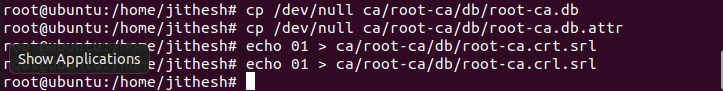
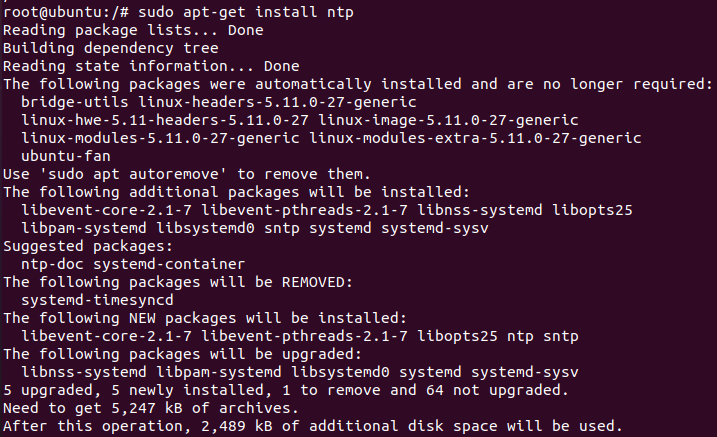
**Public Key Infrastructure**

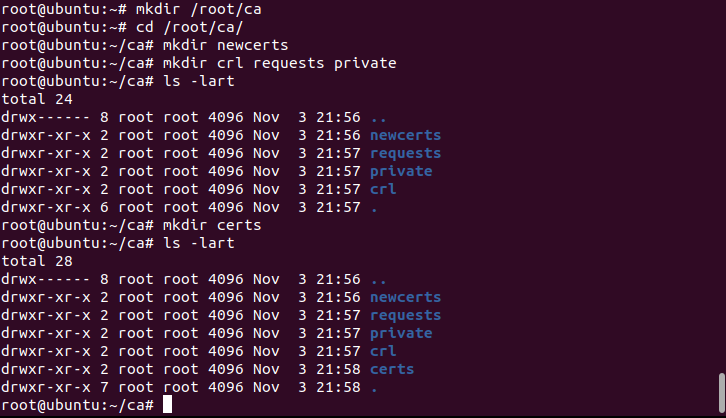
Creating Directories for certificate Installation



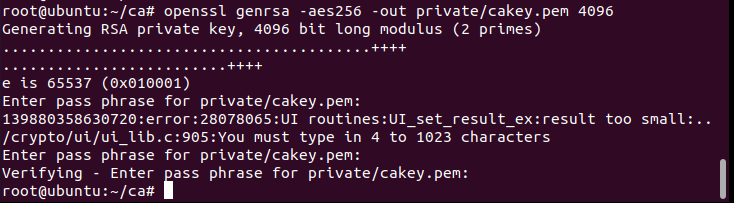
Create Database



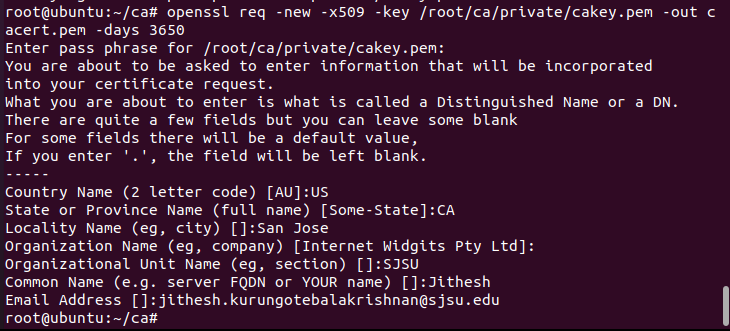




Generating private key (cakey.pem) File and set the Pass phrase



Create root certificate with private key (cakey.pem) as an input. Root certificate is cacert.pem

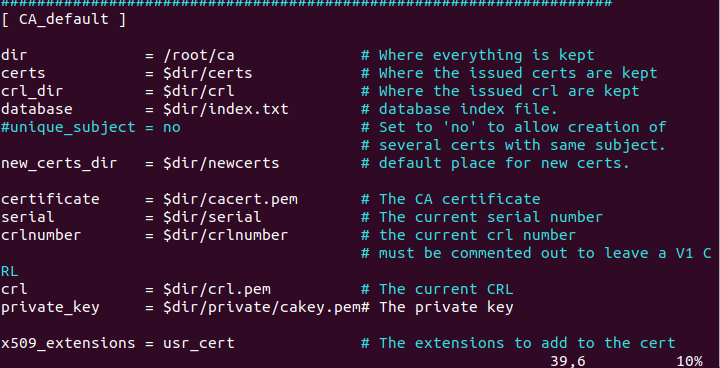


Change the permission to 600 recursively for the files and folders present in /root/ca directory.

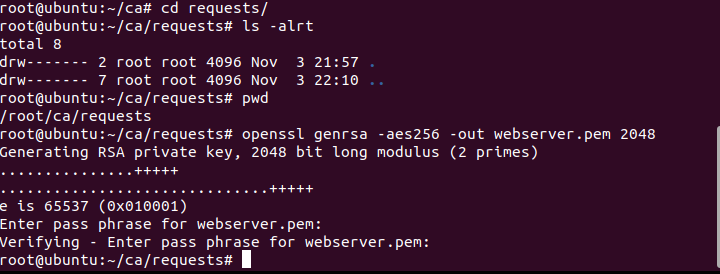


Edit the openssl file to change the directory i.e. dir=/root/ca

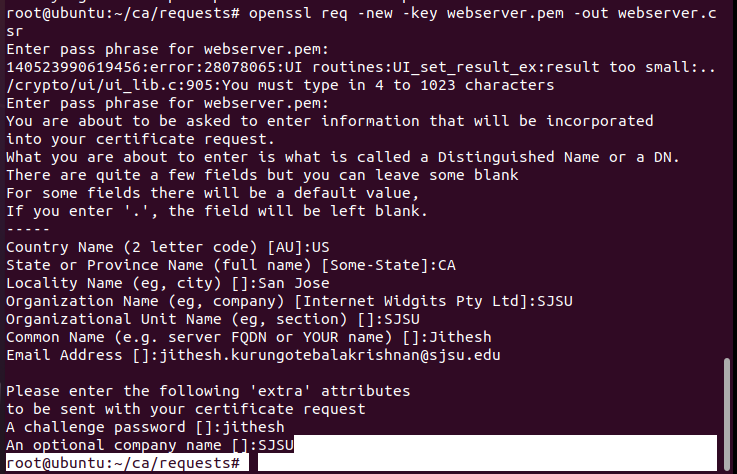




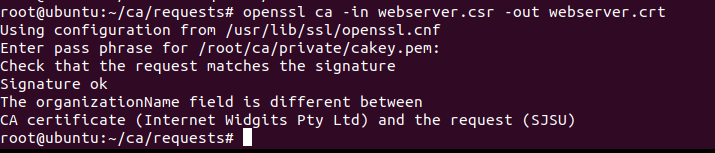
Signing the certificate by initiating the request by traversing to requests folder.

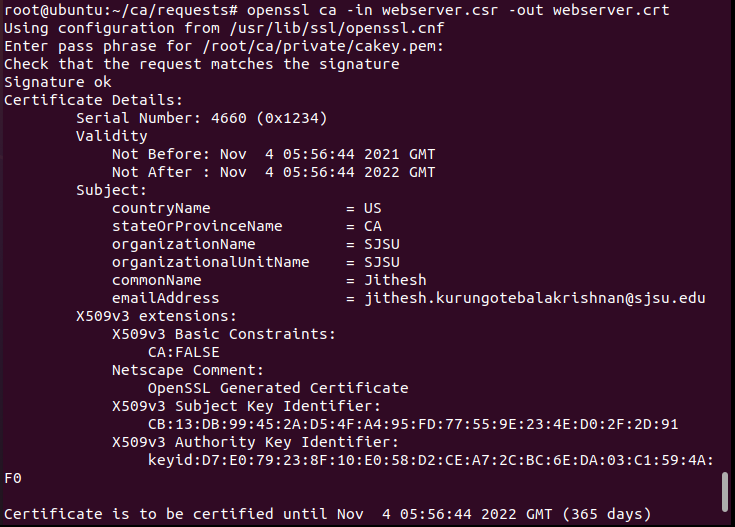


Create certificate request and save it in csr format.

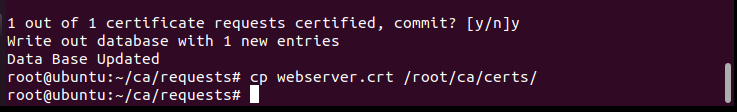


Signing the request certificate and enter the passphrase

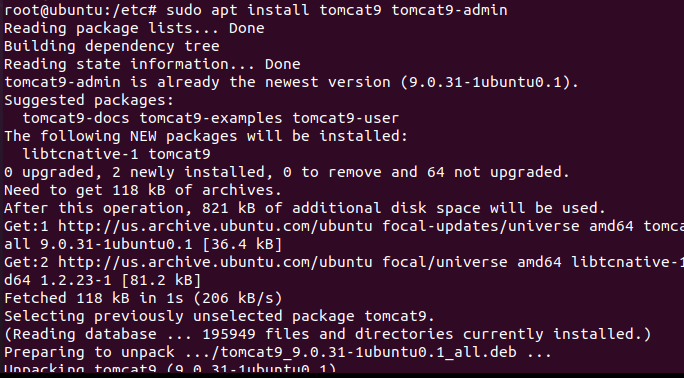




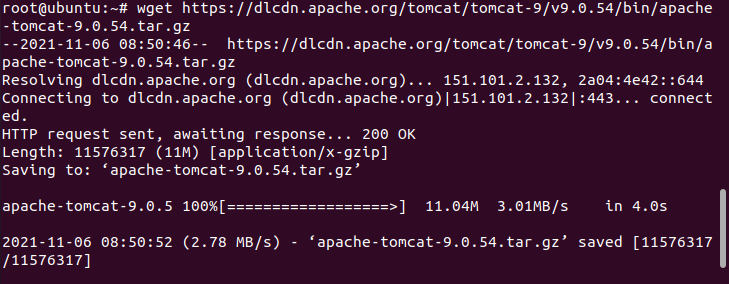
Move the certificate to /root/ca/certs folder



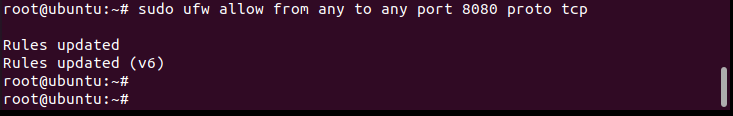
Installing tomcat



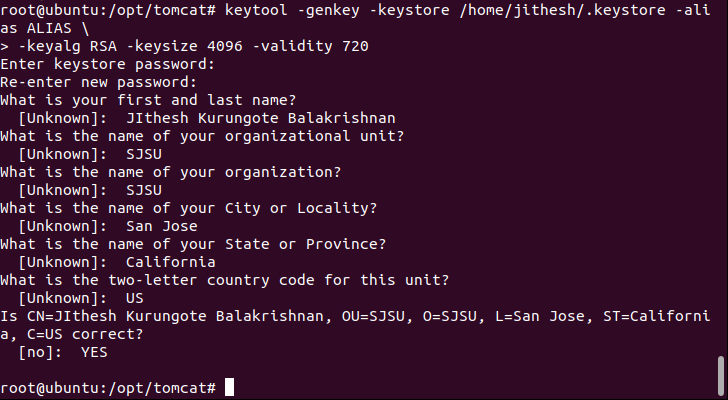
Downloading the tomcat packages for running the services on the server



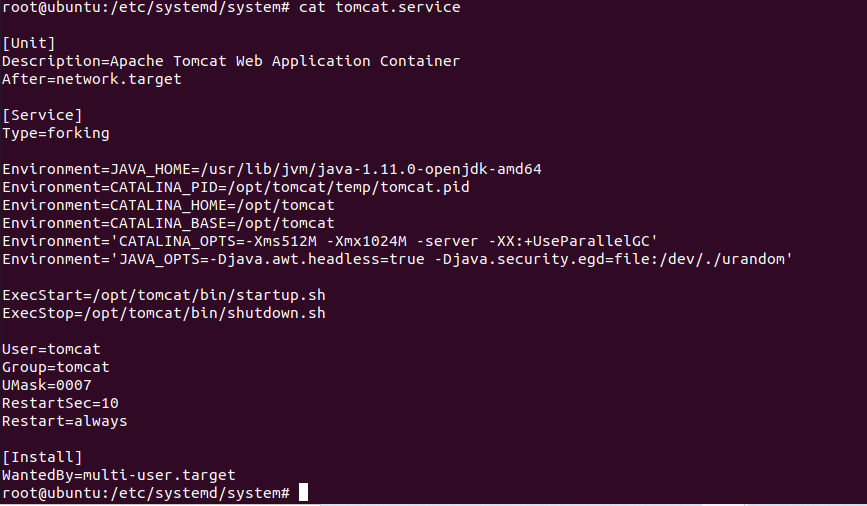
Allowing other ports to be enabled for tomcat service to run



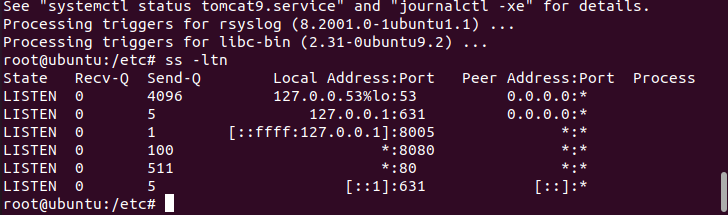
Generating the keystore file



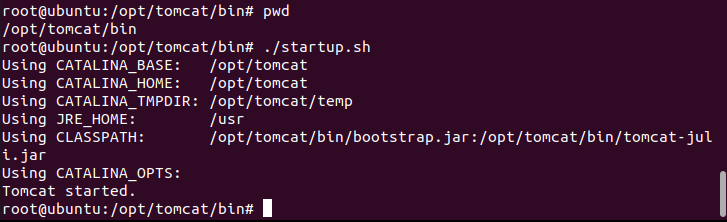
Tomcat service script configured on server by configuring the Java environment

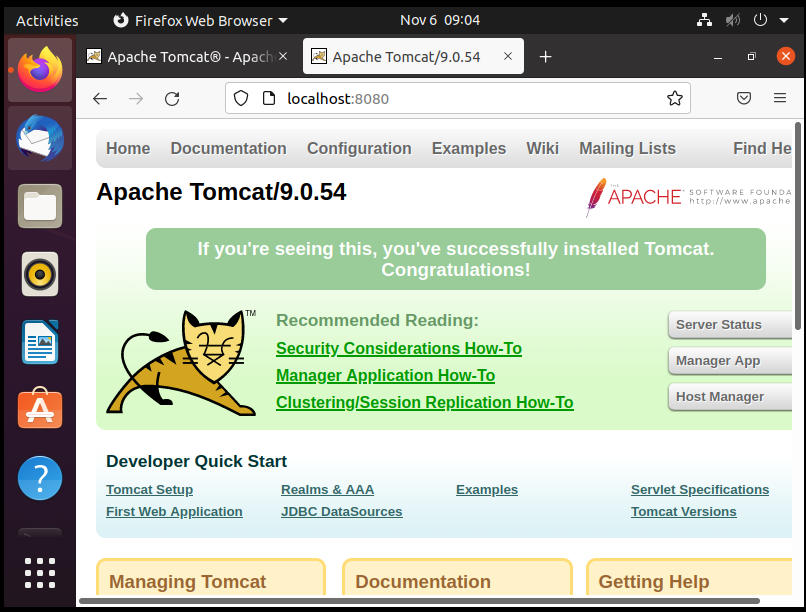


Verifying if port 8080 is enabled for tomcat service using ss-ltn command

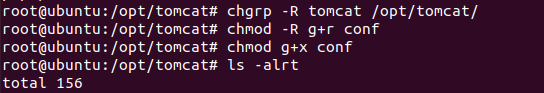


Running the tomcat service using startup script by traversing to /opt/tomcat/bin folder

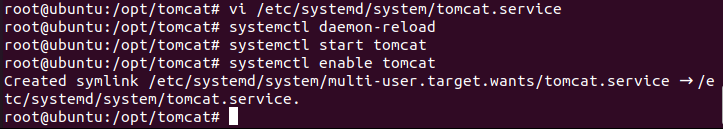


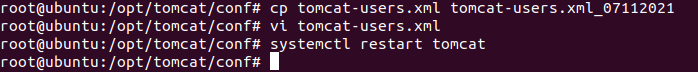
Apache tomcat webpage is displayed on localhost 8080

Changing the group and permission folders and files and present in /opt/tomcat file

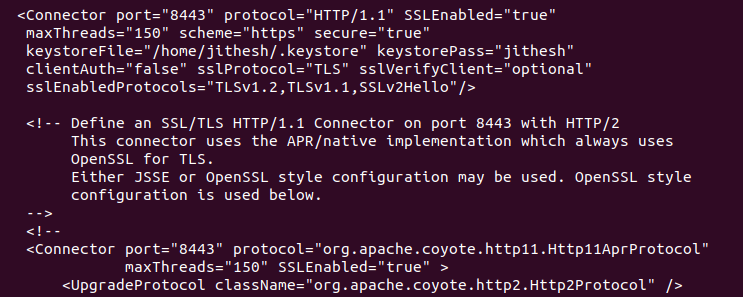








Enable SSL and configure the keystore file for tomcat to run on 8443 port via https



Tomcat webpage is being displayed after necessary configuration change using https with ssl certificate being enabled over the port 8443.

