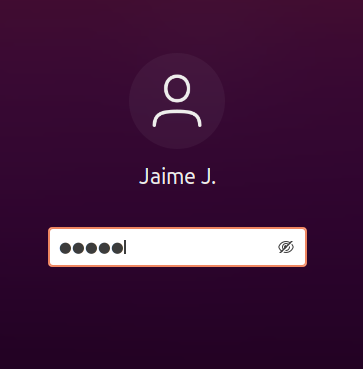
**Actividad 2. Configuración de servidor**

En la siguiente guía se a desarrollar el proceso de creación, instalación y configuración de un servidor en la máquina virtual con ubuntu.

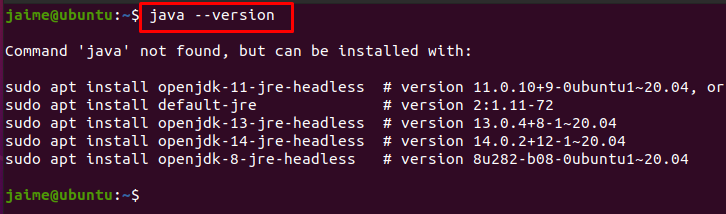
Se hará una explicación de cada comando utilizado y una captura para ir viendo las salidas que da la terminal con cada comando.

Para comenzar se arrancará la maquina virtual ya instalada previamente con el usuario que se creó en la instalación.



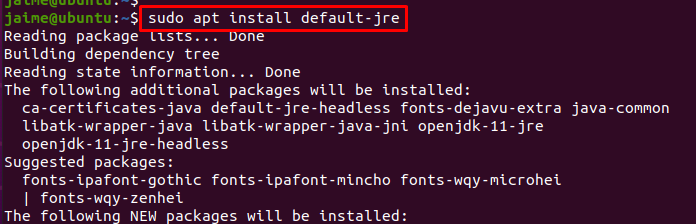
1.- **Verificación e instalación de java**

Se comprueba si está instalado java para ello se utiliza el comando **java –versión**

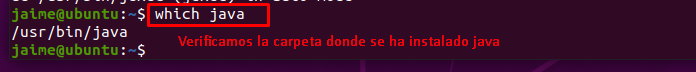


Como se puede apreciar no está instalado java. Para ello se utilizarán los comandos proporcionados por la propia terminal de Ubuntu.

Con el comando sudo apt install default-jre se instala el jre por defecto.

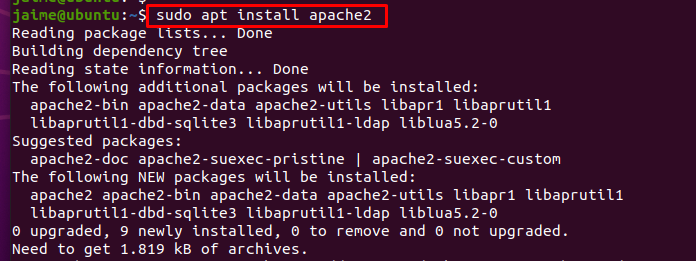


Con el comando which java verificamos donde se ha instalado

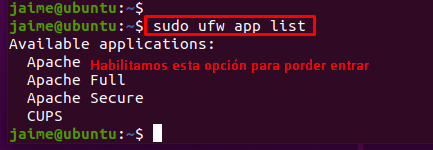


**2.- Verificación e instalación de Apache.**

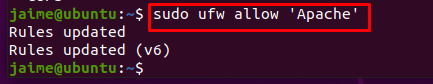
Para iniciar la instalación del servidor apache, primero actualizo todos los paquetes y luego instalo con sudo apt install apache2.



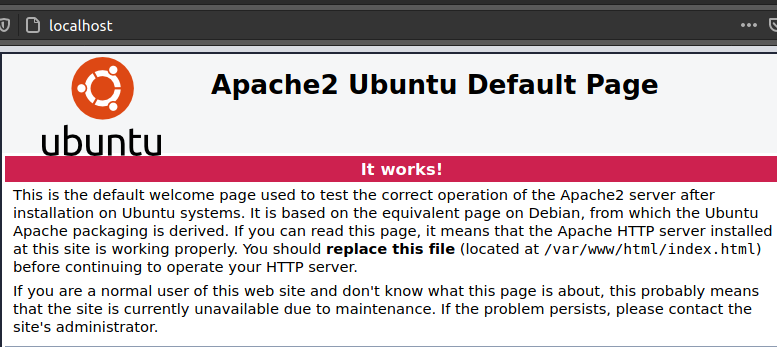
Una vez instalado vemos la lista de los perfiles disponibles con el comando sudo ufw app list, Solo habilitaremos el Apache con el puerto 80, ya que si no se van a hacer conexiones seguras no es necesario activar Apache full ya que te abre el puerto 80 y el 443 que es por dónde se accede de forma segura.



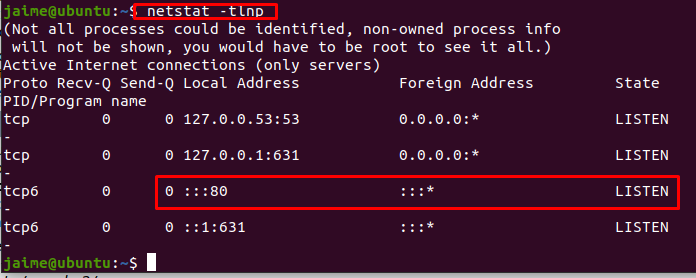
Para habilitar esa configuración se hace con el siguiente comando sudo ufw allow ‘Apache’, en la siguiente imagen veremos cómo se realiza.



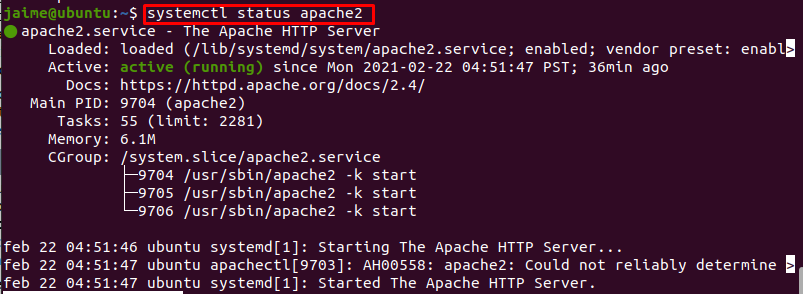
Como podemos apreciar en la siguiente imagen metiendo el la barra de dirección la url de la consola local nos deja entrar en apache y poder trabajar con el.



Con el comando netstat -tlnp vemos la lista de puertos que tenemos a la escucha y muestra como el puerto 80 está a la escucha.

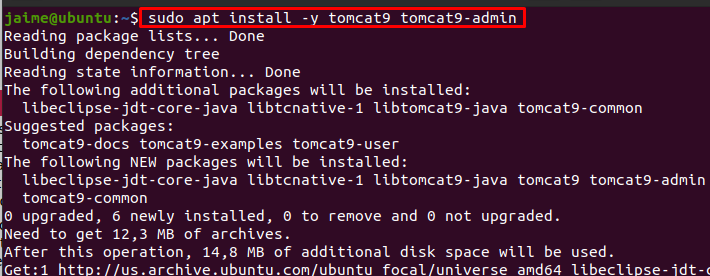


Si se quiere comprobar el estado del apache, con el comando systemctl status apache2 lo podemos verificar



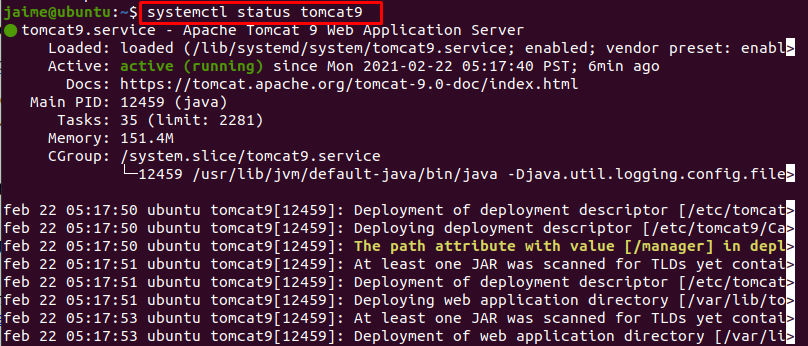
**3.- Verificación e instalación de Tomcat.**

Para comenzar con la instalación del servidor de aplicaciones web Tomcat lo haremos con el siguiente comando sudo apt install -y tomcat9 tomcat9-admin esto nos instalará los paquetes principales al núcleo de tomcat y sus aplicaciones administrativas.

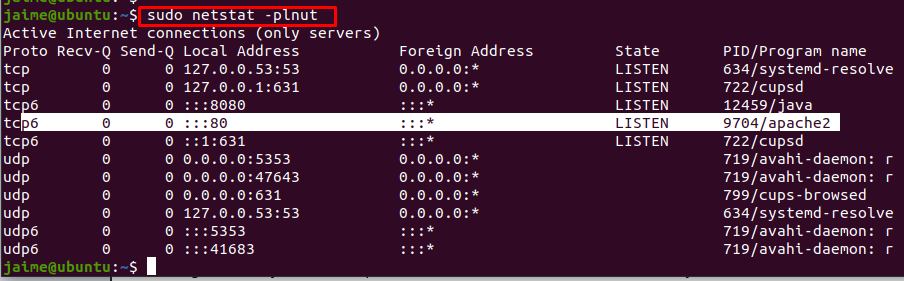


Posteriormente si se quiere también se pueden instalar la documentación y algunos ejemplos de este servidor con los comandos sudo apt install tomcat9-doc y sudo apt install tomcat9-examples.

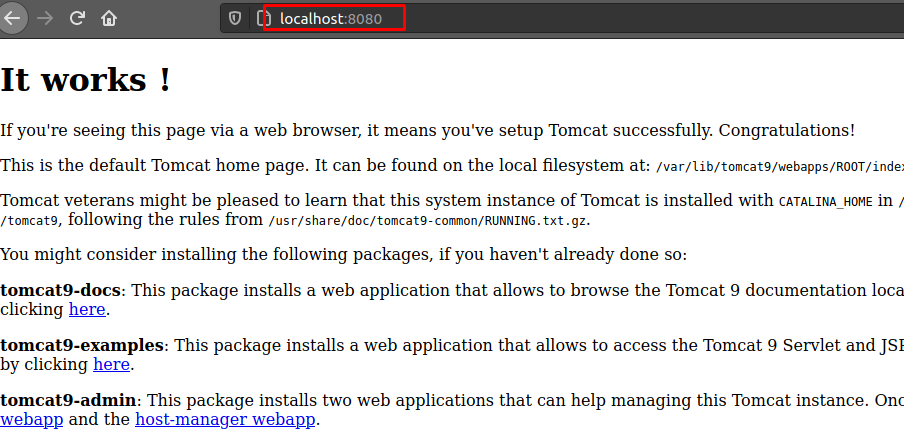
Para verificar el estado del tomcat a través del comando systemctl status tomcat9 o systemctl status tomcat9.service podemos observar que se ha iniciado de forma correcta y está corriendo



Volvemos a comprobar los puertos abiertos para saber si está a la escucha el puerto 8080 que es por el que tenemos que entrar en el tomcat. Con el comando sudo netstat -plunt se puede ver que está a la escucha y que el programa que lo usa es java.



Se abre el navegador de internet y para entrar al tomcat se puede verificar tecleando localhost.8080, si se abre el tomcat es que podemos empezar a trabajar con el.

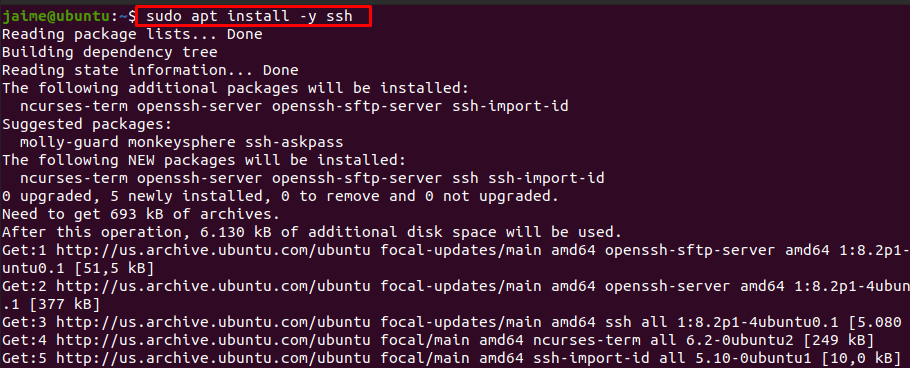


**4.- Instalación de Open SSH**

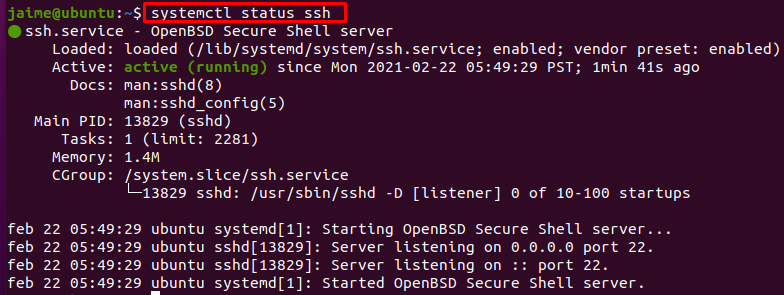
El openssh sirve para realizar conexiones de forma remota y otras comunicaciones especiales de red de forma más segura.

Para instalar ssh desde los repositorios del sistema, primero es aconsejable actualizar todos los paquetes y una vez actualizados comenzar la instalación.

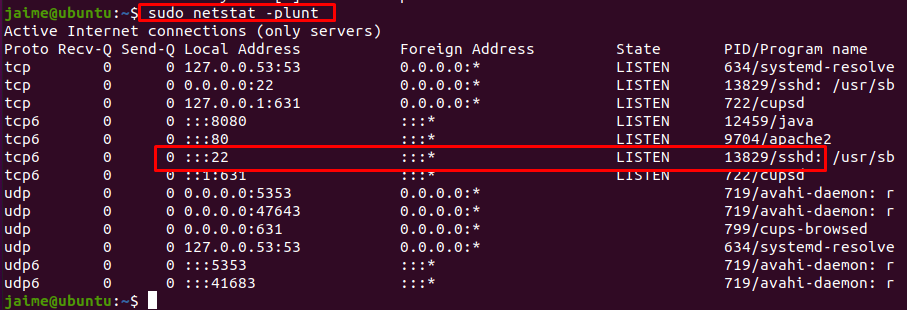
A través de este comando sudo apt install -y ssh instalamos ssh en nuestra máquina virtual.



Una vez instalado se puede comprobar el estado del servicio para corroborar su correcta instalación esta parte se haría a través de este comando systemctl status ssh



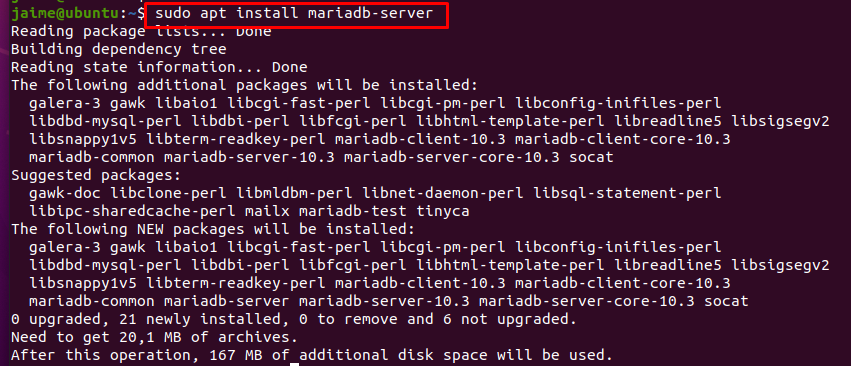
Podemos ver que se ha habilitado el puerto 22 que es por el cual se entra con seguridad al servidor web. Si comprobamos de nuevo los puerto a la escucha veremos habilitado el puerto 22.



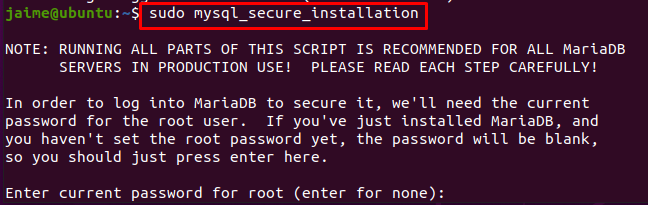
**5.- Instalación de MariaDB**

Para la instalación de MariaDB no se instalará ningún entorno de desarrollo como workbench o xammp, se hará todo por consola y mostraremos que ponemos meter alguna query a través de comandos en el terminal.

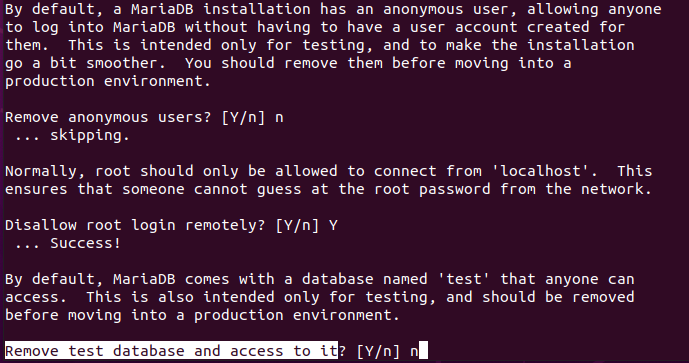
Para comenzar la instalación actualizamos todos los paquetes de Ubuntu con el comando sudo apt update y luego con el siguiente comando sudo apt install mariadb-server instalamos el server de la BBDD.



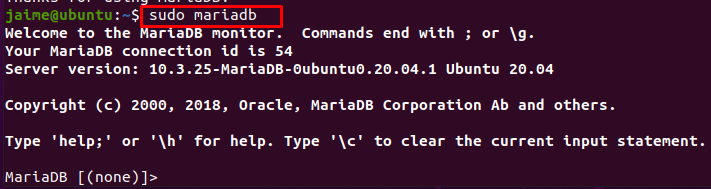
Con siguiente comando sudo mysql\_secure\_installation haremos que MariaDB sea más segura y nos pida una contraseña para poder realizar el inicio de sesión.



De momento no le metemos ninguna contraseña para poder acceder como root sin contraseña y también dejamos la basa de datos de prueba para poder mostrar que se accede y está disponible desde el exterior.

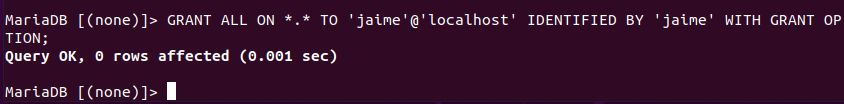


Abrimos maríadb para configurar un usuario y una contraseña. Abrirmos MariaDB con el siguiente comando sudo mariadb y nos mete dentro del servidor.



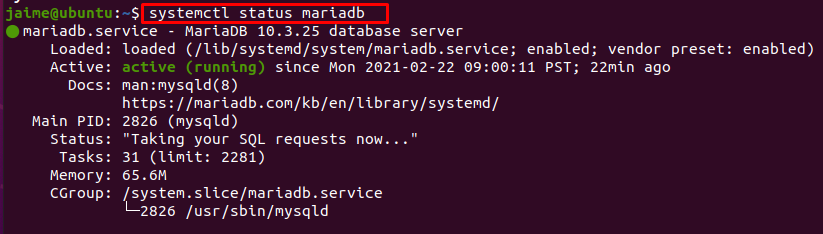
Se procede a crear un usuario con privilegios de root y una contraseña

GRANT ALL ON \*.\* TO 'jaime'@'localhost' IDENTIFIED BY 'jaime' WITH GRANT OPTION



Vaciamos los privilegios para que garantizar que se guarden y estén disponibles en la sesión actual con FLUSH PRIVILEGES

Cerramos Mariadb con exit y probamos la conexión con el usuario y la password configurados anteriormente. Finalmente comprobamos el estado de MariaDB con el siguiente comando systemctl status mariadb y podemos comprobar que esta activo y funciona correctamente.



\*Archivo incluido en el repositorio de github: https://github.com/jjuarez1986/Despliegues.git