**Actividad 2 -Perfil de Servidor**



**Despliegue de Aplicaciones**

**José Ignacio Gutiérrez Cerrato 2-DAW**

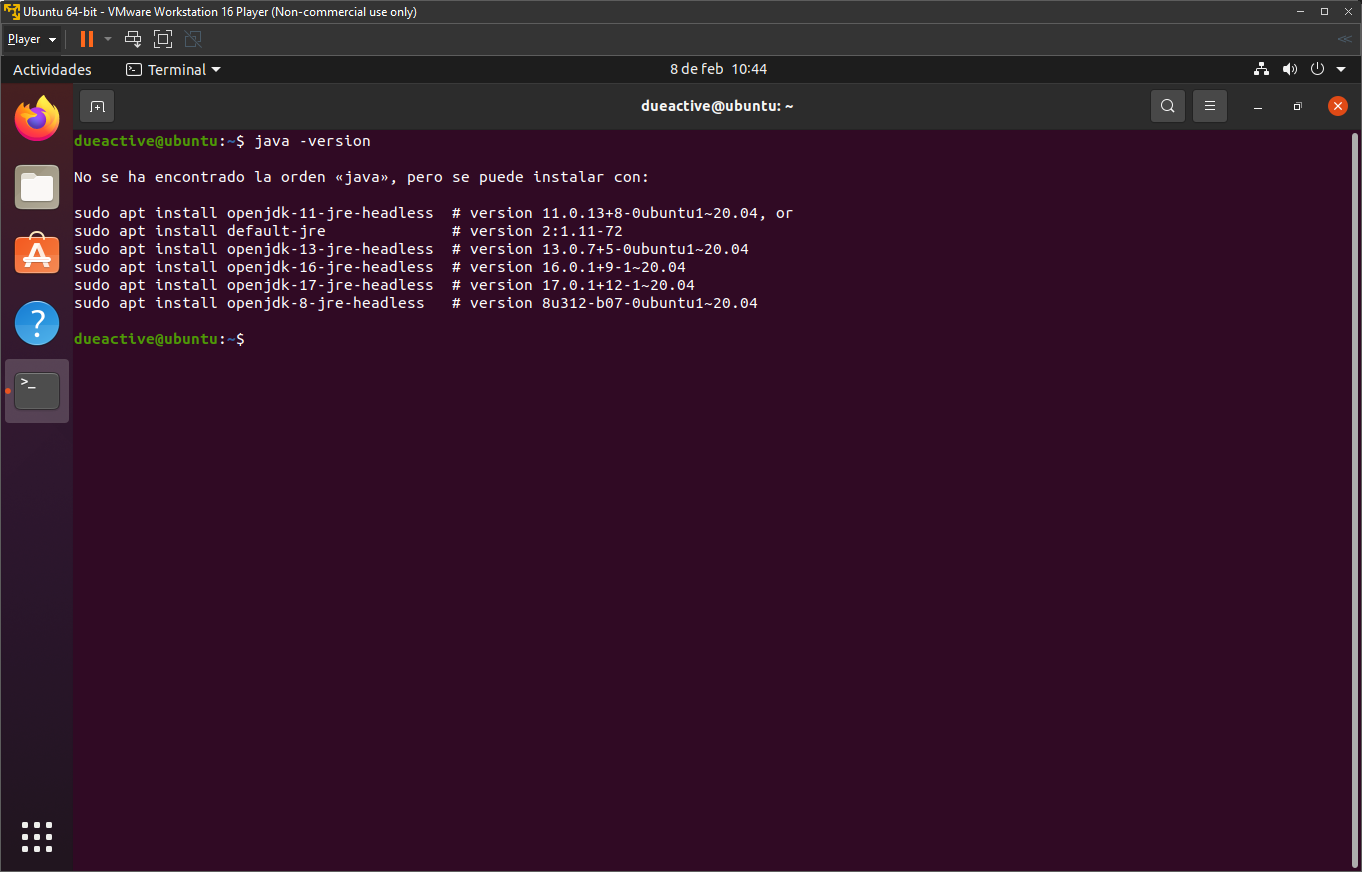
**https://github.com/dueactive/DespliegueAplicaciones.git**

**Requerimiento 1**

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

1. Java

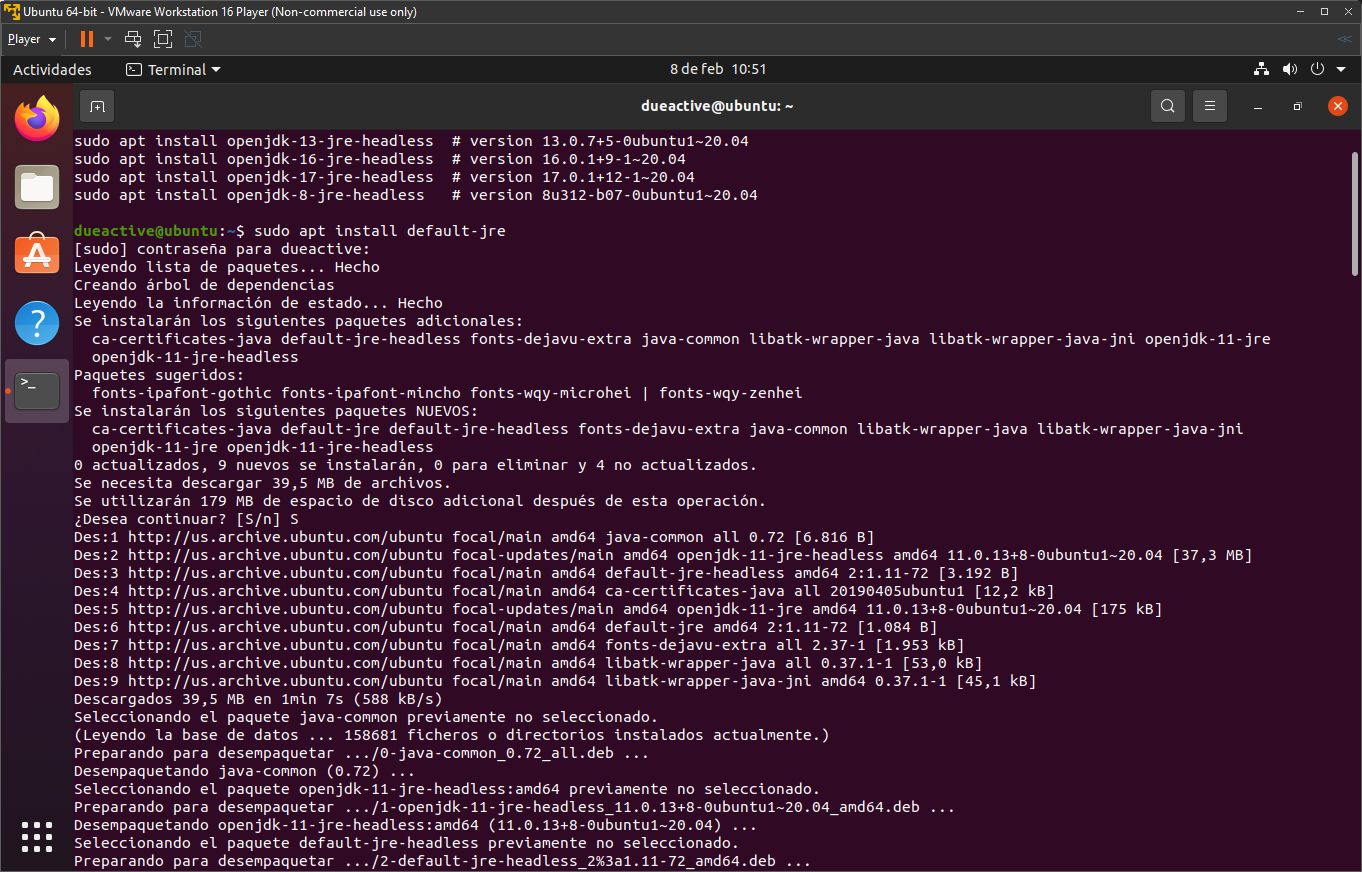
Para validar si tenemos instalado Java usamos el siguiente comando **java -version ,** en mi caso no tengo instalado Java , por lo que procedmos con la instalación.



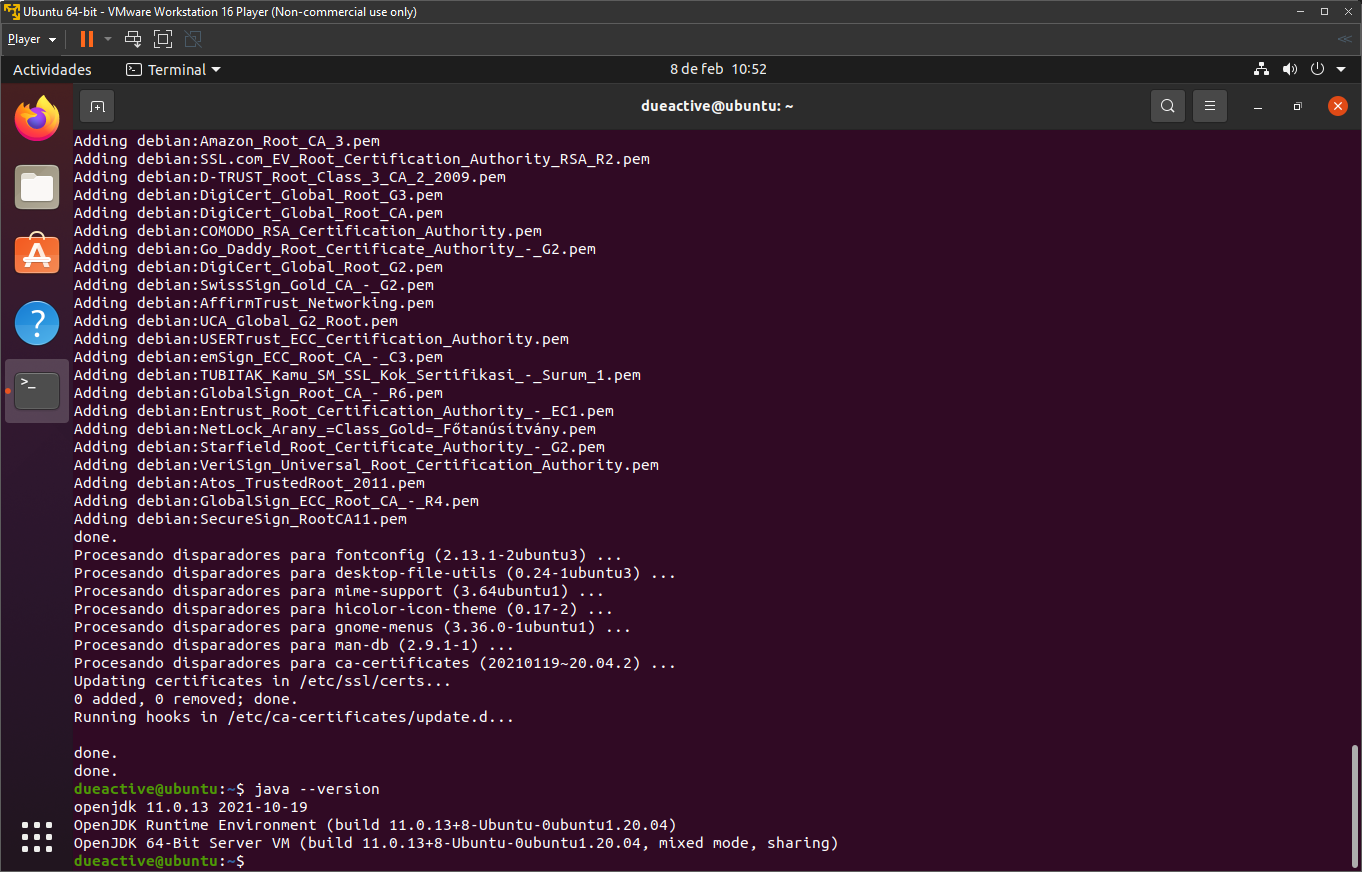
Lo primero que hacemos es actualizar el índice de paquetes

**sudo apt update**

Podemos instalar cualquiera de las versiones que nos indican, en mi caso usaré **sudo apt install default-jre**



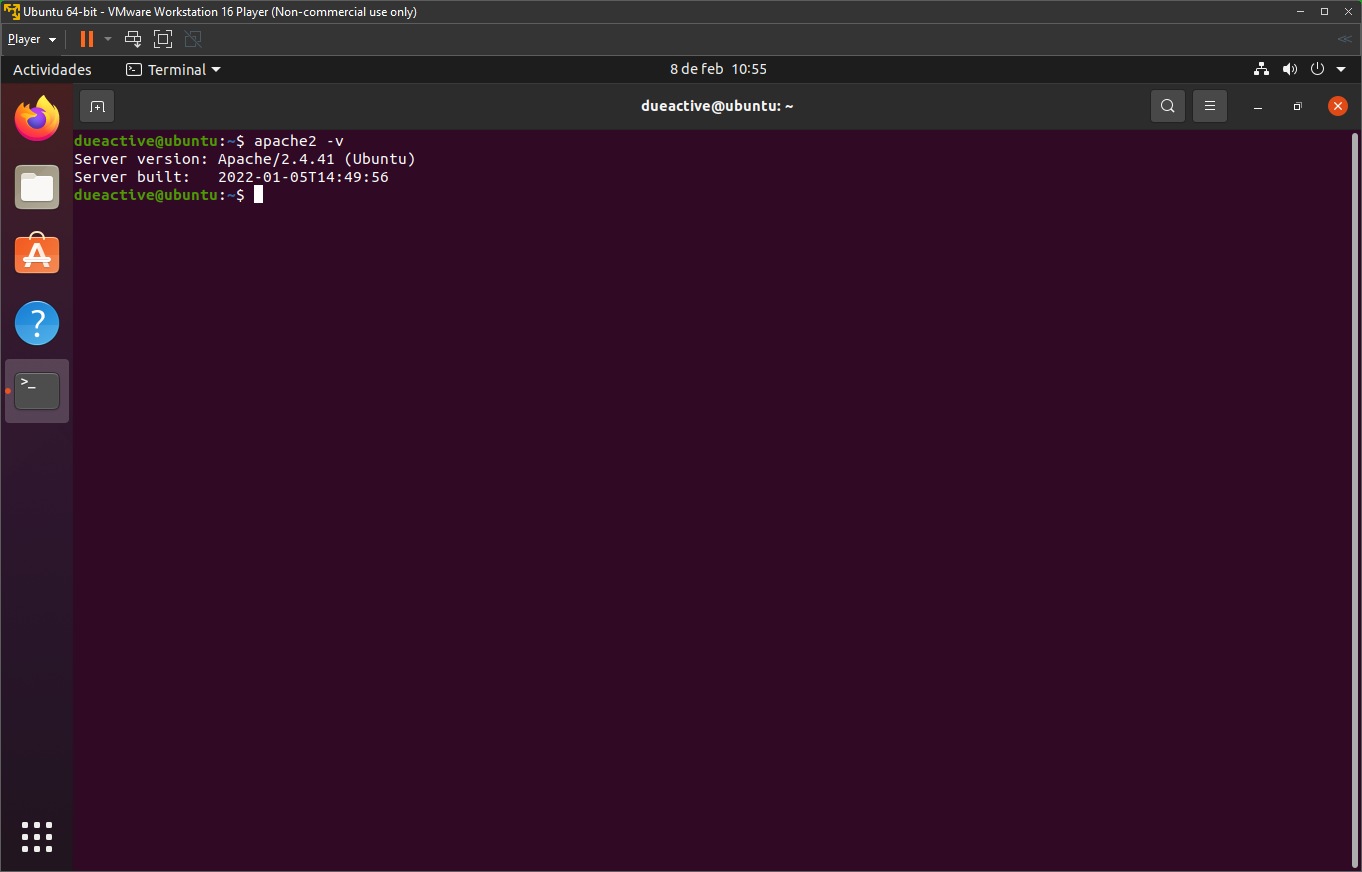
A continuación verificamos la instalación **con java -version**



Y ya tenemos Java instalado.

1. Apache

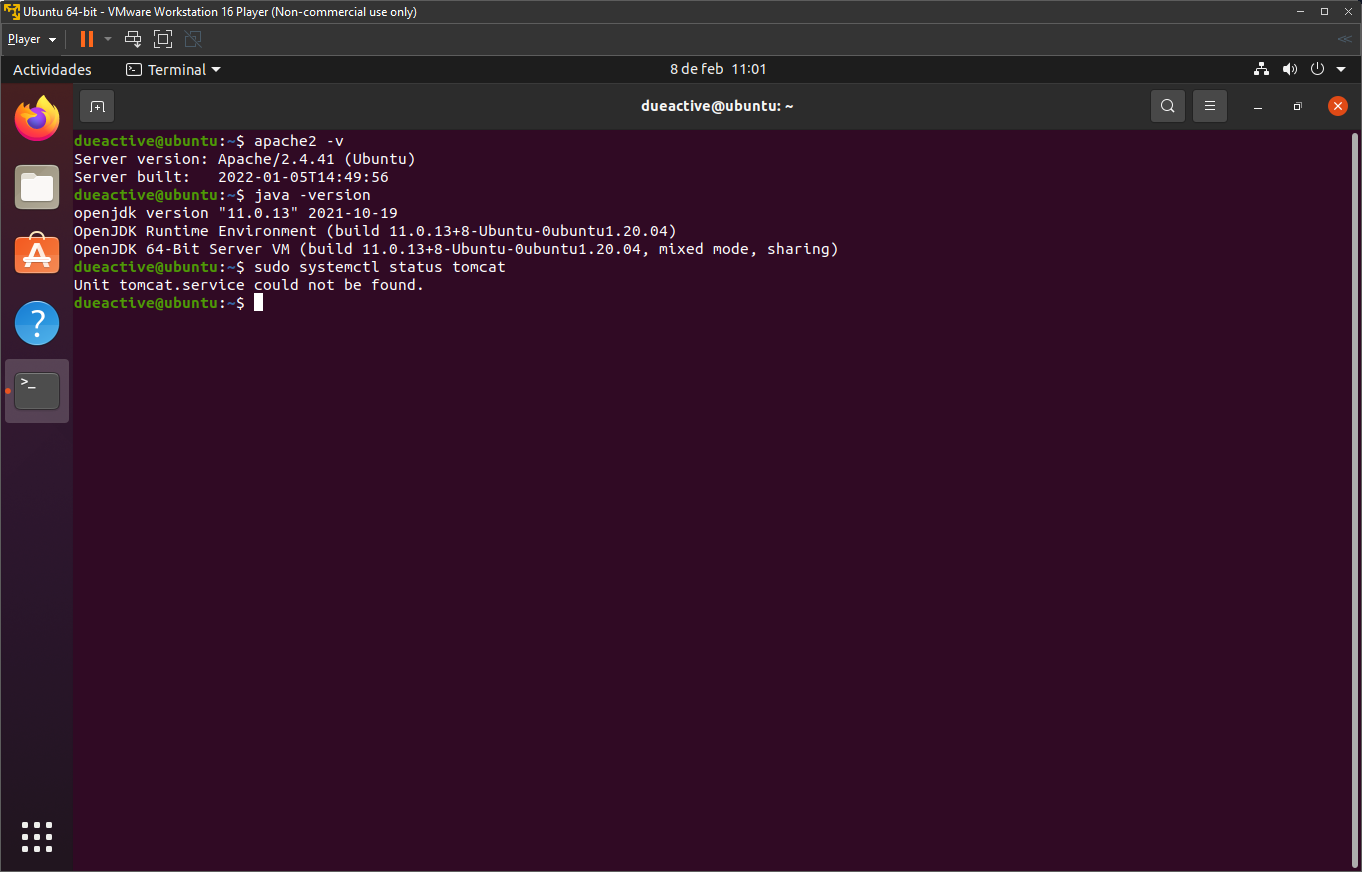
Lo primero que hacemos es verificar si tenemos instalado Apache con el siguiente comando **apache2 -v**



En mi caso ya tengo instalado apache de lo contrario tendría que instalarlo con el comando **sudo apt install apache2**

1. Tomcat

Como en los casos anteriores lo primero es verificar si tenemos instalado Tomcat previamente mediante el comando **sudo systemctl status tomcat el cual nos mosntrará el mensaje en este caso que no encuentra el servicio de tomcat.**



Antes de instalar tomcat necesitamos asegurarnos que tenemos Java instalado, en mi caso lo instalamos anteriormente, en el punto 1.

Como buena práctica de seguridad es recomendable no correr tomcat con el usuario root para ello crearemos un usuario para este fin.

Creamos un grupo y un usuario para tomcat

**sudo groupadd tomcat**

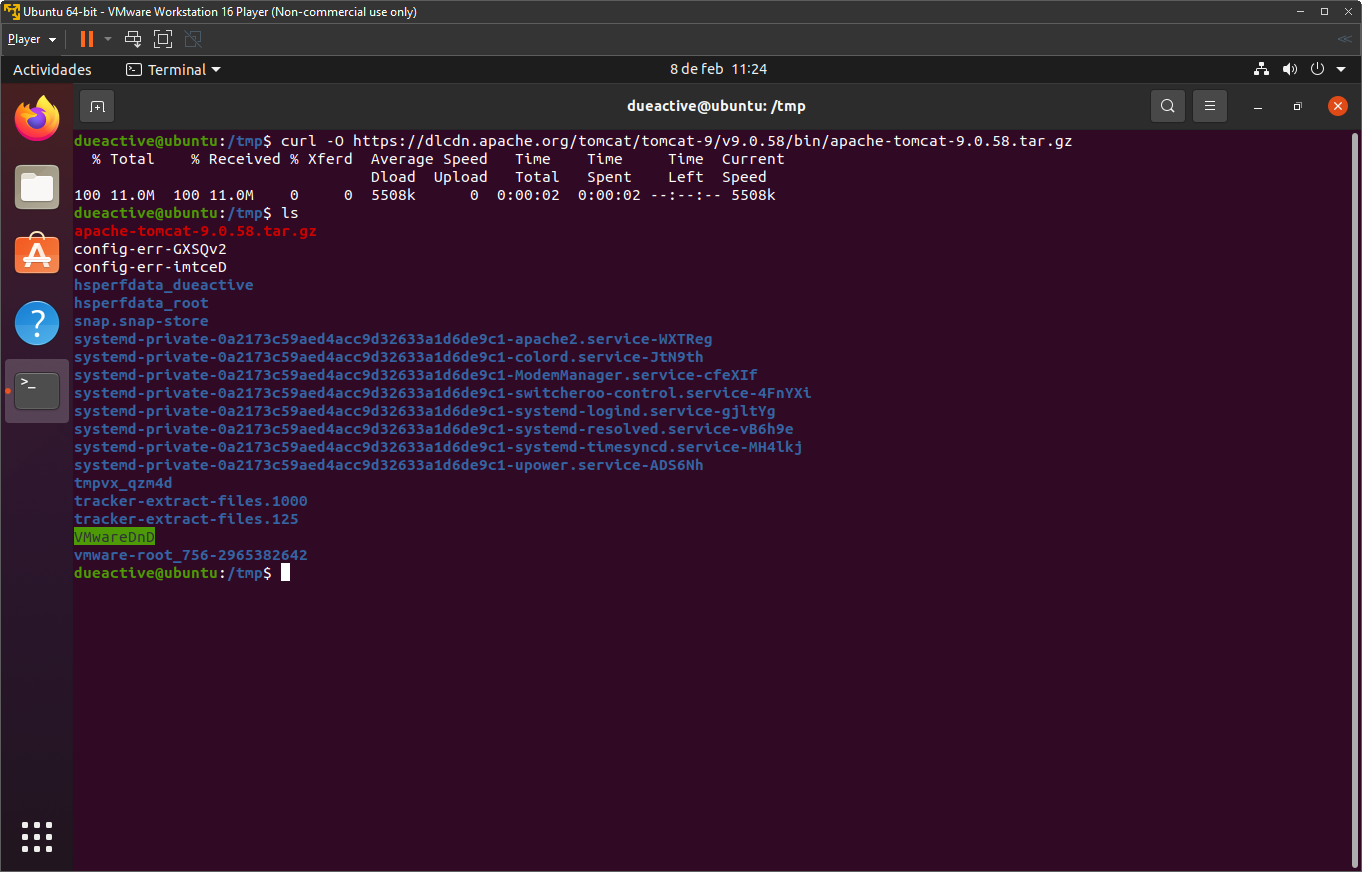
**y a continuación creamos el usuario**

**sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat**

Una vez creado el usuario descargamos tomcat.

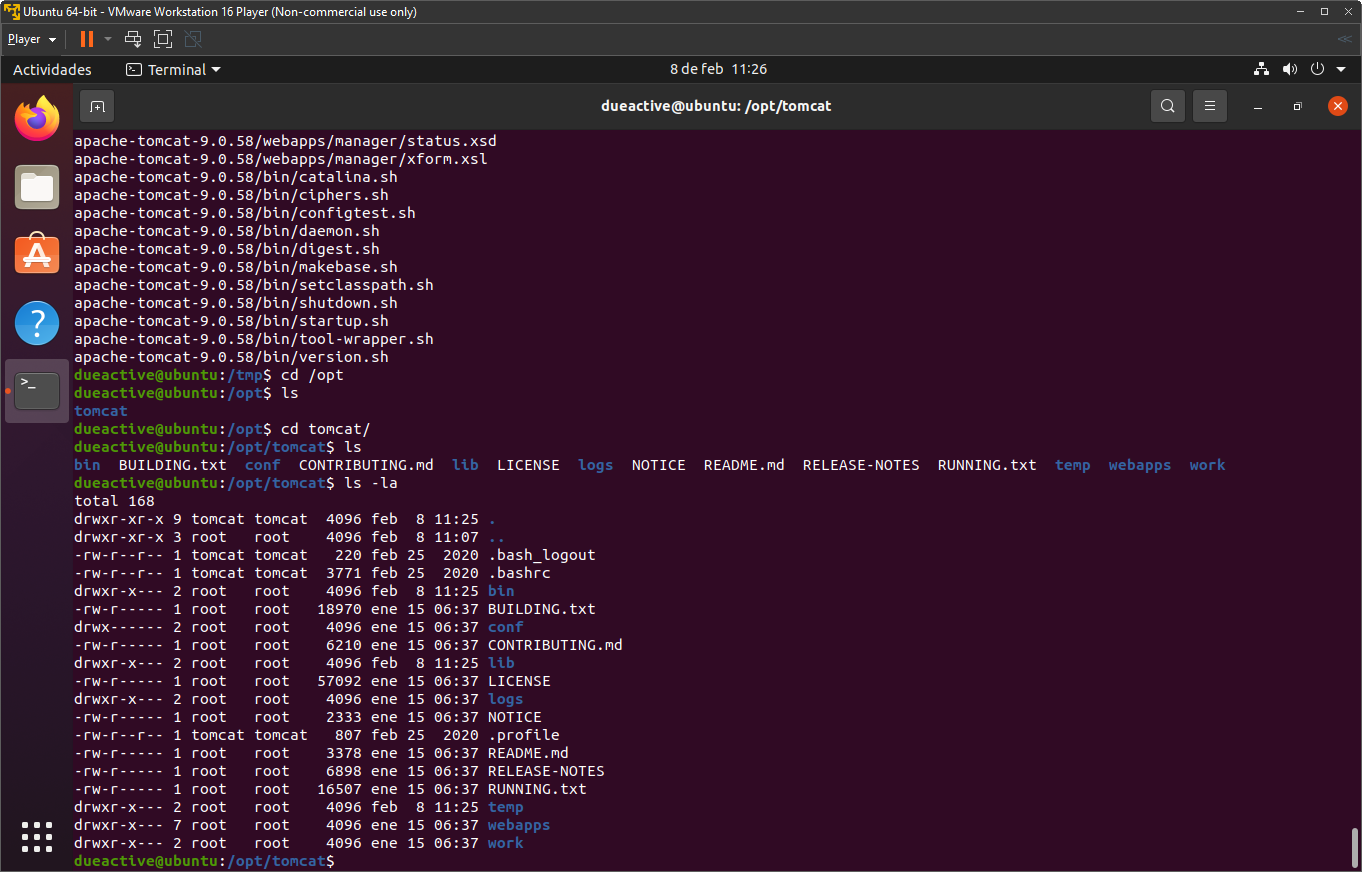
Nos posicionamos en el directorio /tmp desde el que descargaremos tomcat.

**curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.58/bin/apache-tomcat-9.0.58.tar.gz**



Una vez descargado, instalaremos tomcat en el directorio /opt/tomcat

**sudo tar xzvf apache-tomcat-\*tar.gz -C /opt/tomcat --strip-components=1**



Asignamos los permisos sobre la instalación al grupo tomcat

**sudo chgrp -R tomcat /opt/tomcat**

A continuación, proporcionamos al grupo tomcat acceso de lectura al directorio conf y a todos sus contenidos, y acceso de **ejecución** al directorio:

**sudo chmod -R g+r conf**

**sudo chmod g+x conf**

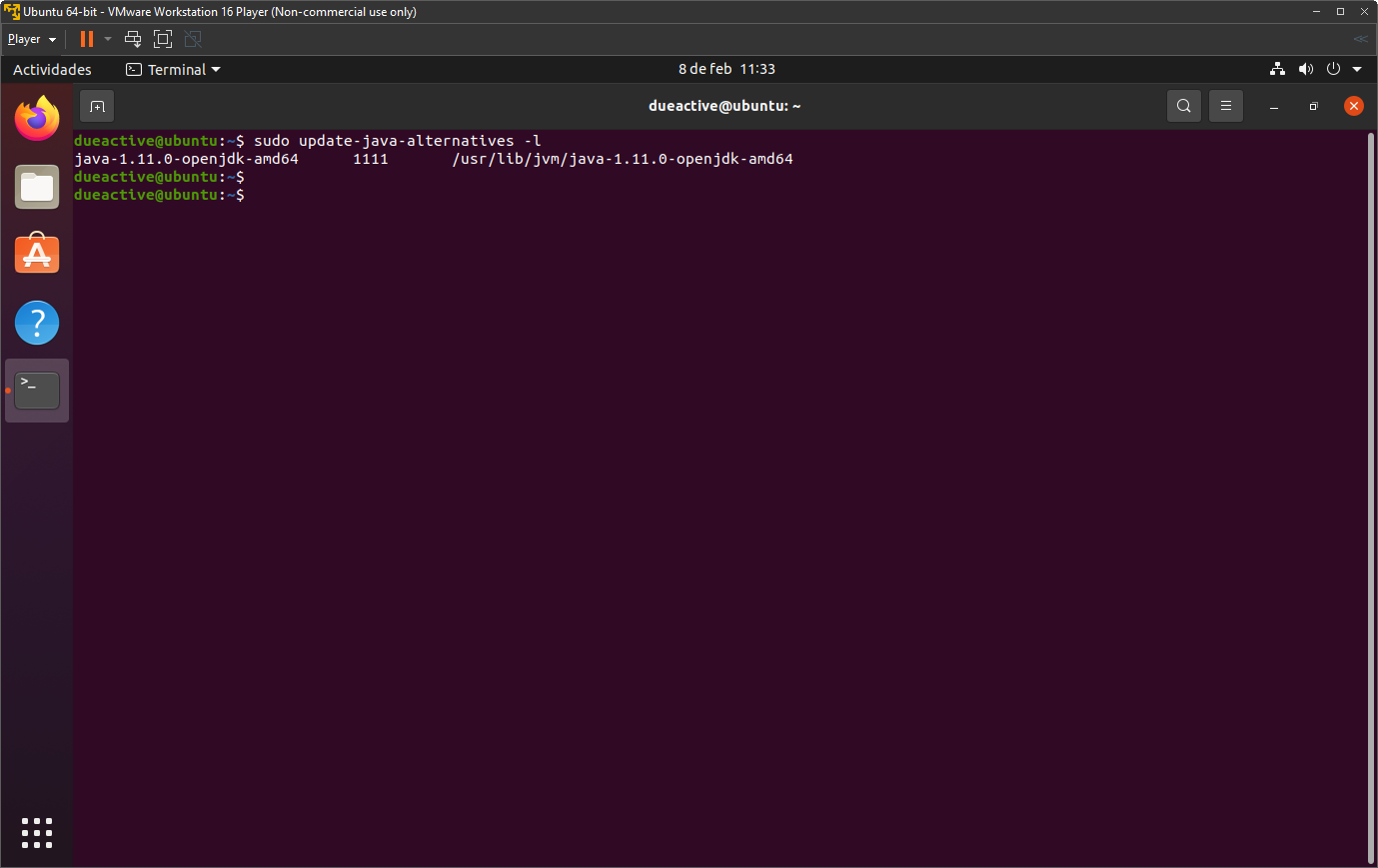
Nos aseguramos de que el usuario tomcat sea el propietario de los directorios webapps, work, temp y logs

**sudo chown -R tomcat webapps/ work/ temp/ logs/**

Ahora que están configurados los permisos, podemos crear un archivo de servicio systemd para gestionar el proceso de Tomcat.

Tomcat necesita saber dónde está instalado Java , Esta ruta se denomina comúnmente “JAVA\_HOME”. La alternativa más sencilla para revisar esa ubicación es ejecutar el siguiente comando:

**sudo update-java-alternatives -l**



Con esta información, podemos crear el archivo de servicio systemd. Abra un archivo llamado tomcat.service en el directorio /etc/systemd/system escribiendo lo siguiente :

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

\*\*Usamos el editor de texto nano , aunque podemos usar vim o alguno similar.

\*\*\*\*\* Begin fichero \*\*\*\*\*\*\*\*

[Unit]

Description=Apache Tomcat Web Application Container

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\_BASE=/opt/tomcat

Environment='CATALINA\_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC'

Environment='JAVA\_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'

ExecStart=/opt/tomcat/bin/startup.sh

ExecStop=/opt/tomcat/bin/shutdown.sh

User=tomcat

Group=tomcat

UMask=0007

RestartSec=10

Restart=always

[Install]

WantedBy=multi-user.target

\*\*\*\*\* End fichero \*\*\*\*\*\*\*\*

A continuación, cargamos de el demonio systemd para que reciba información sobre nuestro nuevo archivo de servicio:

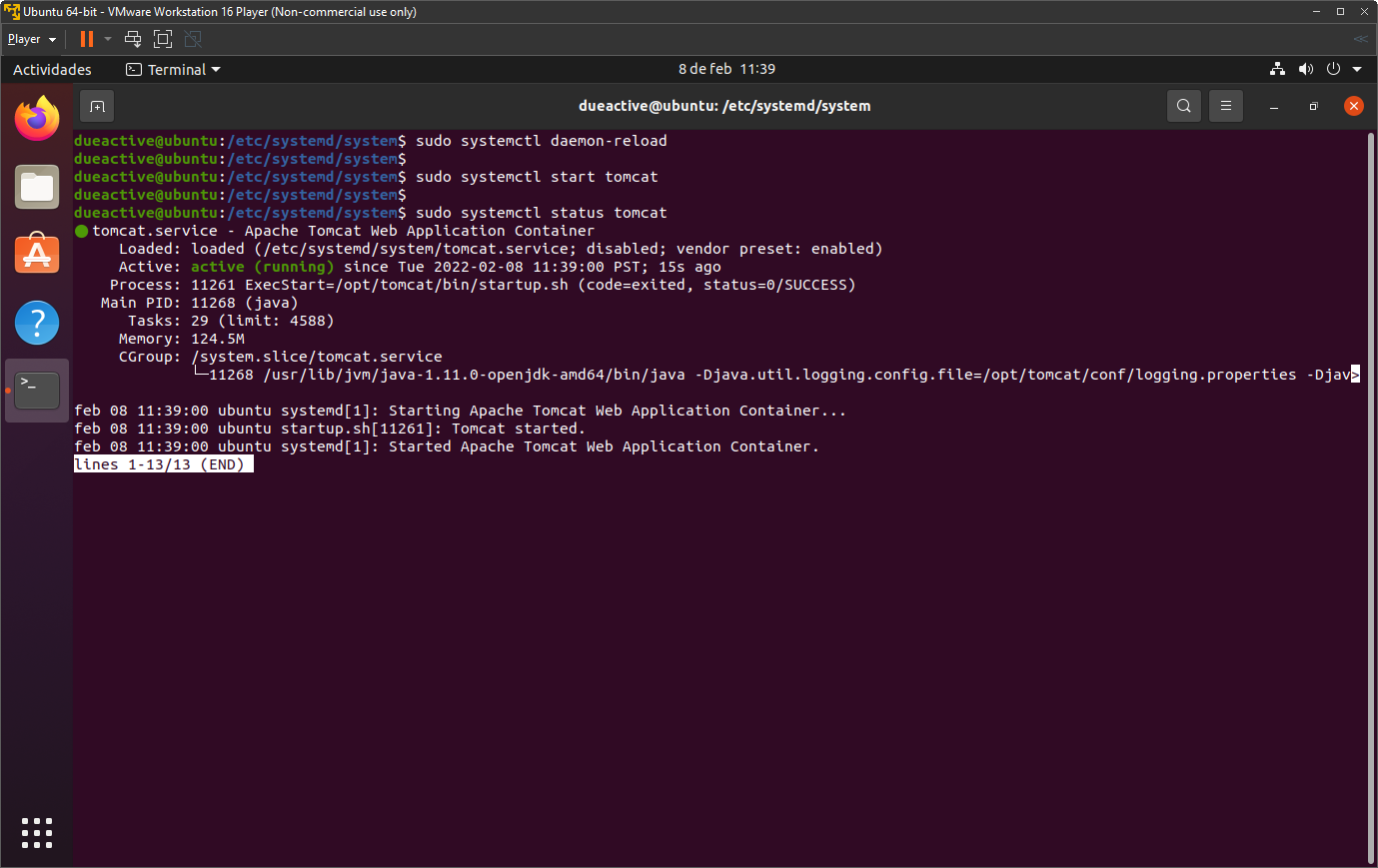
sudo systemctl daemon-reload

Iniciamos el servicio Tomcat escribiendo lo siguiente:

**sudo systemctl start tomcat**

Comprobamos que se inició sin errores escribiendo lo siguiente:

**sudo systemctl status tomcat**

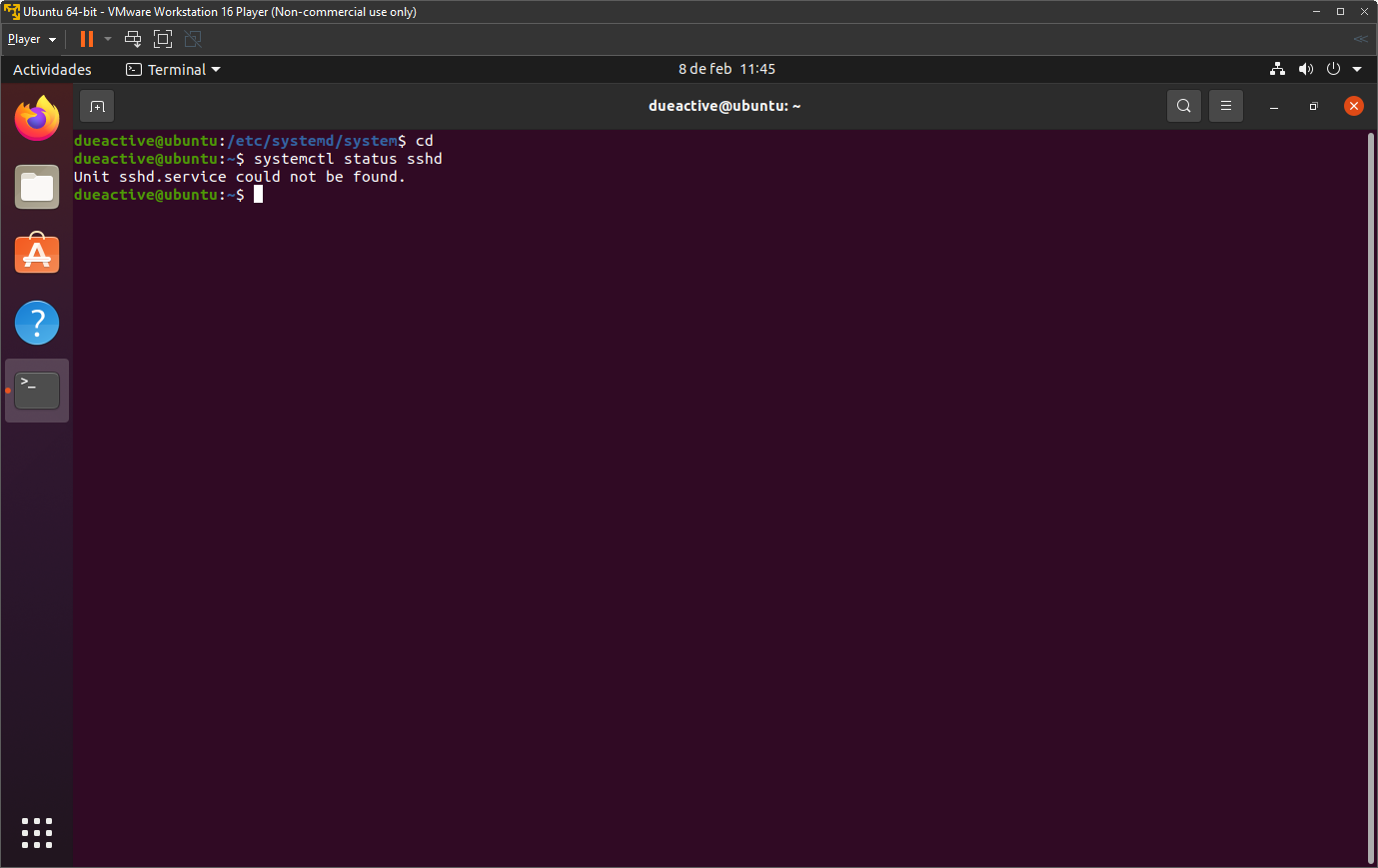


1. openSSH

Actualuizamos el índice de paquetes

**sudo apt update**

Parra validar que no tenemos instalado ssh podemos preguntar por el estado del servicio sshd y nos dirá que no lo encuentra.



A continuación, procedemos con la instalación mediante el comando

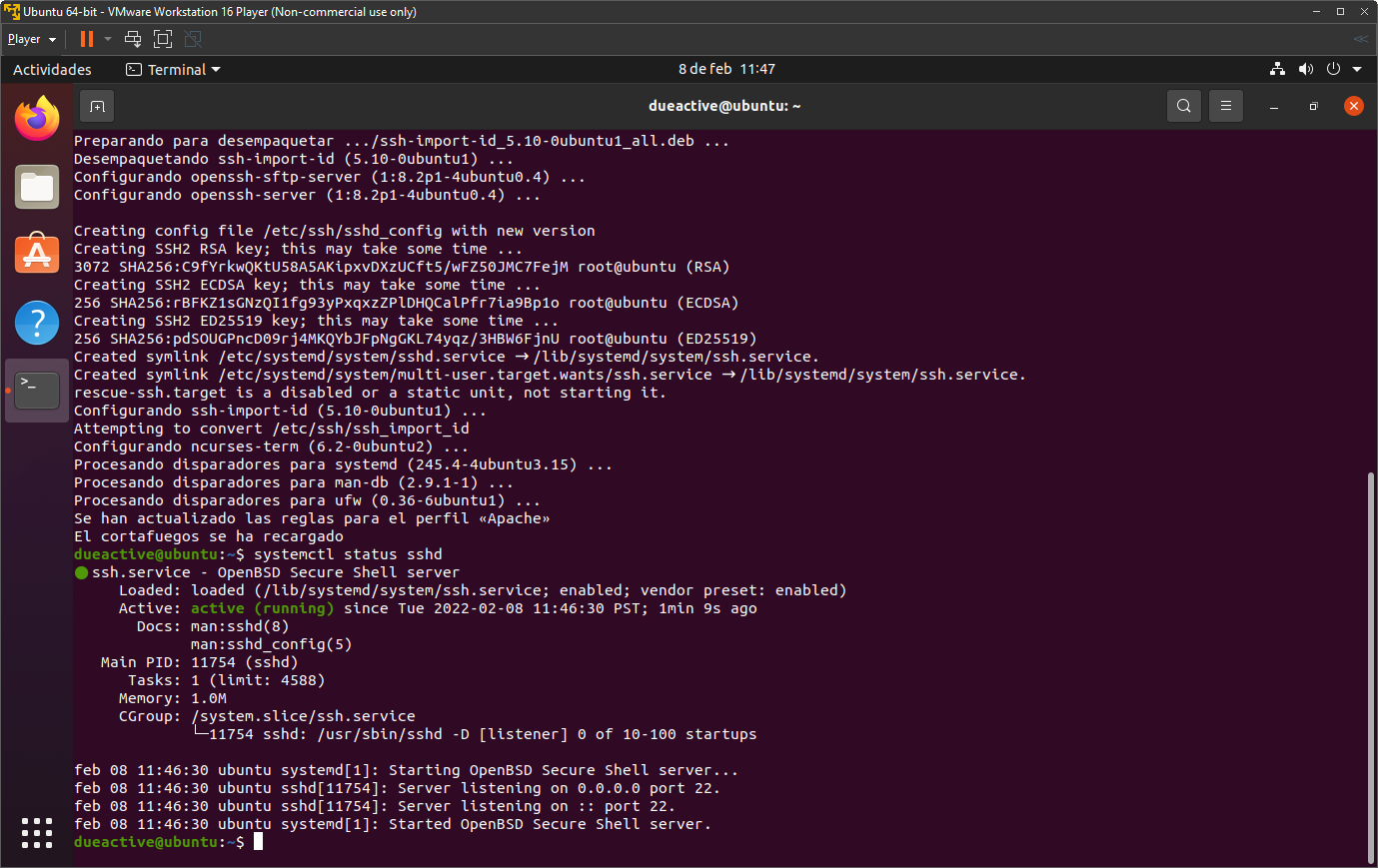
**sudo apt install openssh-server**



Para validar la instalación podemos preguntar de nuevo por el servicio sshd

**systemctl status sshd**

En este caso nos mostrará que está activo.



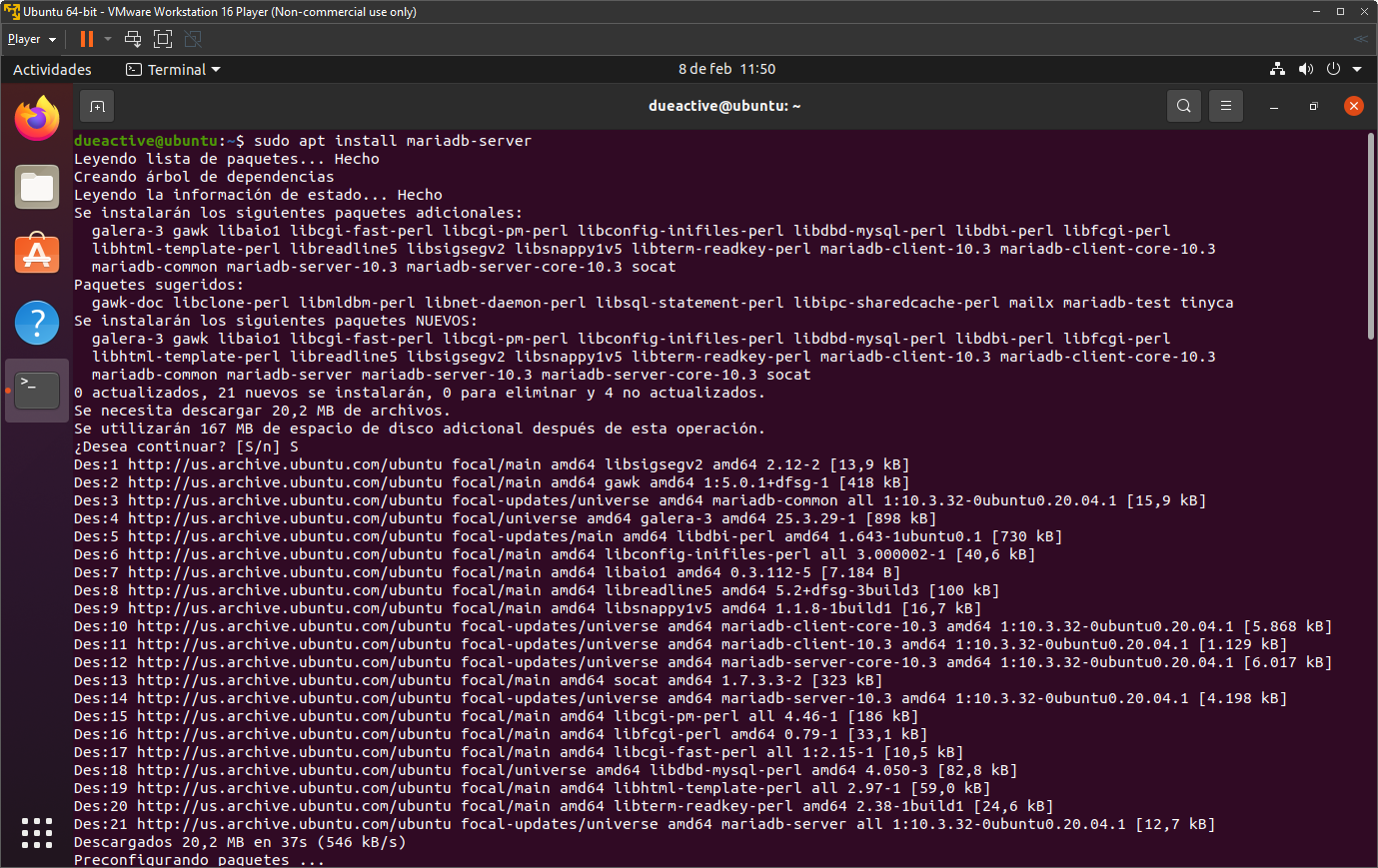
1. MariaDB

Lo primero que hacemos es actualizar el índice de paquetes

**sudo apt update**

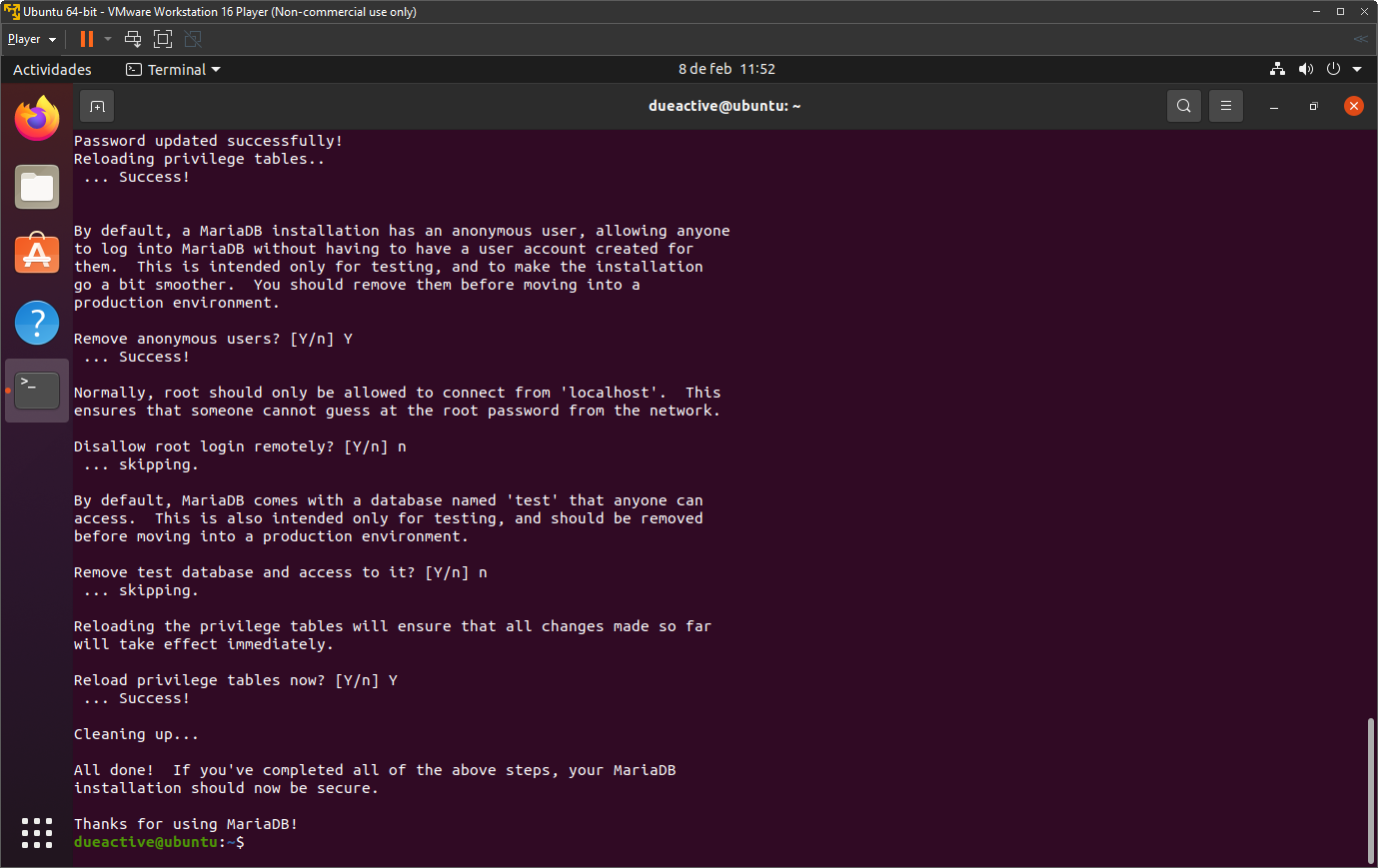
Luego instalamos el paquete

**sudo apt install mariadb-server**



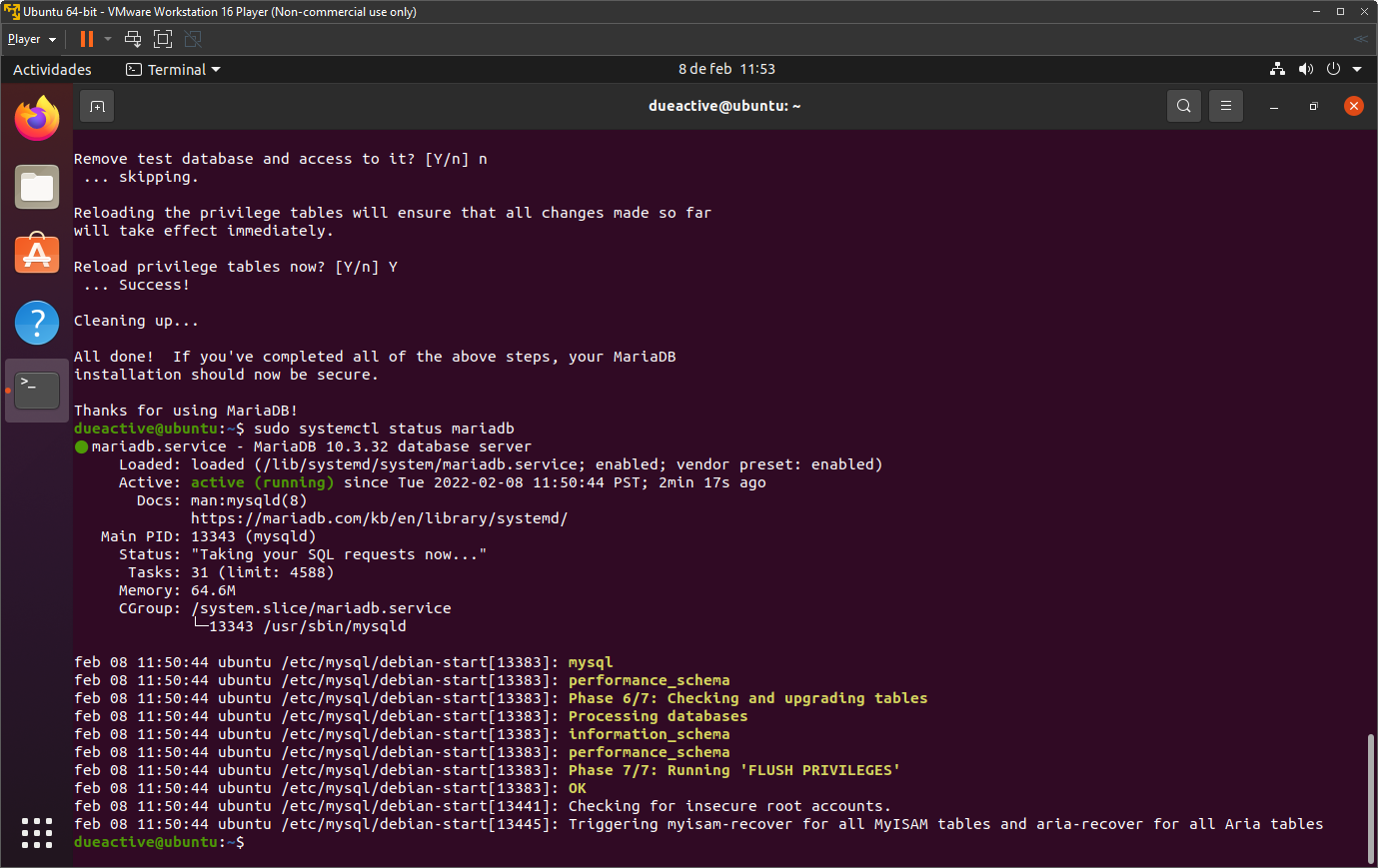
En las nuevas instalaciones de MariaDB, el siguiente paso es ejecutar la secuencia de comandos de seguridad incluida

**sudo mysql\_secure\_installation**



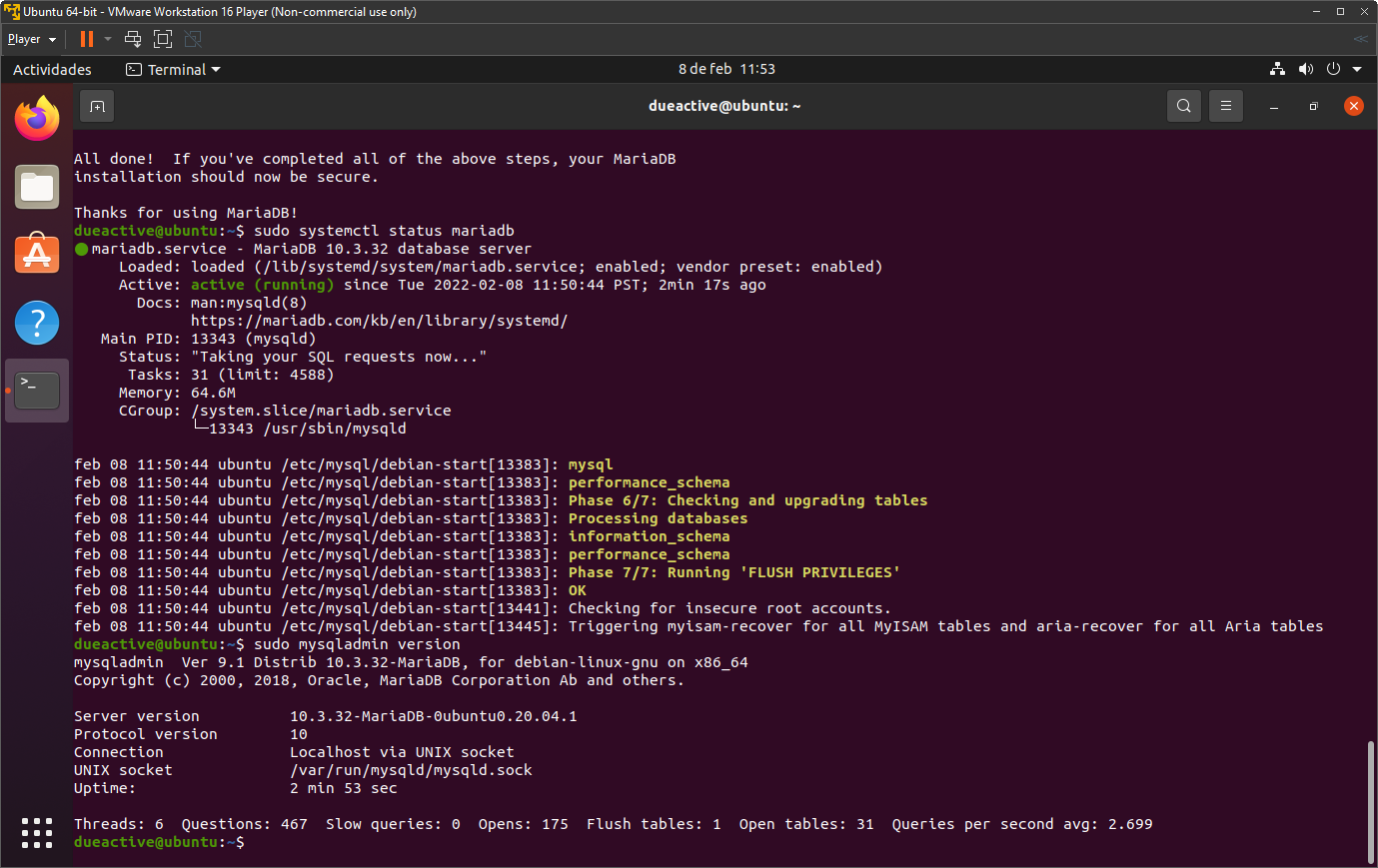
A continuación, podemos ver si el servicio está corriendo correctamente mediante el comando

**sudo systemctl status mariadb**



Como comprobación adicional, puede intentar establecer conexión con la base de datos usando la herramienta **mysqladmin**, que es un cliente que le permite ejecutar comandos administrativos.

**sudo mysqladmin version**



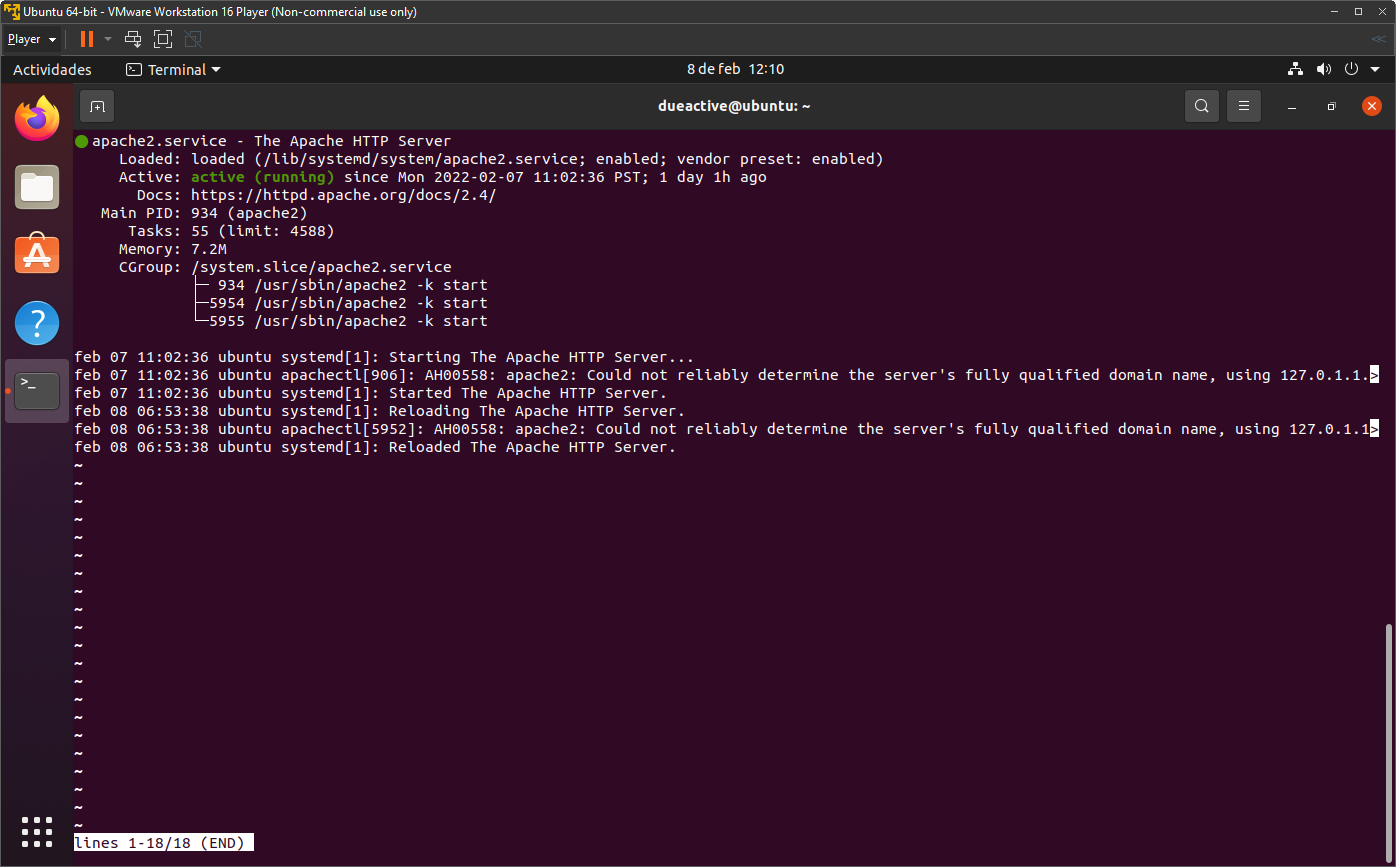
**Requerimiento 2**

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.

Apache

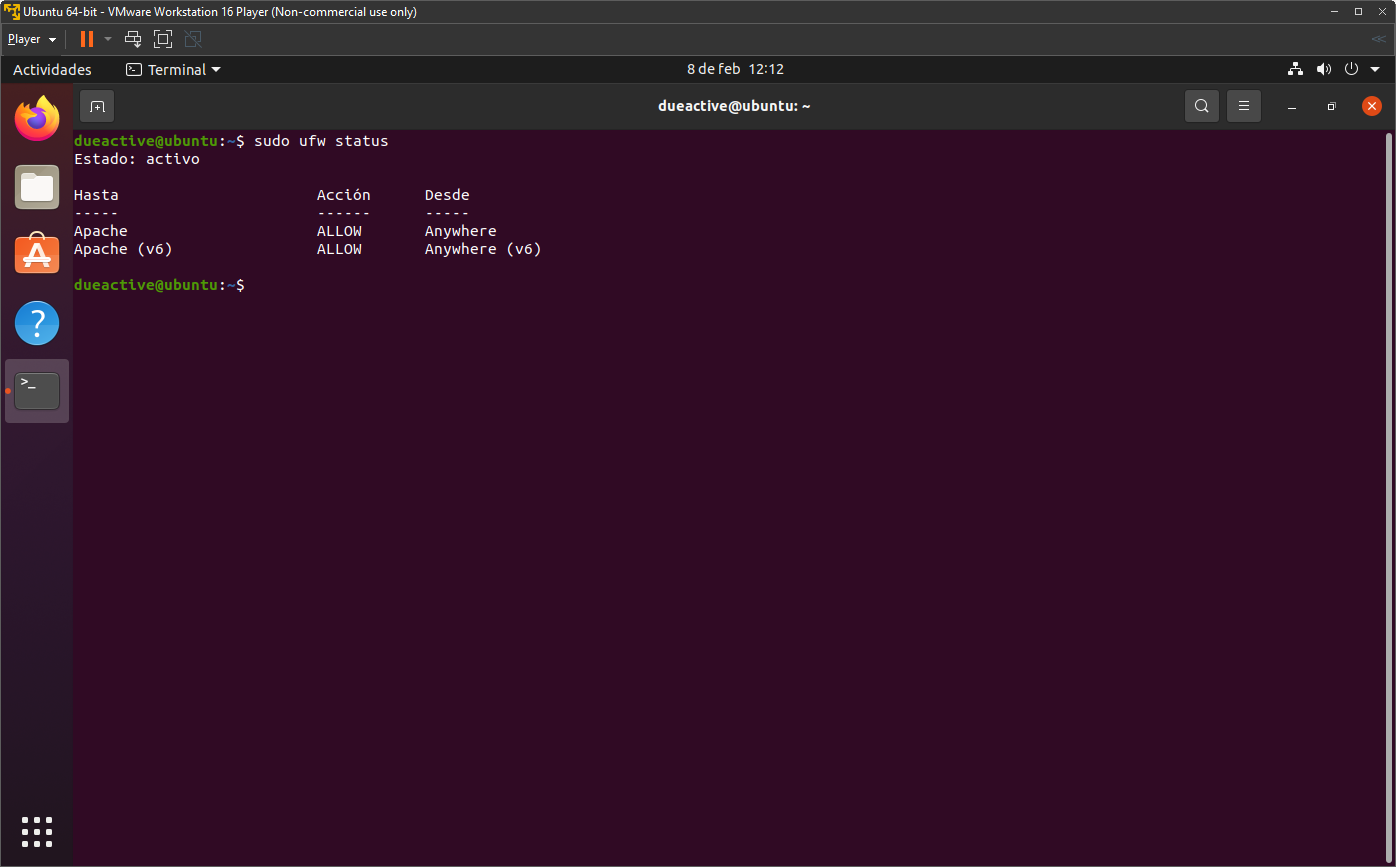
Primero vemos que el servicio de apache está correindo correctamente mediante el comando

sudo systemctl status apache2



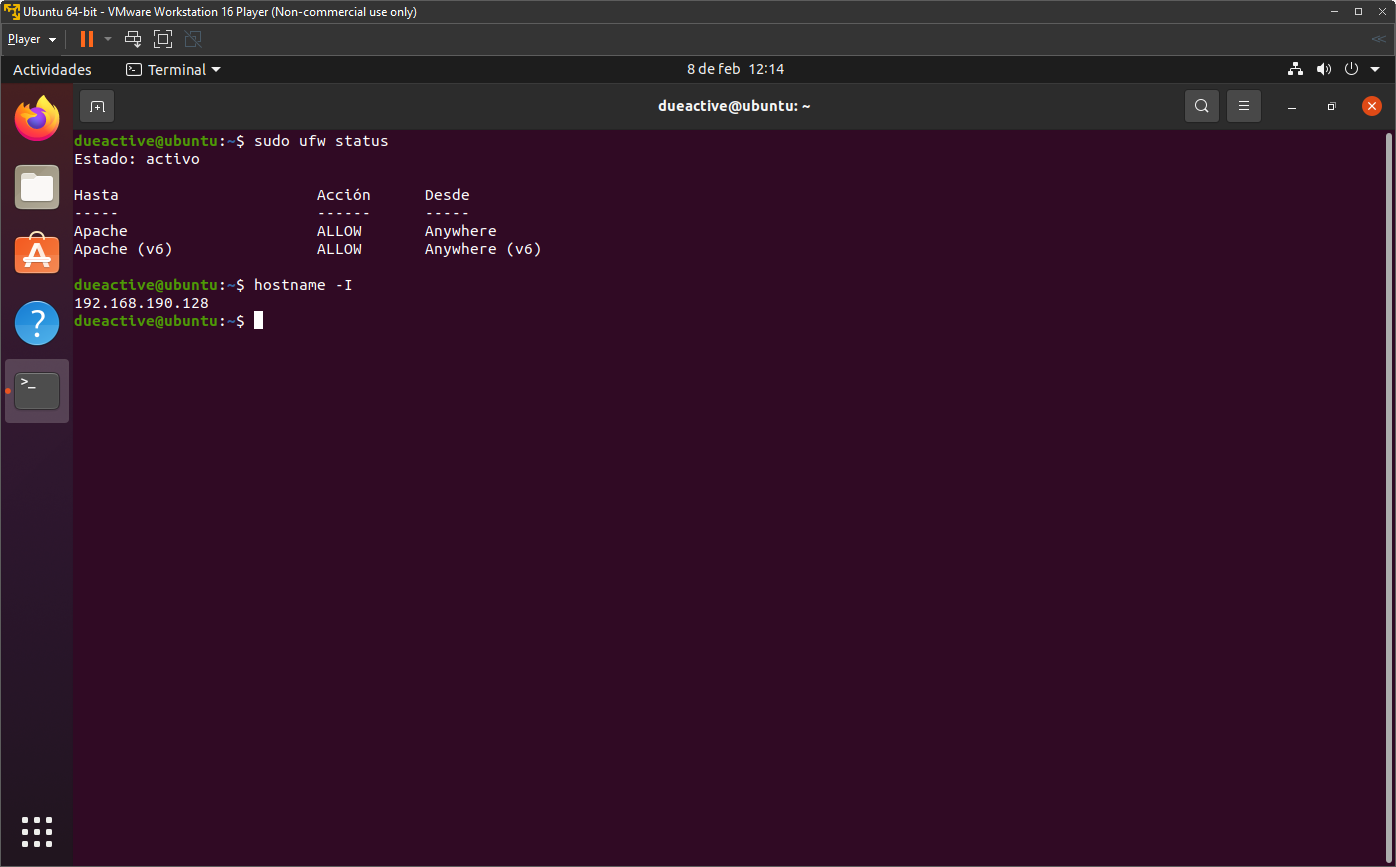
A continuación, vemos si tiene visibilidad desde el exterior revisando las reglas de firewall.

**sudo ufw status**

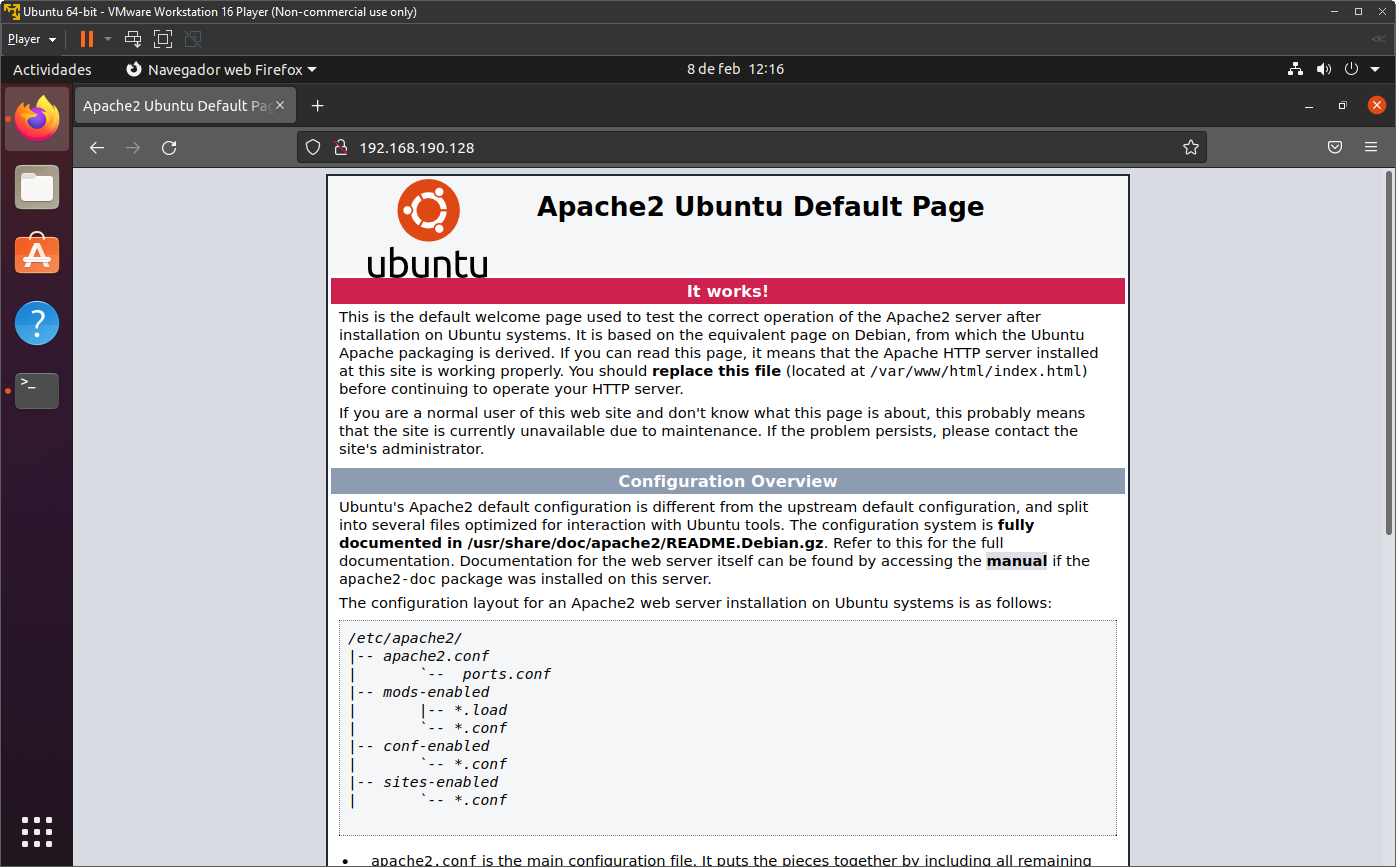


En caso de tener que habilitar las reglas de firewall para apache usaremos el comando **sudo ufw allow 'Apache'**

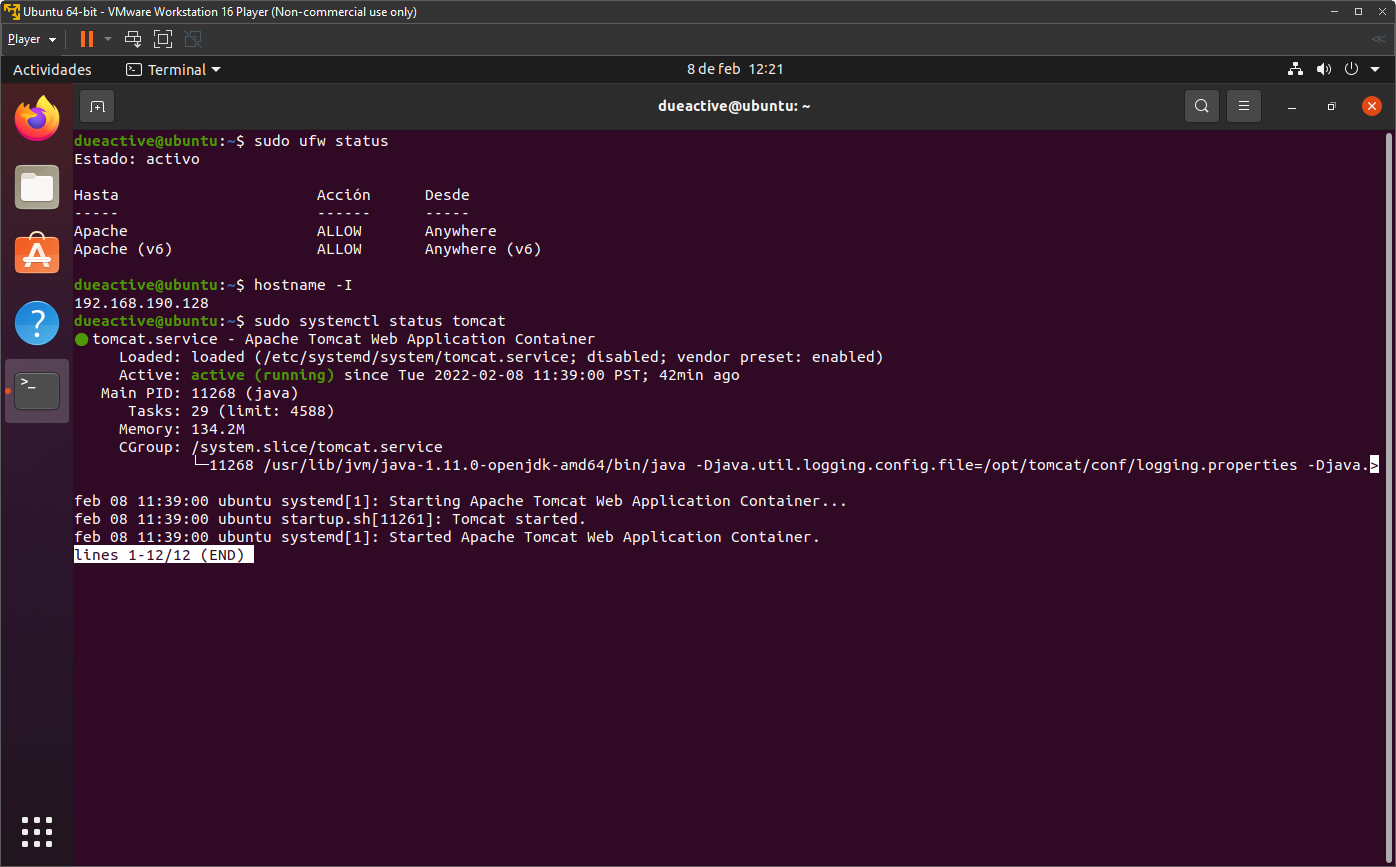
Verificamos nuestra ip de salida con **hostname -I**



Y accedemos desde el navegador a la ip facilitado y veremos que está corriendo Apache correctamente.

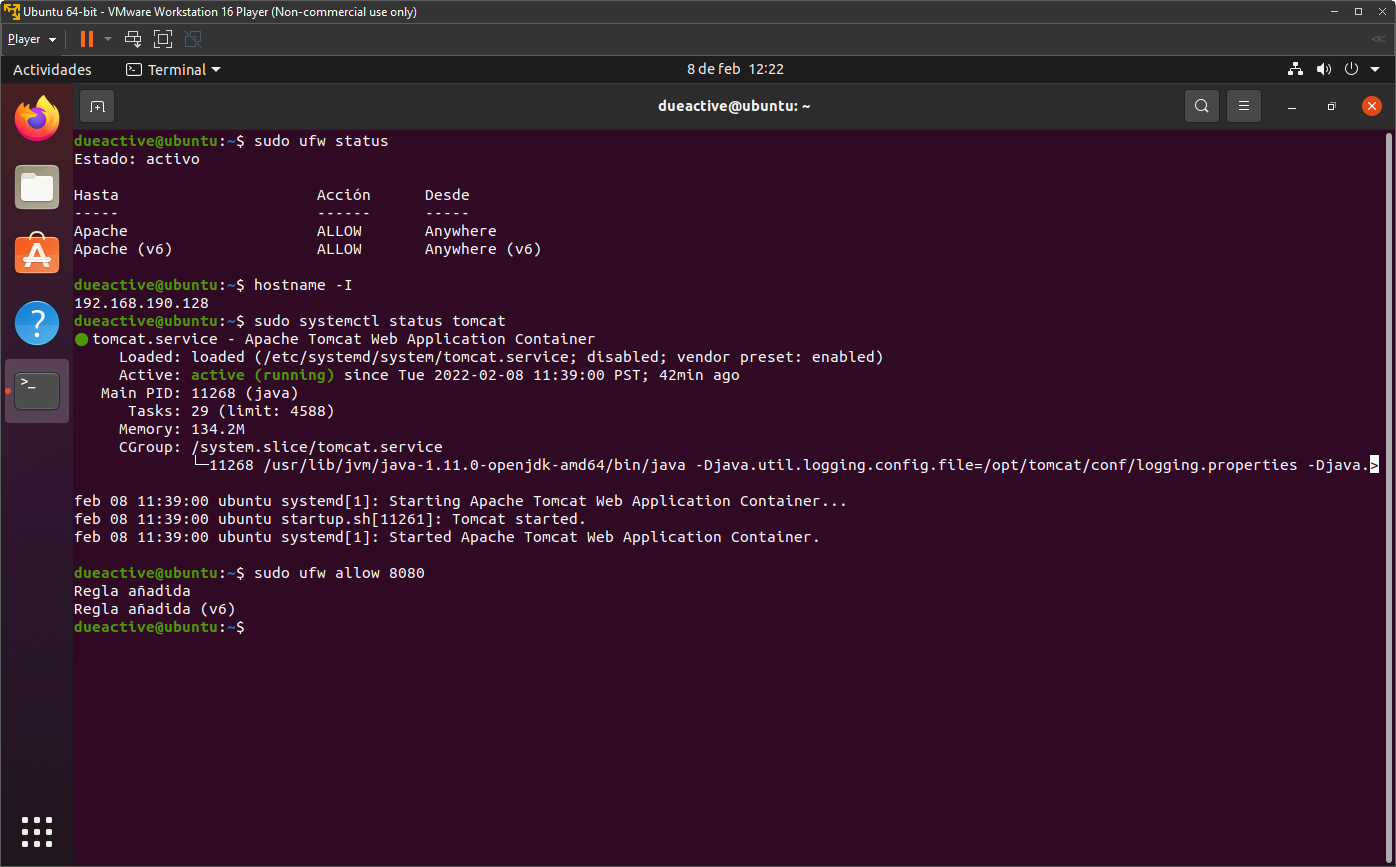


Tomcat

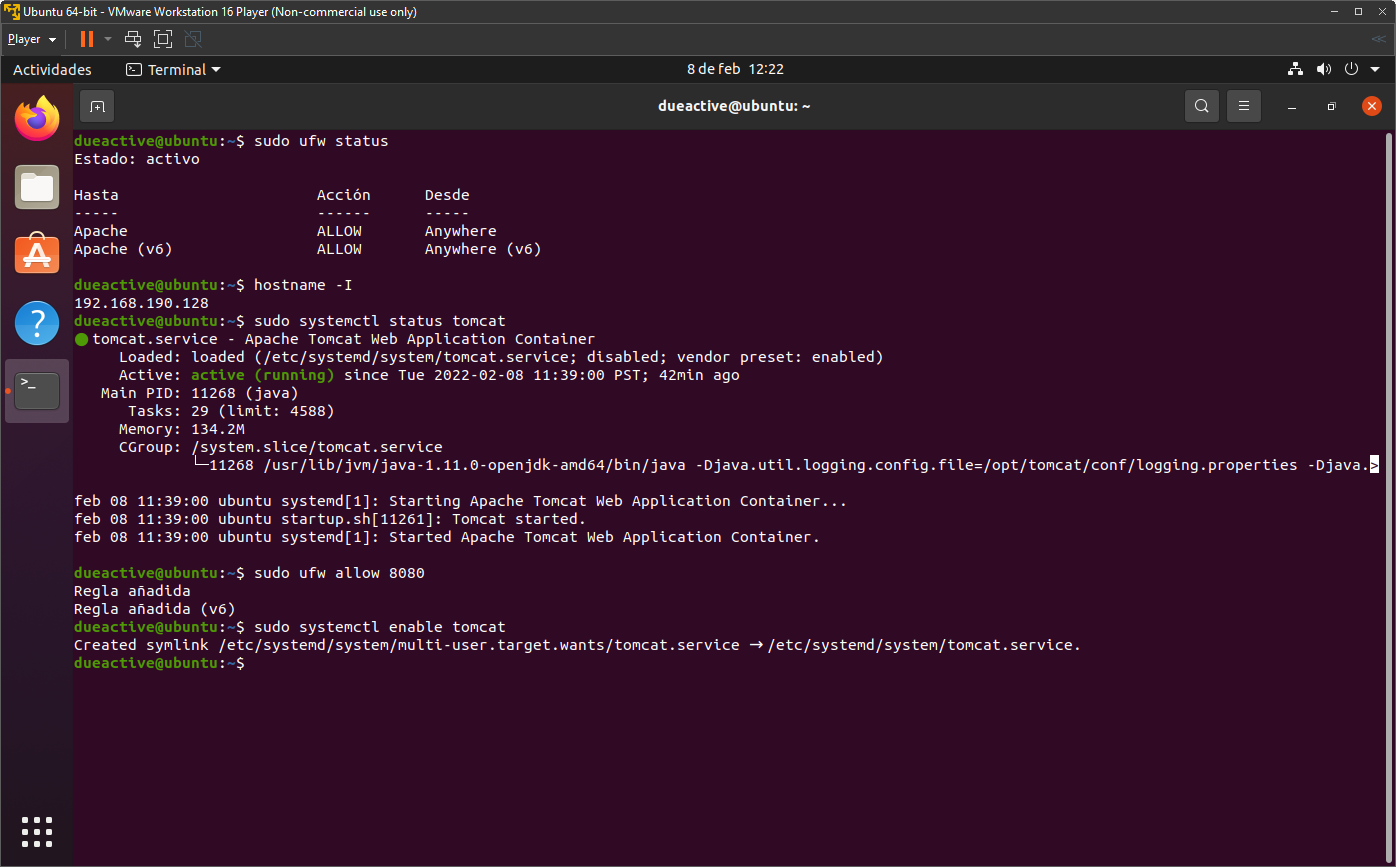


Para acceder desde el exterior necesitamos exponemos el puerto 8080

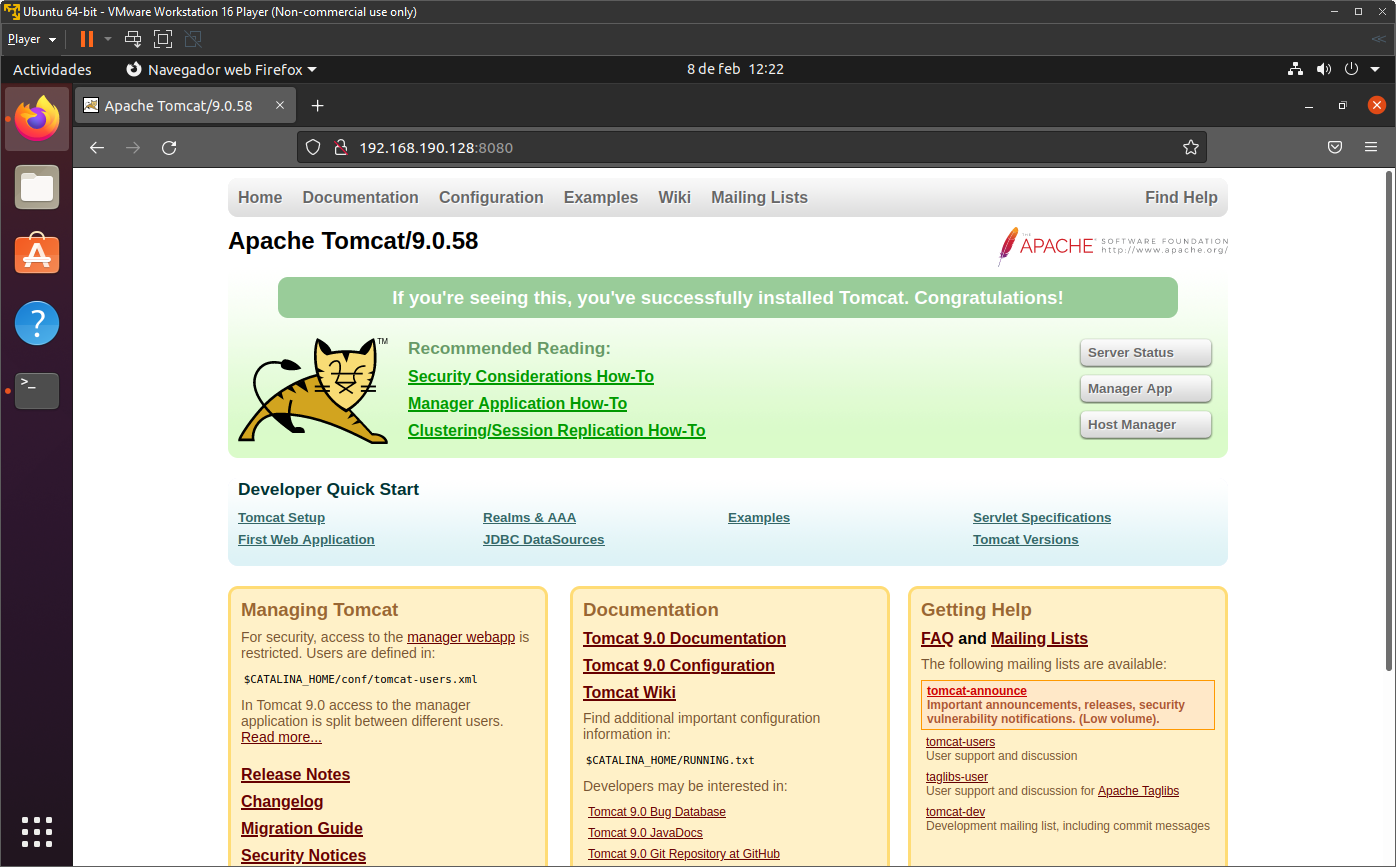
**sudo ufw allow 8080**



**sudo systemctl enable tomcat**



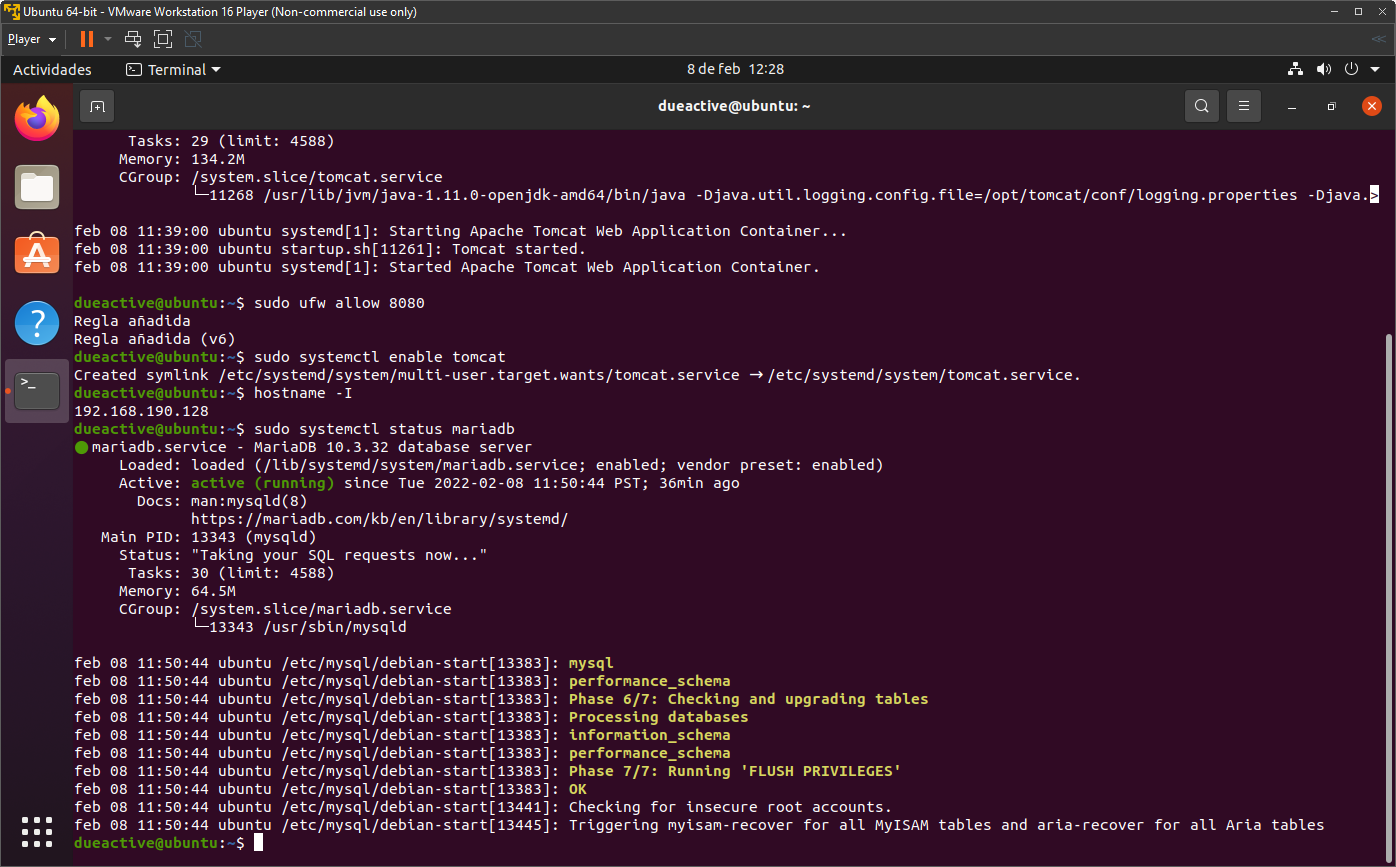
Finalmente, para ver que hemos expuesto correctamente el puerto 8008 y es accesible escribimos en el navegador la ip publica junto con el puerto que hemos asicado al servicio de Apache y como podemos ver se mostrará la web de Apache.



**MariaDb**

Para comprobar que MariaDb está funcionando correctamente podemos usar el comando

sudo systemctl status mariadb

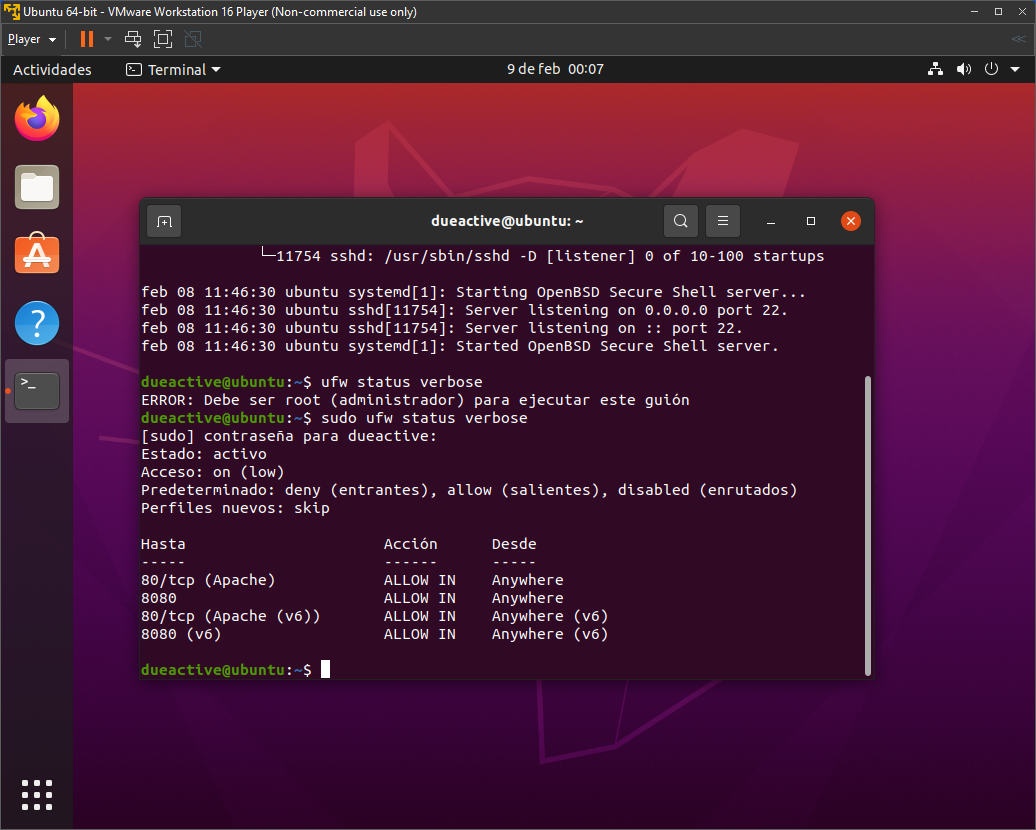


Para poder acceder a la base de datos usaremos un túnel ssh , para ello verificamos si tenemos habilitado el servicio ssh.

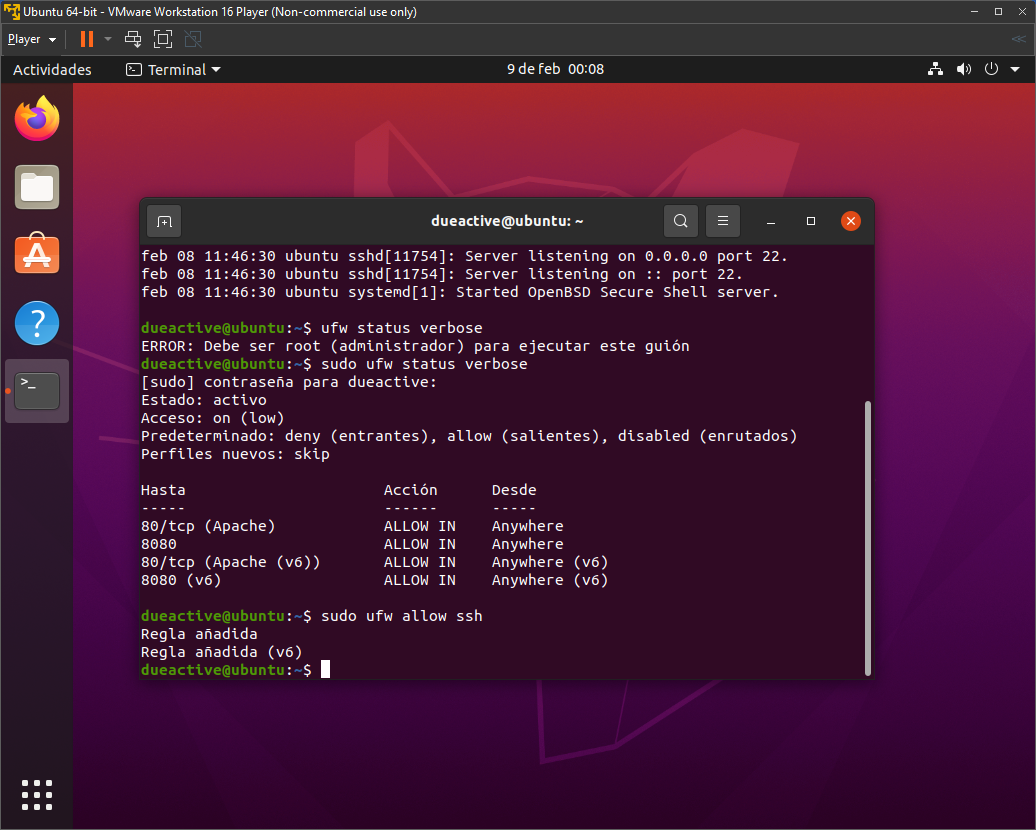
**systemctl status sshd**

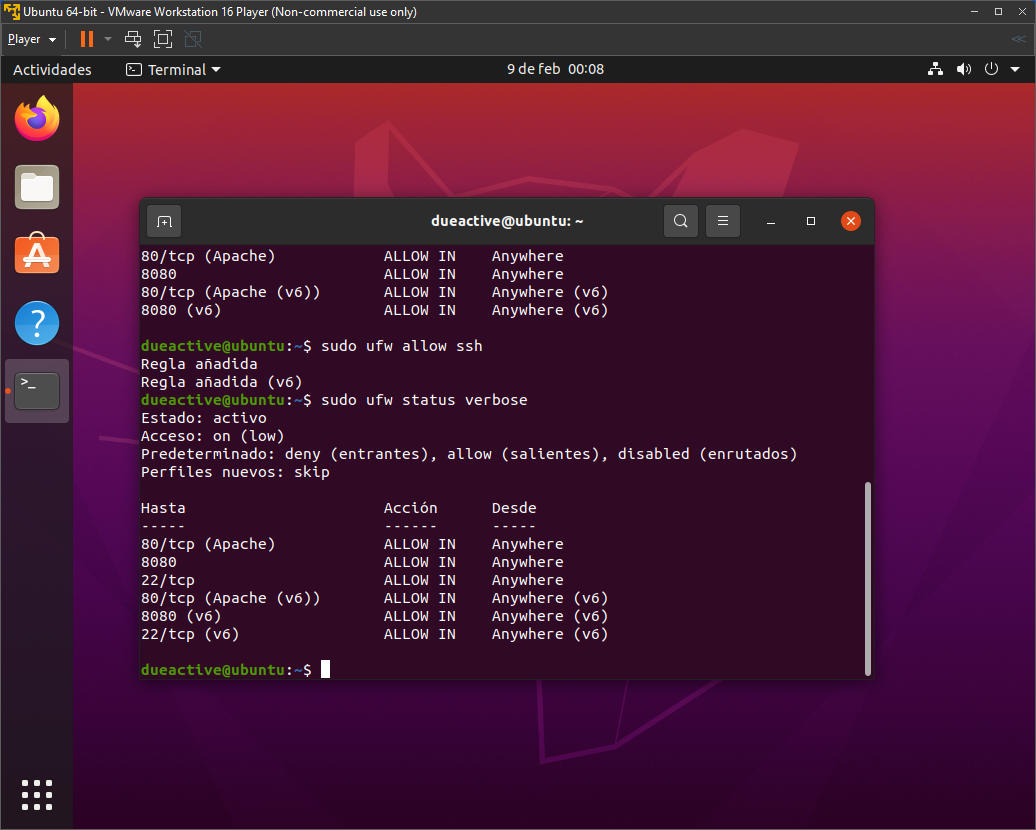
A continuación verificamos si tenemos habilitado la regla de firewall para el puerto 22 que es el puerto que usa ssh

*Sudo ufw status verbose*



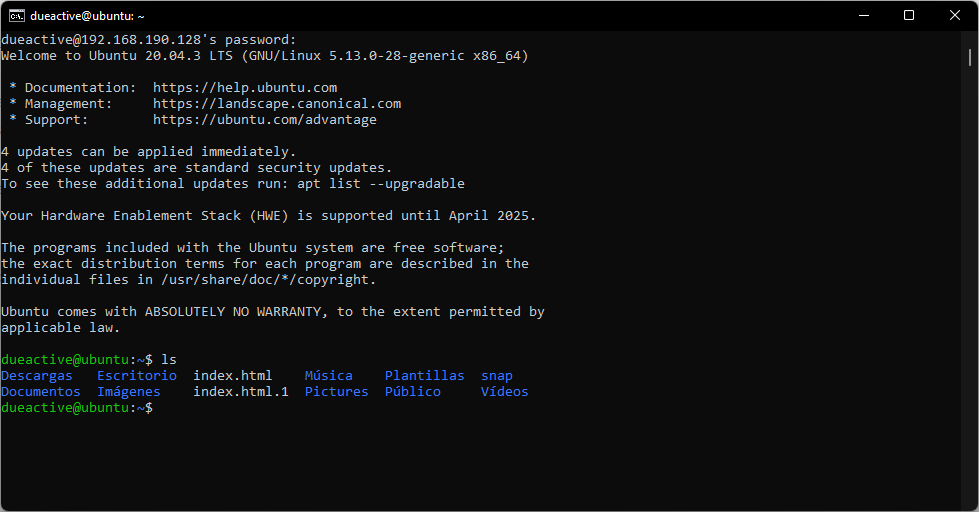
En este caso tenemos que incluir el puerto 22 en el firewall mediante el comando *sudo ufw allow ssh*





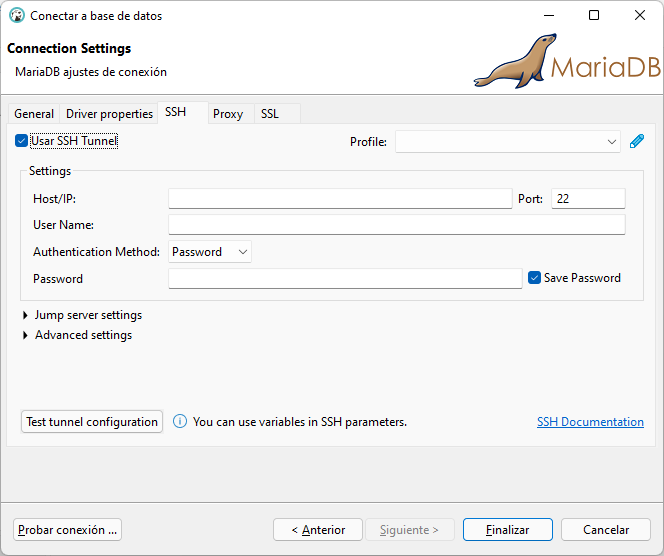
Ahora vemos que tenemos habilitado el puerto 22 y podemos acceder mediante ssh

Ssh usuario@ip en mi caso sería **ssh** [**dueactive@192.168.190.128**](mailto:dueactive@192.168.190.128) **, la ip la obtenemos con hostname -I**

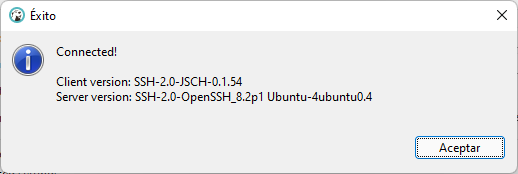


Ahora ya podemos conectarnos mediante ssh a la base de datos MariaDB.

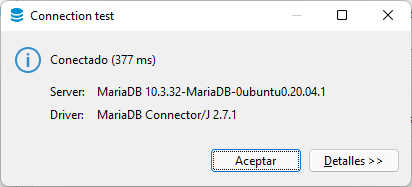
Mediante un gestor de base de datos, como DBeaver , MySQLWorkbench,…

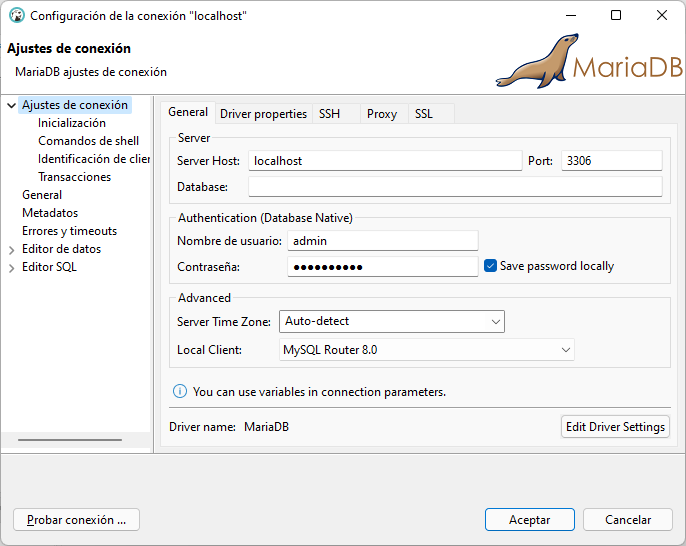


Probamos la conexión ssh



Probamos la conexión





Y vemos que tenemos acceso veríamos las base de datos si tuviéramos alguna creada.

