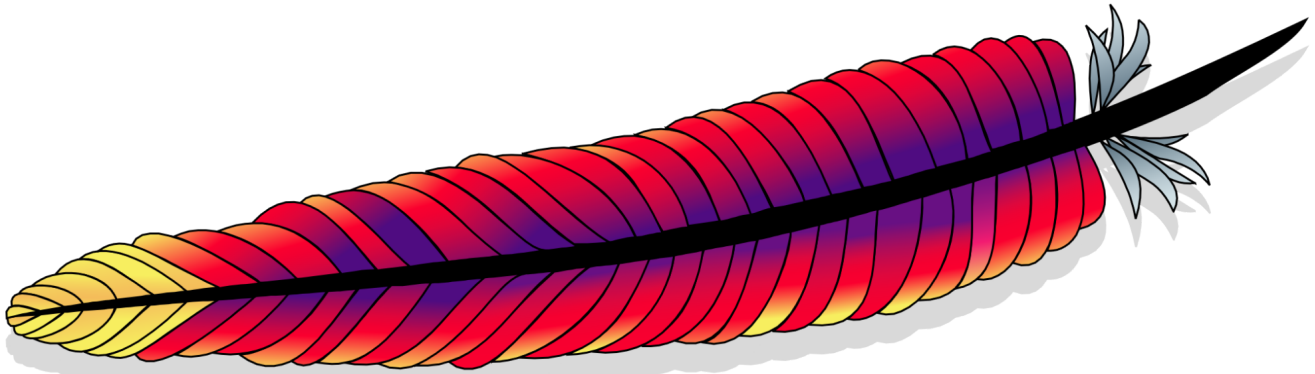


## Tarea 1.1- Instalación de servidor web Apache

---



# Apache Web Server

## Contenidos

---

- Capas de la arquitectura web: ¿cuáles son? y funciones
- Plataformas web: LAMP y WISA
- Instalación de servidor web APACHE
  - 1. Actualizar repositorio e instalación desde la terminal
  - 2. Comprobamos la instalación desde la terminal
  - 3. Comprobamos la instalación desde navegador
  - 4. Cambio del puerto de escucha de Apache
- Instalación de Tomcat
  - 1. Instalación de Java
  - 2. Creación de usuario Tomcat
  - 3. Descarga y extracción del paquete Tomcat
  - 4. Configuración del Tomcat
  - 5. Comprobación de Tomcat

## Capas de la arquitectura web: ¿cuáles son? y funciones

La arquitectura en capas es un modelo de diseño de software, cuya base es la separación de las diferentes funcionalidades del sistema en capas o niveles, donde cada capa se encarga de un conjunto de tareas específicas y se comunica con los niveles adyacentes mediante interfaces bien definidas.

En una web se pueden identificar **tres capas principales: la capa de presentación, la capa de lógica de negocio y la capa de acceso a datos**. La primera se encarga de mostrar la información al usuario; la segunda, de procesa la información y la tercera, de almacenar y recuperar la información.

## Plataformas web: LAMP y WISA

**LAMP** es un sistema de infraestructura web con usa las siguientes herramientas: **Linux** como sistema operativo; **Apache** como servidor web; **MySQL/MariaDB** como gestor de bases de datos y **PHP** como lenguaje de programación.

Por otro lado, **WISA** usa como componentes: **Windows** (sistema operativo), **Internet Information Services** (servidor web), **SQL Server** (manejador de bases de datos) y **ASP** (que hace de intermediario para trabajar con un lenguaje de programación, **Visual Basic**).

## Instalación de servidor web APACHE

Para realizar la instalación de nuestro servidor web APACHE, tenemos que realizar las siguientes acciones:

### 1. Actualizar repositorio e instalación desde la terminal

```
# Actualizamos el repositorio de paquetes del sistema
sudo apt-get update

# Instalamos el paquete de Apache
sudo apt-get install apache2
```

### 2. Comprobamos la instalación desde la terminal

Como la actividad de Apache se observa como un servicio del sistema, podemos comprobar su instalación y funcionamiento con:

```
systemctl status apache2
```

```
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Fri 2024-11-01 17:27:58 WET; 3min 8s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3778 (apache2)
    Tasks: 55 (limit: 9436)
   Memory: 5.7M (peak: 6.4M)
      CPU: 86ms
   CGroup: /system.slice/apache2.service
           └─3778 /usr/sbin/apache2 -k start
             └─3779 /usr/sbin/apache2 -k start
               └─3780 /usr/sbin/apache2 -k start
```

### 3. Comprobamos la instalación desde navegador

También podemos comprobar su funcionamiento a través del navegador, accediendo a [localhost](http://localhost).



#### 4. Cambio del puerto de escucha de Apache

Por defecto, APACHE viene configurado para ajustarse en el puerto 80. Para esta actividad, cambiaremos dicha configuración para que su puerto de escucha sea el 82. Para ello, primero accederemos con privilegios de *root* al archivo de configuración **ports.conf** con el siguiente comando:

```
sudo nano etc/apache2/ports.conf
```

En este fichero encontraremos toda la información y configuración relacionada con los puertos. Para cambiar el puerto de escucha, basta con reemplazar el valor de **Listen 80** a **Listen 82**.

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```

Se cambia a "Listen 82"

Y comprobamos si se realizó el cambio correctamente accediendo a **localhost:82**.



[!NOTE] Es posible que al intentar acceder a **localhost:82** no funcione, esto es porque al realizar cambios en la configuración de un servicio es posible que se requiera reiniciar el mismo. En dicho caso, bastaría con ejecutar en la consola `systemctl restart apache2` para aplicar los cambios de puertos y que funcione correctamente.

## Instalación de Tomcat

A grandes rasgos, **Tomcat** se presenta como un servidor web que implementa las tecnologías de servlets y JSP (JavaServer Pages), brindando así un entorno robusto para la ejecución de aplicaciones Java en la web. Dotado de una arquitectura flexible y escalable, Tomcat se erige como una opción privilegiada para albergar sitios web dinámicos y sofisticados.

### 1. Instalación de Java

Por ello, es un servicio que tiene como prerequisite la posesión de Java en el sistema operativo. Para suplir esta necesidad, instalaremos Java con:

```
sudo apt-get install openjdk-11-jdk
```

Y posteriormente comprobamos su instalación con:

```
java --version
```

### 2. Creación de usuario Tomcat

No es una buena práctica correr el servicio de Tomcat con privilegios de `root`, para ello, crearemos un usuario de bajo nivel para eso.

```
# Creamos un grupo para alojar los usuarios de tomcat
sudo groupadd tomcat

# Creamos un directorio para la ubicación de la configuración
sudo mkdir /opt/tomcat

# Creamos un nuevo usuario con el grupo previamente creado
sudo useradd -g tomcat -d /opt/tomcat -s /usr/sbin/nologin tomcat
```

### 3. Descarga y extracción del paquete Tomcat

Descargamos la última versión del paquete Tomcat:

```
curl https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.tar.gz -o apache-tomcat-9.0.96.tar.gz
```

Y la extraemos y movemos a nuestro directorio `/opt/tomcat`:

```
sudo tar -zxvf apache-tomcat-*.tar.gz
sudo mv apache-tomcat-*/ /opt/tomcat/
```

Y ahora cambiamos la propiedad al usuario Tomcat para que le sea permitido escribir en los ficheros

```
sudo chown -R tomcat:tomcat /opt/tomcat/
```

### 4. Configuración del Tomcat

Tomcat requiere distintos pasos para iniciar con su funcionamiento. Primeramente, necesitamos saber la dirección en donde se encuentra Java instalado en nuestro sistema, para ello ejecutamos el siguiente comando que nos devuelve la ruta:

```
sudo update-java-alternatives -l

# OUTPUT: java-1.11.0-openjdk-amd64 1111 /usr/lib/jvm/java-1.11.0-openjdk-amd64
```

Crearemos un fichero `systemd` para Tomcat

```
sudo nano /etc/systemd/system/tomcat.service
```

Y agregamos la siguiente información:

```
[Unit]
Description=Apache Tomcat Web Application Container
Wants=network.target
After=network.target
[Service]
Type=forking
Environment=JAVA_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment='CATALINA_OPTS=-Xms512M -Xmx1G -Djava.net.preferIPv4Stack=true'
Environment='JAVA_OPTS=-Djava.awt.headless=true'
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
SuccessExitStatus=143
User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
```

Y recargamos *systemd daemon* :

```
sudo systemctl daemon-reload
```

## 5. Comprobación de Tomcat

Finalmente, podemos comprobar si se puede iniciar el servidor y su estatus con:

```
sudo systemctl start tomcat
systemctl status tomcat
```

```
● tomcat.service - Apache Tomcat Web Application Container
   Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; preset: enabled)
   Active: active (running) since Fri 2024-11-01 18:19:06 WET; 3s ago
     Process: 7390 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
    Main PID: 7397 (java)
      Tasks: 37 (limit: 9436)
     Memory: 99.3M (peak: 100.1M)
        CPU: 4.835s
    CGroup: /system.slice/tomcat.service
```