apt-get update && apt-get upgrade

4. Install Java

In order to run Tomcat server, Java has to be installed on the VPS. Execute the following command to install the Java Development Kit package (JDK)

apt-get install default-jdk

After the installation is completed, You can check the Java installed on your server using the command below

java -version

If you receive an output similar to the one above, the installation is successful.

openjdk version "1.8.0\_91"

OpenJDK Runtime Environment (build 1.8.0\_91-8u91-b14-0ubuntu4~16.04.1-b14)

OpenJDK 64-Bit Server VM (build 25.91-b14, mixed mode)

Once Java is installed on the server, we will proceed installing Tomcat 9.

5. Install and Configure Tomcat 9

For security reasons it is not recommended to run Tomcat as user root, so we will create a new system user

useradd -r tomcat9 --shell /bin/false

Change the current working directory to /opt and download the latest Tomcat 9 release to the /opt directory on your server. At the moment of writing this tutorial Tomcat version 9.0.0.M6 is the latest one. You should go to Tomcat’s official website and check if a newer version is available.

cd /opt

wget http://mirror.symnds.com/software/Apache/tomcat/tomcat-9/v9.0.0.M9/bin/apache-tomcat-9.0.0.M9.tar.gz

Extract the content of the ‘apache-tomcat-9.0.0.M6.tar.gz’ tarball archive

tar -zxf apache-tomcat-9.0.0.M6.tar.gz

Create a symbolic link of the Tomcat directory to /opt/tomcat-latest and set the appropriate ownership:

ln -s apache-tomcat-9.0.0.M6 tomcat-latest

chown -hR tomcat9: tomcat-latest apache-tomcat-9.0.0.M6

Tomcat can be started, stopped and restarted using the bash scripts located in the /opt/tomcat-latest/bin directory or even better, we will create a systemd init file for that purpose

nano /etc/systemd/system/tomcat.service

and add the following content to the file

[Unit]

Description=Tomcat9

After=network.target

[Service]

Type=forking

User=tomcat9

Group=tomcat9

Environment=CATALINA\_PID=/opt/tomcat-latest/tomcat9.pid

Environment=JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64

Environment=CATALINA\_HOME=/opt/tomcat-latest

Environment=CATALINA\_BASE=/opt/tomcat-latest

Environment="CATALINA\_OPTS=-Xms512m -Xmx512m"

Environment="JAVA\_OPTS=-Dfile.encoding=UTF-8 -Dnet.sf.ehcache.skipUpdateCheck=true -XX:+UseConcMarkSweepGC -XX:+CMSClassUnloadingEnabled -XX:+UseParNewGC"

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

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

[Install]

WantedBy=multi-user.target

Save the file, reload the systemd daemon, start the Tomcat server and enable it to start on boot

systemctl daemon-reload

systemctl start tomcat

systemctl enable tomcat

Finally, if you closely followed the instructions in this tutorial you will be able to access Tomcat by navigating your favorite web browser to <http://ipDelServidor:8080>

If you followed that installation tutorial, those file can be found at:

/opt/tomcat/webapps/manager/META-INF/context.xml

sudo nano /opt/tomcat/webapps/host-manager/META-INF/context.xml

To allow remote access, edit the files and comment out the part of the configuration file specifying the IP address restriction or replace them with a list of IP addresses that you would like to whitelist.

context.xml files for Tomcat webapps

<Context antiResourceLocking="false" privileged="true" >

<!--<Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />-->

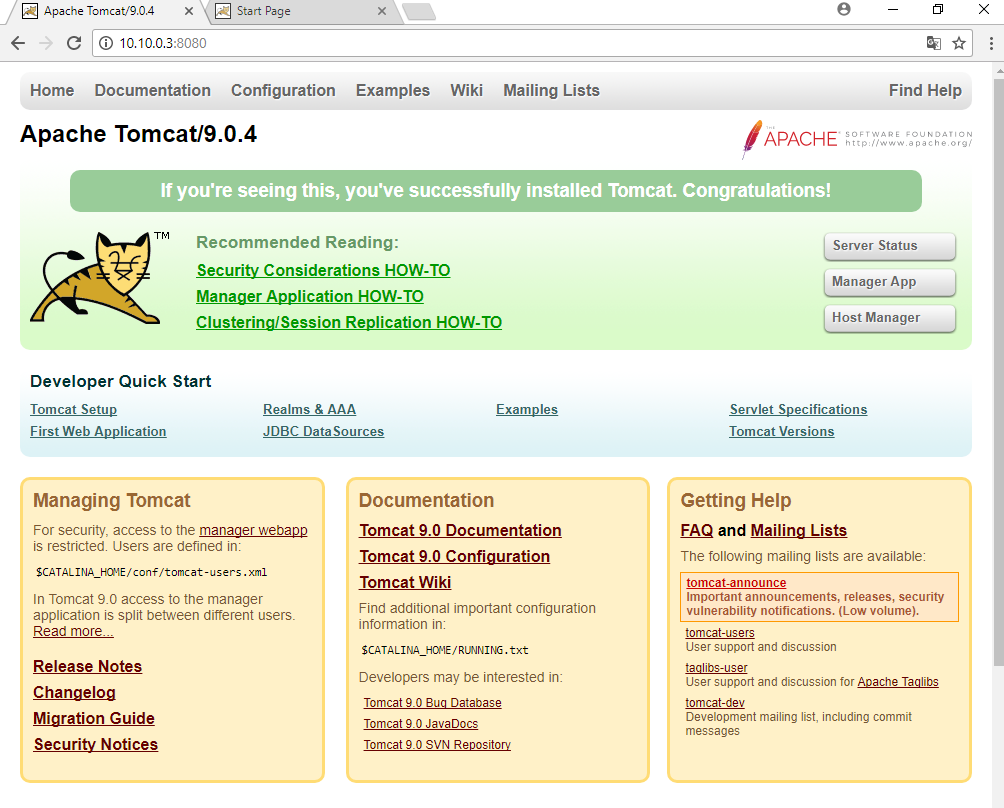
</Context>

Restart Tomcat for the changes to take effect:

sudo systemctl restart tomcat

entramos en /opt/tomcat/conf/tomcat-users.xml y añadimos la siguiente linea





Manager app



En desplegar archive war, elegimos la aplicacion que queramos subir y clicamos en desplegar.

