 ubuntu tomcat9[6584]: Server startup in [1.710] milliseconds
lines 1-20/20 (END)

```

Accedemos al Firewall:

```
$ sudo ufw allow 8080/tcp
```

```
pilarbermejo@ubuntu:~$ sudo ufw allow 8080/tcp
Reglas actualizadas
Reglas actualizadas (v6)
```

Se podría cambiar el Puerto de escucha, editando el archivo server.xml, sustituyendo el valor de port.

```
$ sudo nano /etc/tomcat9/server.xml
```

Al entrar en la URL localhost:8080, nos sale este mensaje:

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in /etc/tomcat9/tomcat-users.xml.

Algunas aplicaciones de Tomcat 9, como las aplicaciones administrativas, requieren el acceso autenticado de usuarios con cierto nivel de privilegios o roles.

Por ejemplo, el Gestor de Aplicaciones Web requiere usuarios con rol *manager-gui*, mientras que el Gestor de Máquina Virtual requiere el rol *admin-gui*.

Podemos crear los usuarios que consideremos con contraseña y con uno o ambos roles, en este caso será un solo usuario con ambos roles, para lo que editaremos el archivo *tomcat-users.xml*:

Y lo editamos.

```
$ sudo nano /etc/tomcat9/tomcat-users.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
               xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
               version="1.0">
<!--
NOTE: By default, no user is included in the "manager-gui" role required
to operate the "/manager/html" web application. If you wish to use this app,
you must define such a user - the username and password are arbitrary. It is
strongly recommended that you do NOT use one of the users in the commented out
section below since they are intended for use with the examples web
application.
-->
<!--
NOTE: The sample user and role entries below are intended for use with the
examples web application. They are wrapped in a comment and thus are ignored
when reading this file. If you wish to configure these users for use with the
examples web application, do not forget to remove the <!-- ... --> that surrounds
them. You will also need to set the passwords to something appropriate.
-->
<!--
<role rolename="admin-gui"/>
<role rolename="manager-gui"/>
<user username="tomcat" password=<must-be-changed> roles="tomcat"/>
<user username="admin" password="admin" roles="admin-gui,manager-gui"/>
<user username="role1" password=<must-be-changed> roles="role1"/>
-->
</tomcat-users>
```

Descomentamos los roles para que nos deje acceder al Manager.

Paramos el servicio con **\$ service tomcat9 stop**

Y lo volvemos a levantar con **\$ service tomcat9 start**

Algunas aplicaciones, de nuevo entre ellas las aplicaciones administrativas, restringen en su configuración personal el acceso desde red, por lo que debemos editar su archivo de configuración *context.xml*. Son configuraciones de las aplicaciones en particular, no de Tomcat 9, y algunas aplicaciones tendrán esta característica y otras no.

En el caso del Gestor de Aplicaciones Web, o aplicación «*Manager*», editamos su archivo *context.xml*:

\$ sudo nano /usr/share/tomcat9-admin/manager/META-INF/context.xml

```
GNU nano 4.8                               /usr/share/tomcat9-admin/manager/META-INF/context.xml                         Modificado
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements. See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
<!--
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
        allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:1" />
-->
<Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|String)|org\.apache\.catalina\.filters\.SessionAttributeValueFilter" />
</Context>
```

Para el Gestor de Máquina Virtual, o aplicación «*Host Manager*», habría que hacer un cambio exactamente igual en su archivo *context.xml*, ubicado en su propia ruta de configuración.

\$ sudo nano /usr/share/tomcat9-admin/host-manager/META-INF/context.xml

```
GNU nano 4.8                               /usr/share/tomcat9-admin/host-manager/META-INF/context.xml                         Modificado
<?XML Version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements. See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
<!--
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
        allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:1" />
-->
<Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|String)|org\.apache\.catalina\.filters\.SessionAttributeValueFilter" />
</Context>
```

Y reiniciamos Tomcat:

\$ sudo systemctl restart tomcat9

A continuación usando nuestro usuario y.contraseña configurados accedemos al Manager.

The screenshot shows the Apache Tomcat Manager interface at <http://localhost:8080/manager/html>. The title bar says "Gestor de Aplicaciones Web de Tomcat". The main content area displays a table of deployed applications:

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Below the table, there is a "Desplegar" section with the sub-instruction "Desplegar directorio o archivo WAR localizado en servidor".

Mediante la sección de Deploy puedo instalar más aplicaciones.

Podemos comprobar que Java funciona a través de la descarga de este ejemplo:

<http://tomcat.apache.org/tomcat-7.0-doc/appdev/sample/>

Descargamos el archivo war y lo subimos a través del Deploy.

/sample	Ninguno especificado	Hello, World Application
---------	----------------------	--------------------------

Podemos ver que funciona Java y se puede parar y arrancar para utilizarlo o Undeploy y borrarlo.

The screenshot shows the output of a JSP page titled "Sample Application JSP Page". It features a cartoon cat icon and the text "Hello, World Application". Below the title, it says "This is the output of a JSP page that is part of the Hello, World application." and "Hello!".

/sample	Ninguno especificado	Hello, World Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
---------	----------------------	--------------------------	------	---	--

Cuando tenemos un servidor de aplicaciones tenemos usuarios que se conectan a ello y crean sesiones.

Active HttpSession informations									
Refresh Sessions list 1 active Sessions									
Session Id	Type	Guessed Locale	Guessed User name	Creation Time	Last Accessed Time	Used Time	Inactive Time	TTL	
964AFC46DAB73A8E5049813BE1446414	Primary			2021-02-08 15:30:32	2021-02-08 15:34:50	00:04:17	00:00:25	00:29:34	
Invalidate selected Sessions									

Estas sesiones nos informan de quién se ha conectado y como.

Hacemos accesible el Firewall con el comando.

```
$ sudo ufw allow 8080
```

```
pilarbermejo@ubuntu:~$ sudo ufw allow 8080
[sudo] contraseña para pilarbermejo:
Reglas actualizadas
Reglas actualizadas (v6)
```

SSH (Secure Shell) es un protocolo que nos permite acceder a una máquina remota de forma segura. **OpenSSH** es un conjunto de herramientas opensource basadas en el protocolo SSH.

Instalamos el servidor OpenSSH y sus dependencias y la aplicación de cliente.

```
$ sudo apt-get install openssh-server openssh-client
```

```
pilarbermejo@ubuntu:~$ sudo apt-get install openssh-server openssh-client
[sudo] contraseña para pilarbermejo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
openssh-client ya está en su versión más reciente (1:8.2p1-4ubuntu0.1).
fijado openssh-client como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  gir1.2-nautilus-3.0 python3-nautilus
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  ncurses-term openssh-sftp-server ssh-import-id
Paquetes sugeridos:
  molly-guard monkeysphere ssh-askpass
Se instalarán los siguientes paquetes NUEVOS:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 688 kB de archivos.
```

Reiniciamos el servicio: **\$ sudo systemctl restart sshd.service**

Y comprobamos el estado: **\$ sudo systemctl status sshd.service**

```
pilarbermejo@ubuntu:~$ sudo systemctl status sshd.service
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Mon 2021-02-08 16:03:10 CET; 20s ago
    Docs: man:sshd(8)
          man:sshd_config(5)
   Process: 3980 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 3981 (sshd)
     Tasks: 1 (limit: 4619)
    Memory: 1.3M
      CGroup: /system.slice/sshd.service
              └─3981 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

feb 08 16:03:10 ubuntu systemd[1]: Starting OpenBSD Secure Shell server...
feb 08 16:03:10 ubuntu sshd[3981]: Server listening on 0.0.0.0 port 22.
feb 08 16:03:10 ubuntu sshd[3981]: Server listening on :: port 22.
feb 08 16:03:10 ubuntu systemd[1]: Started OpenBSD Secure Shell server.
```

Averiguo la IP del cliente y del servidor (ifconfig) e intento realizar la conexión desde mi PC al servidor Ubuntu mediante el comando:

```
$ ssh pilarbermejo@192.168.150.2
```

```
[pilarbermejo@iMac-de-M ~ % ssh pilarbermejo@192.168.150.2
The authenticity of host '192.168.150.2 (192.168.150.2)' can't be established.
ECDSA key fingerprint is SHA256:JPSxolaDa8yTUe1qmwoyXo8Ew8ApK4o/r0wPoMga/yg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? ]
```

```

[status: active]
[pilarbermejo@iMac-de-M ~ % ssh pilarbermejo@192.168.150.2
The authenticity of host '192.168.150.2 (192.168.150.2)' can't be established.
ECDSA key fingerprint is SHA256:JPSxolaDa8yTUE1qmwoyXo8Ew8ApK4o/r0wPoMga/yg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.150.2' (ECDSA) to the list of known hosts.
[pilarbermejo@192.168.150.2's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.8.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

0 actualizaciones se pueden instalar inmediatamente.
0 de estas actualizaciones son una actualización de seguridad.

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

pilarbermejo@ubuntu:~$ 

```

Ya estamos dentro del otro ordenador, el prompt es el del usuario del ordenador remoto y el nombre del equipo es el remoto, ahora mediante los comandos en el terminal podemos actuar con el ordenador remotamente como si estuviéramos delante de él.

On el comando **\$ exit** salimos.

```

[pilarbermejo@ubuntu:~$ exit
cerrar sesión
Connection to 192.168.150.2 closed.
pilarbermejo@iMac-de-M ~ % 

```

MARIADB es un sistema de administración relacional de bases de datos de código abierto, que comúnmente se usa como alternativa para la parte de MySQL de la popular pila LAMP (Linux, Apache, MySQL, PHP/Python/Perl). Se diseñó como un reemplazo a medida de MySQL.

Instalamos el paquete MariaDV server.

\$ sudo apt install mariadb-server

```

pilarbermejo@ubuntu:~$ sudo apt install mariadb-server
[sudo] contraseña para pilarbermejo:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
  gir1.2-nautilus-3.0 python3-nautilus
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  galera-3 gawk libaio1 libcgi-fast-perl libcgipm-perl
  libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libfcgi-perl
  libhtml-template-perl libreadline5 libsigsegv2 libsnapy1v5
  libterm-readkey-perl mariadb-client-10.3 mariadb-client-core-10.3
  mariadb-common mariadb-server-10.3 mariadb-server-core-10.3 socat
Paquetes sugeridos:
  gawk-doc libclone-perl liblmbm-perl libnet-daemon-perl
  libsql-statement-perl libipc-sharedcache-perl mailx mariadb-test tinyca
Se instalarán los siguientes paquetes NUEVOS:
  galera-3 gawk libaio1 libcgi-fast-perl libcgipm-perl
  libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libfcgi-perl
  libhtml-template-perl libreadline5 libsigsegv2 libsnapy1v5
  libterm-readkey-perl mariadb-client-10.3 mariadb-client-core-10.3

```

Comprobamos el estado de MariaDB

\$ sudo systemctl status mariadb

```
mariadb.service - MariaDB 10.3.25 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2021-02-08 18:16:06 CET; 4min 6s ago
       Docs: man:mysqld(8)
             https://mariadb.com/kb/en/library/systemd/
     Main PID: 5627 (mysqld)
      Status: "Taking your SQL requests now..."
        Tasks: 31 (limit: 4619)
       Memory: 65.6M
      CGroup: /system.slice/mariadb.service
              └─5627 /usr/sbin/mysqld

feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: mysql
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: performance_schema
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: Phase 6/7: Checking and upgrading tables
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: Processing databases
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: information_schema
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: performance_schema
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: Phase 7/7: Running 'FLUSH PRIVILEGES'
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5665]: OK
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5724]: Checking for insecure root accounts.
feb 08 18:16:06 ubuntu /etc/mysql/debian-start[5728]: Triggering myisam-recover for all MyISAM tables and aria-recover for all Aria tables
```

Como comprobación adicional, puede intentamos establecer conexión con la base de datos usando la herramienta mysqladmin.

\$ sudo mysqladmin versión

```
pilarbermejo@ubuntu:~$ sudo mysqladmin version
mysqladmin Ver 9.1 Distrib 10.3.25-MariaDB, for debian-linux-gnu on x86_64
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Server version          10.3.25-MariaDB-0ubuntu0.20.04.1
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket              /var/run/mysqld/mysqld.sock
Uptime:                  6 min 2 sec

Threads: 7  Questions: 458  Slow queries: 0  Opens: 177  Flush tables: 1  Open tables: 31  Queries per second avg: 1.265
pilarbermejo@ubuntu:~$
```

Comprobamos que los puertos están escuchando:

\$ sudo lsof -i -P -n

```
pilarbermejo@ubuntu:~$ sudo lsof -i -P -n
COMMAND   PID   USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd-r  673 systemd-resolve  12u  IPv4  31998      0t0  UDP 127.0.0.53:53
systemd-r  673 systemd-resolve  13u  IPv4  31999      0t0  TCP 127.0.0.53:53 (LISTEN)
avahi-dae  717      avahi  12u  IPv4  34716      0t0  UDP *:5353
avahi-dae  717      avahi  13u  IPv6  34717      0t0  UDP *:5353
avahi-dae  717      avahi  14u  IPv4  34718      0t0  UDP *:47969
avahi-dae  717      avahi  15u  IPv6  34719      0t0  UDP *:48516
cupsd     720      root   6u  IPv6  35354      0t0  TCP [::1]:631 (LISTEN)
cupsd     720      root   7u  IPv4  35355      0t0  TCP 127.0.0.1:631 (LISTEN)
NetworkMa 722      root   23u  IPv4  77682      0t0  UDP 192.168.150.2:68->192.168.150.1:67
cups-brow 812      root   7u  IPv4  35494      0t0  UDP *:631
java      840      tomcat 39u  IPv6  41451      0t0  TCP *:8080 (LISTEN)
apache2   871      root   4u  IPv6  38035      0t0  TCP *:80 (LISTEN)
apache2   872      www-data 4u  IPv6  38035      0t0  TCP *:80 (LISTEN)
apache2   873      www-data 4u  IPv6  38035      0t0  TCP *:80 (LISTEN)
firefox   2088  pilarbermejo 125u  IPv4  76788      0t0  TCP 192.168.150.2:41998->35.165.138.131:443 (ESTABLISHED)
sshd     4782      root   3u  IPv4  74612      0t0  TCP *:22 (LISTEN)
sshd     4782      root   4u  IPv6  74614      0t0  TCP *:22 (LISTEN)
mysqld   5627      mysql  21u  IPv4  81152      0t0  TCP 127.0.0.1:3306 (LISTEN)
```

También podemos utilizar el comando netstat:

\$ sudo netstat -plnut

```
pilarbermejo@ubuntu:~$ sudo netstat -plnut
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local           Dirección remota     Estado      PID/Program name
tcp      0      0 127.0.0.53:53            0.0.0.0:*          ESCUCHAR    673/systemd-resolve
tcp      0      0 0.0.0.0:22             0.0.0.0:*          ESCUCHAR    4782/sshd: /usr/sbin
tcp      0      0 127.0.0.1:631            0.0.0.0:*          ESCUCHAR    720/cupsd
tcp      0      0 127.0.0.1:3306            0.0.0.0:*          ESCUCHAR    5627/mysqld
tcp6     0      0 ::1:22                  ::*:              ESCUCHAR    4782/sshd: /usr/sbin
tcp6     0      0 ::1:631                 ::*:              ESCUCHAR    720/cupsd
tcp6     0      0 ::8080                 ::*:              ESCUCHAR    840/java
tcp6     0      0 ::80                  ::*:              ESCUCHAR    871/apache2
udp      0      0 0.0.0.0:5353            0.0.0.0:*          ESCUCHAR    717/avahi-daemon: r
udp      0      0 127.0.0.53:53            0.0.0.0:*          ESCUCHAR    673/systemd-resolve
udp      0      0 0.0.0.0:631            0.0.0.0:*          ESCUCHAR    812/cups-browsed
udp      0      0 0.0.0.0:47969           0.0.0.0:*          ESCUCHAR    717/avahi-daemon: r
udp6     0      0 ::5353                 ::*:              ESCUCHAR    717/avahi-daemon: r
udp6     0      0 ::48516                ::*:              ESCUCHAR    717/avahi-daemon: r
```

Vamos a comprobar la configuración del **FIREWALL**:

\$ sudo ufw app list

Vemos los perfiles y verificamos el estado.

\$ sudo ufw status

```
pilarbermejo@ubuntu:~$ sudo ufw app list
[sudo] contraseña para pilarbermejo:
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
  OpenSSH
pilarbermejo@ubuntu:~$ sudo ufw status
Estado: activo
Hasta          Acción     Desde
----          ----
Apache          ALLOW     Anywhere
8080/tcp        ALLOW     Anywhere
8080           ALLOW     Anywhere
Apache (v6)      ALLOW     Anywhere (v6)
8080/tcp (v6)   ALLOW     Anywhere (v6)
8080 (v6)       ALLOW     Anywhere (v6)
```

Con UFW podemos abrir y cerrar puertos según necesitemos. Los puertos son interfaces de conexión utilizadas por las aplicaciones para establecer una conexión a un servidor.

Con el comando **\$ sudo netstat -tulpn** podemos ver todos los puertos. Comprobamos si los servicios de la red, los deamons y los programas pertinentes están activos y en estado de escucha al puerto correcto.

```
pilarbermejo@ubuntu:~$ sudo netstat -tulpn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local           Dirección remota     Estado      Inodo usuario      PID/Program name
tcp      0      0 127.0.0.53:53            0.0.0.0:*          ESCUCHAR    101      32062      737/systemd-resolve
tcp      0      0 0.0.0.0:22             0.0.0.0:*          ESCUCHAR    0         38209      952/sshd: /usr/sbin
tcp      0      0 127.0.0.1:631            0.0.0.0:*          ESCUCHAR    0         35883      886/cupsd
tcp      0      0 127.0.0.1:3306            0.0.0.0:*          ESCUCHAR    127      40837      1021/mysqld
tcp6     0      0 ::1:22                  ::*:              ESCUCHAR    998      42578      957/java
tcp6     0      0 ::1:80                  ::*:              ESCUCHAR    0         38762      1099/apache2
tcp6     0      0 ::1:22                  ::*:              ESCUCHAR    0         38211      952/sshd: /usr/sbin
tcp6     0      0 ::1:631                 ::*:              ESCUCHAR    0         35882      886/cupsd
udp      0      0 0.0.0.0:631            0.0.0.0.*          ESCUCHAR    0         36042      897/cups-browsed
udp      0      0 0.0.0.0:5353            0.0.0.0.*          115      35619      784/avahi-daemon: r
udp      0      0 0.0.0.0:57267           0.0.0.0.*          115      35621      784/avahi-daemon: r
udp      0      0 127.0.0.53:53            0.0.0.0.*          101      32061      737/systemd-resolve
udp6     0      0 ::59758                ::*:              ESCUCHAR    115      35622      784/avahi-daemon: r
udp6     0      0 ::5353                 ::*:              ESCUCHAR    115      35620      784/avahi-daemon: r
```

Añadimos los puertos 22 y 3306 de MariaDB y lo mostramos en el Firewall:

```
pilarbermejo@ubuntu:~$ sudo ufw allow 22
Regla añadida
Regla añadida (v6)
pilarbermejo@ubuntu:~$ sudo ufw allow 3306
Regla añadida
Regla añadida (v6)
```

Los puertos por los que escucha MariaDB 3306, Tomcat 8080 y Apache 80 y el seguro 443 los hemos habilitado en el FIREWALL.

```
pilarbermejo@ubuntu:~$ sudo ufw allow 443
Regla añadida
Regla añadida (v6)
pilarbermejo@ubuntu:~$ sudo ufw status
Estado: activo

Hasta          Acción     Desde
----          -----     -----
Apache          ALLOW      Anywhere
8080/tcp        ALLOW      Anywhere
8080           ALLOW      Anywhere
22             ALLOW      Anywhere
3306           ALLOW      Anywhere
443            ALLOW      Anywhere
Apache (v6)    ALLOW      Anywhere (v6)
8080/tcp (v6)  ALLOW      Anywhere (v6)
8080 (v6)      ALLOW      Anywhere (v6)
22 (v6)        ALLOW      Anywhere (v6)
3306 (v6)      ALLOW      Anywhere (v6)
443 (v6)       ALLOW      Anywhere (v6)
```

Comprobamos de nuevo que los servicios Apache, Tomcat y MariaDB están corriendo.

```
pilarbermejo@ubuntu:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; v
  Active: active (running) since Tue 2021-02-09 11:05:10 CET; 43m
    Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 1099 (apache2)
    Tasks: 55 (limit: 4619)
   Memory: 8.5M
  CGroup: /system.slice/apache2.service
          └─1099 /usr/sbin/apache2 -k start
              ├─1100 /usr/sbin/apache2 -k start
              ├─1101 /usr/sbin/apache2 -k start
```

```
pilarbermejo@ubuntu:~$ sudo systemctl status tomcat9
● tomcat9.service - Apache Tomcat 9 Web Application Server
  Loaded: loaded (/lib/systemd/system/tomcat9.service; enabled; v
  Active: active (running) since Tue 2021-02-09 11:04:51 CET; 43m
    Docs: https://tomcat.apache.org/tomcat-9.0-doc/index.html
  Main PID: 957 (java)
    Tasks: 36 (limit: 4619)
   Memory: 210.3M
  CGroup: /system.slice/tomcat9.service
          └─957 /usr/lib/jvm/default-java/bin/java -Djava.util.lo
```

```
pilarbermejo@ubuntu:~$ sudo systemctl status mariadb
● mariadb.service - MariaDB 10.3.25 database server
  Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: en
  Active: active (running) since Tue 2021-02-09 11:05:25 CET; 43min ago
    Docs: man:mysqld(8)
          https://mariadb.com/kb/en/library/systemd/
  Main PID: 1021 (mysqld)
    Status: "Taking your SQL requests now..."
    Tasks: 30 (limit: 4619)
   Memory: 94.2M
  CGroup: /system.slice/mariadb.service
          └─1021 /usr/sbin/mysqld
```