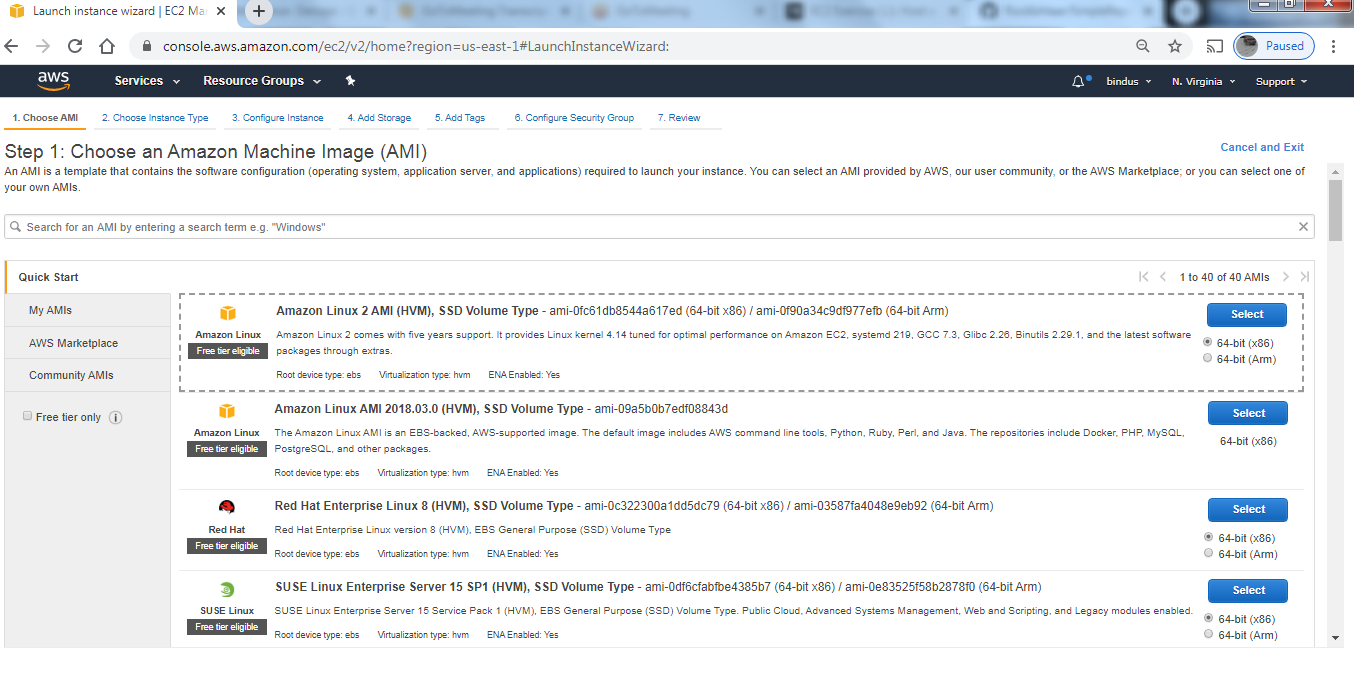
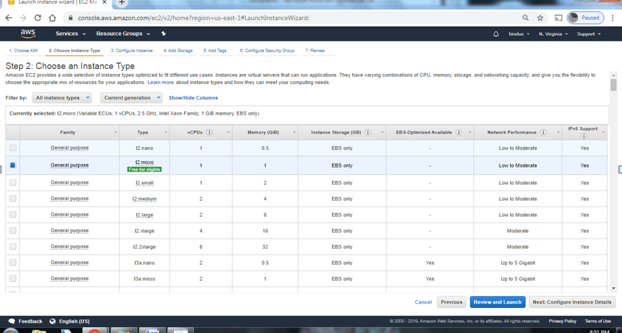
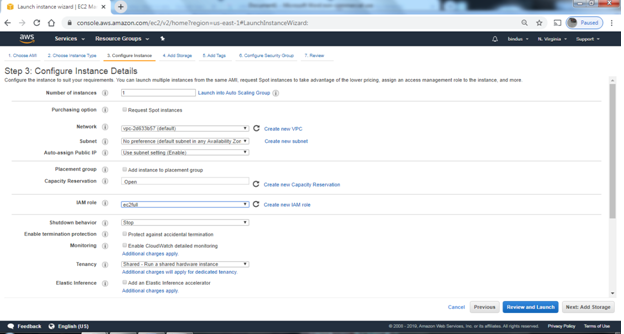
***"Build a Jenkins Pipeline (Pipeline Job) that Builds a sample Java Application and hosts it in Multiple EC2 Apache Tomcat Servers.***

Step 1: Launching EC2

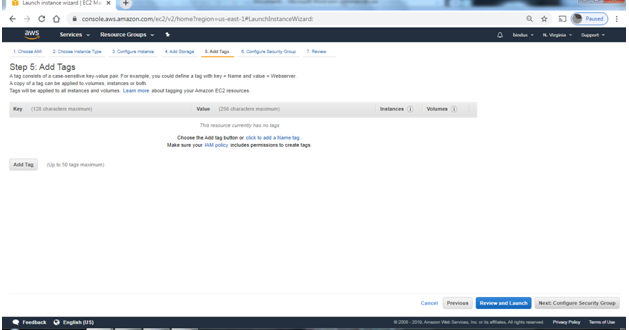
1. Sign in to AWS and click on Services > Compute > EC2 > Instances > Launch Instance and following window should pop up..



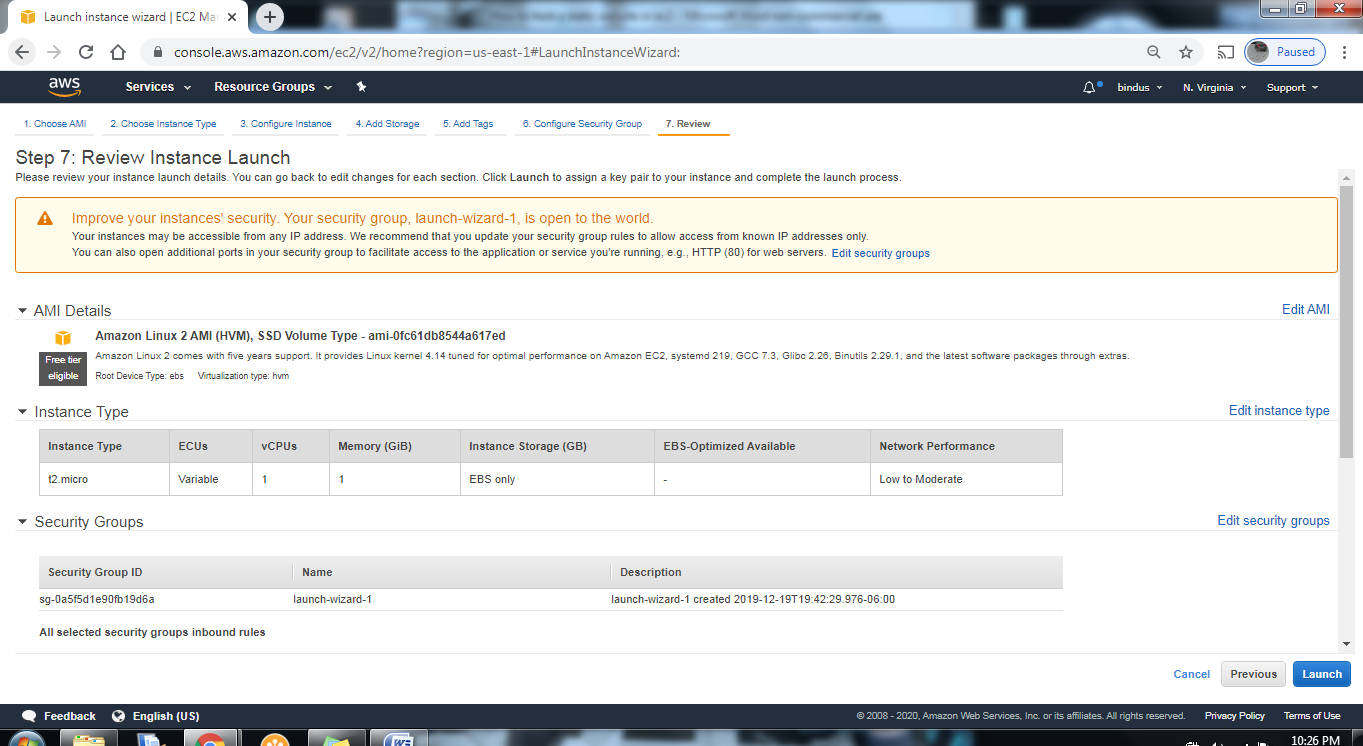


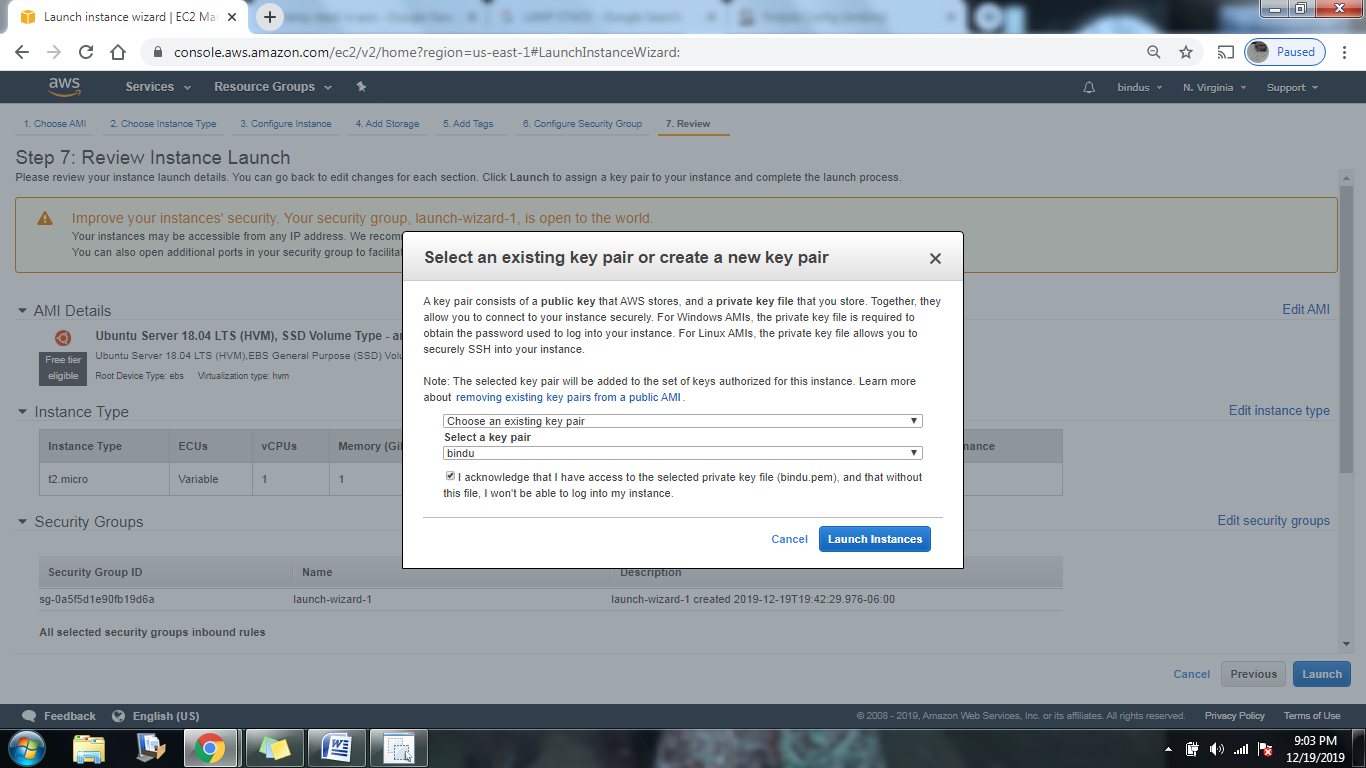


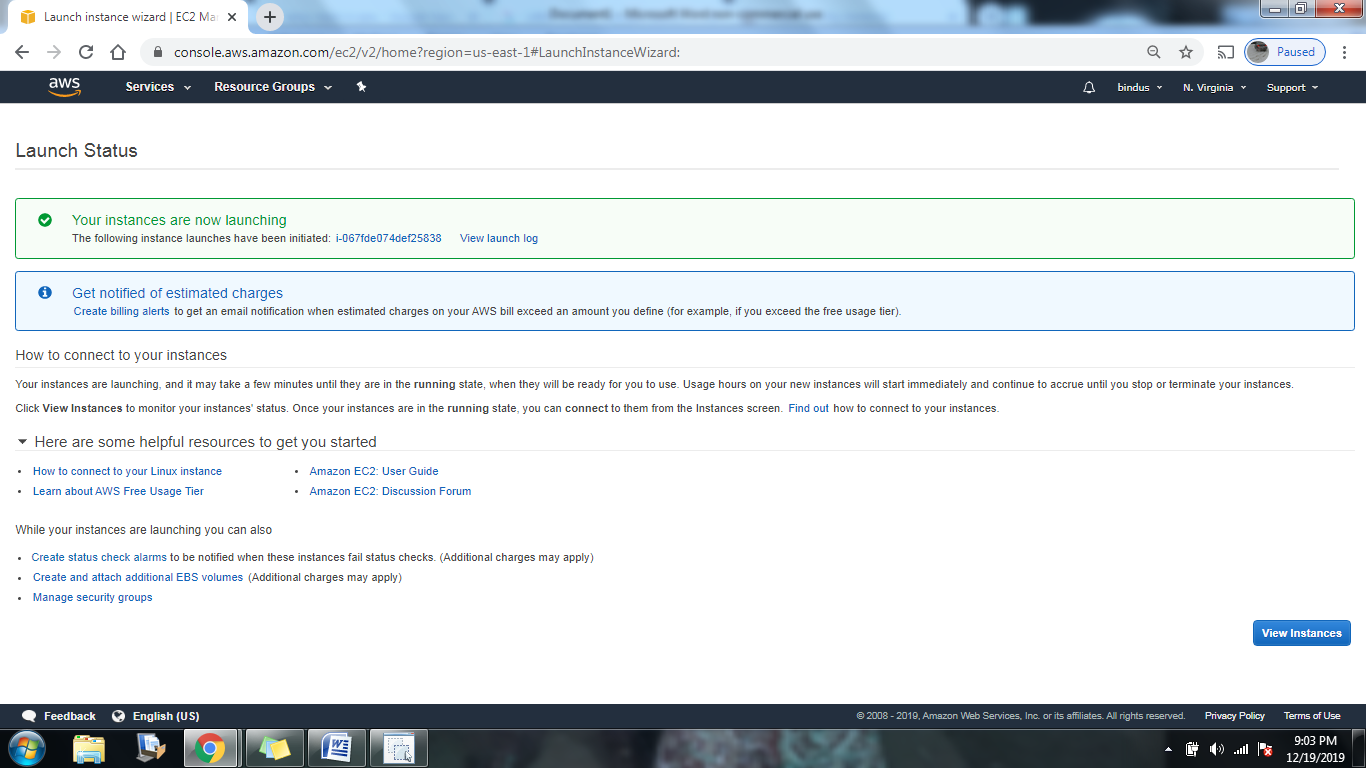


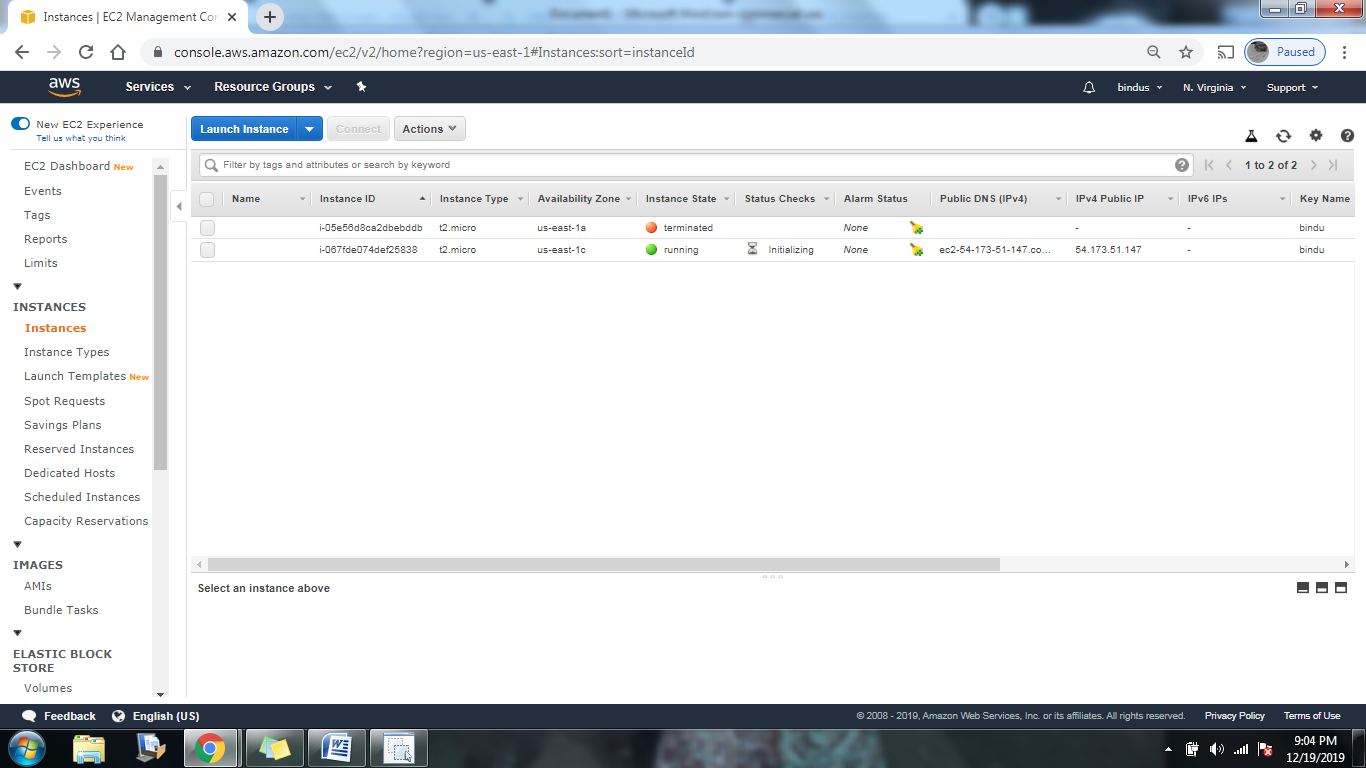


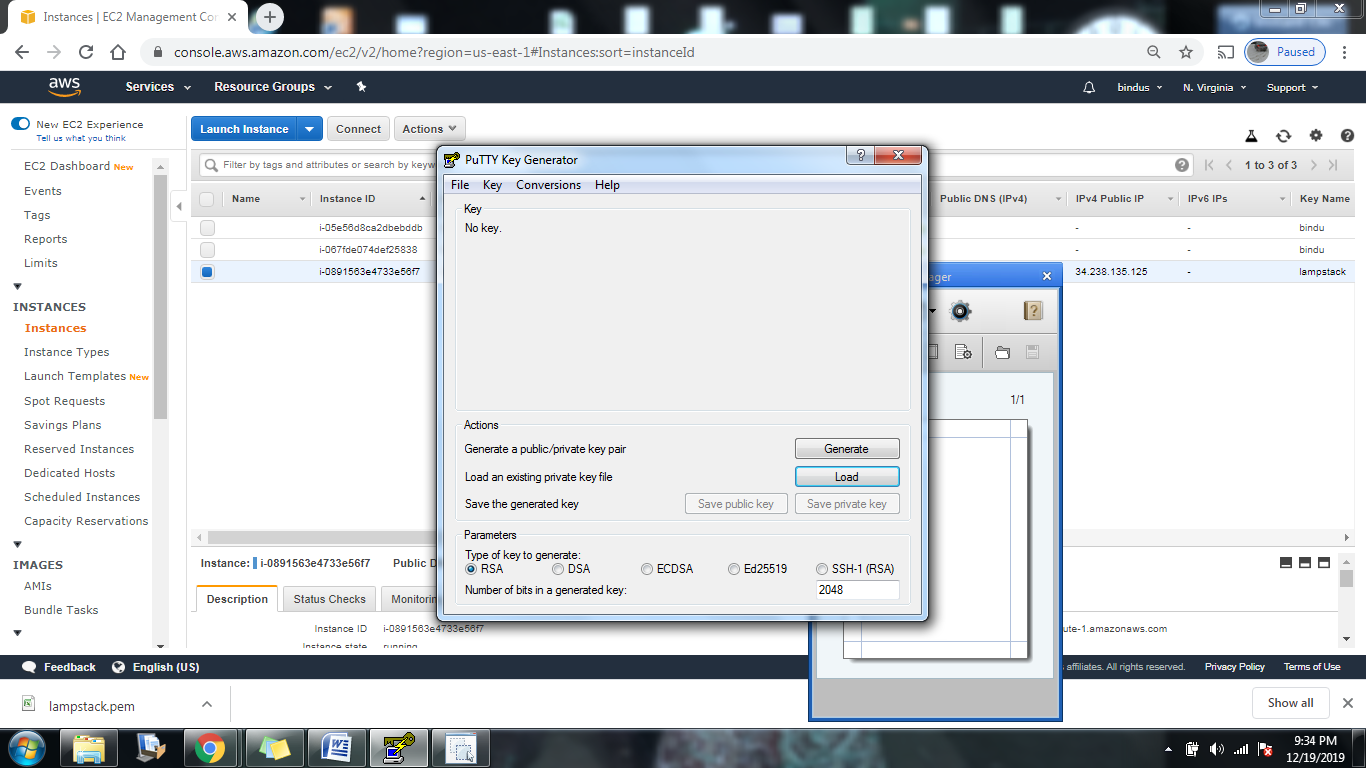


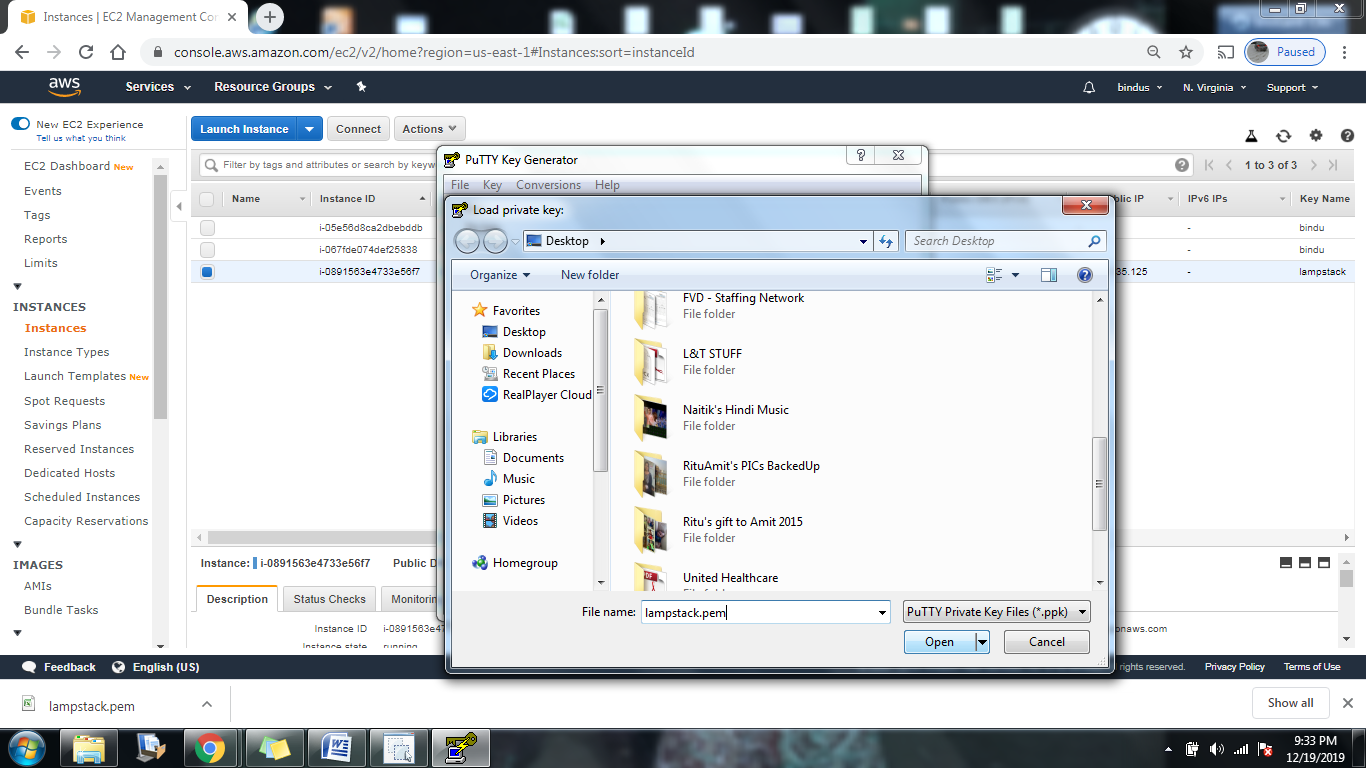


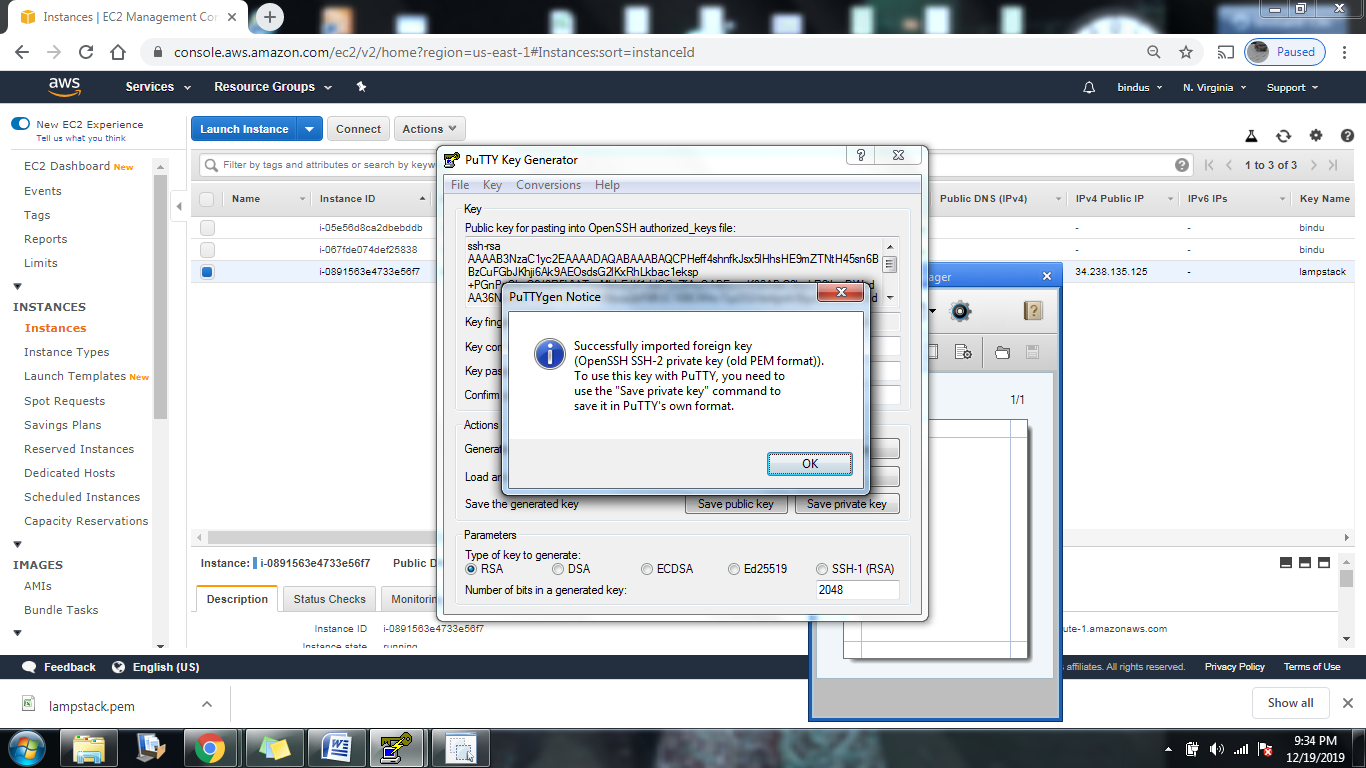


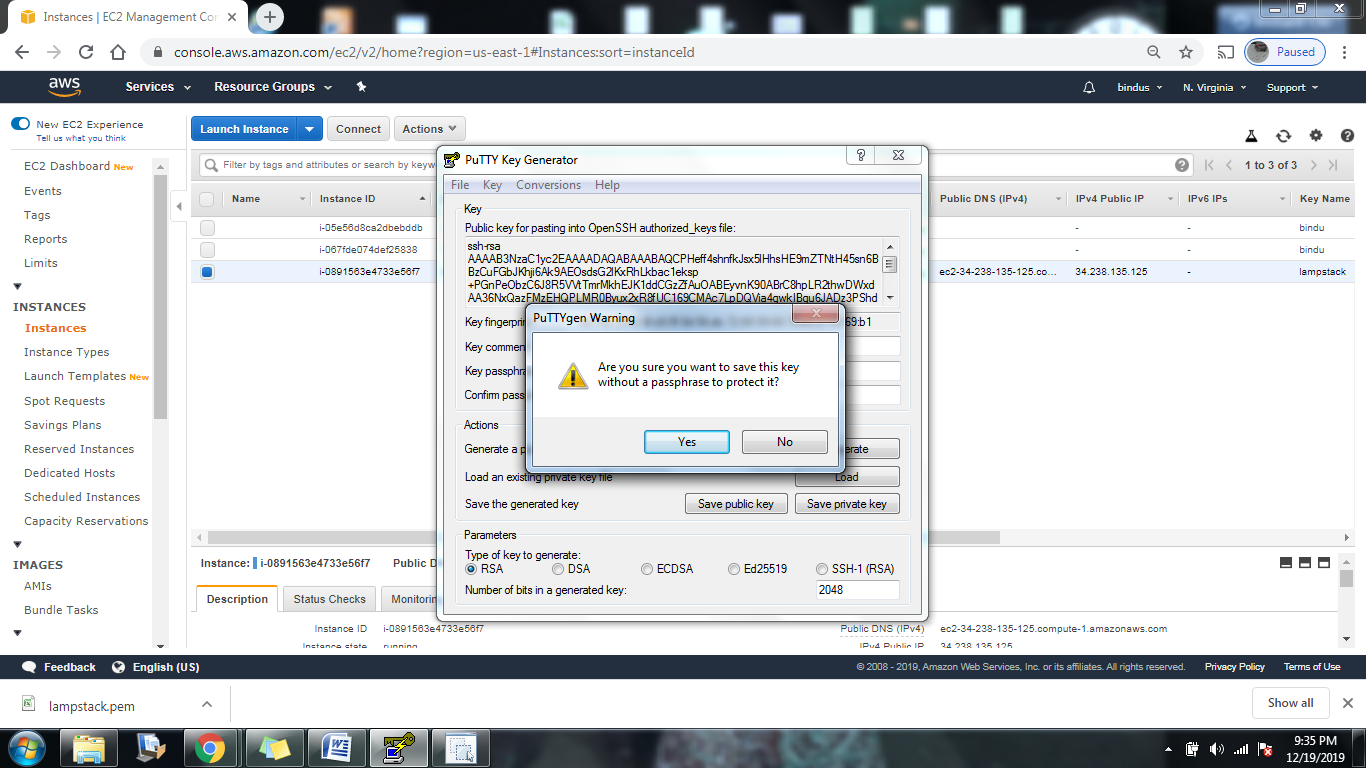


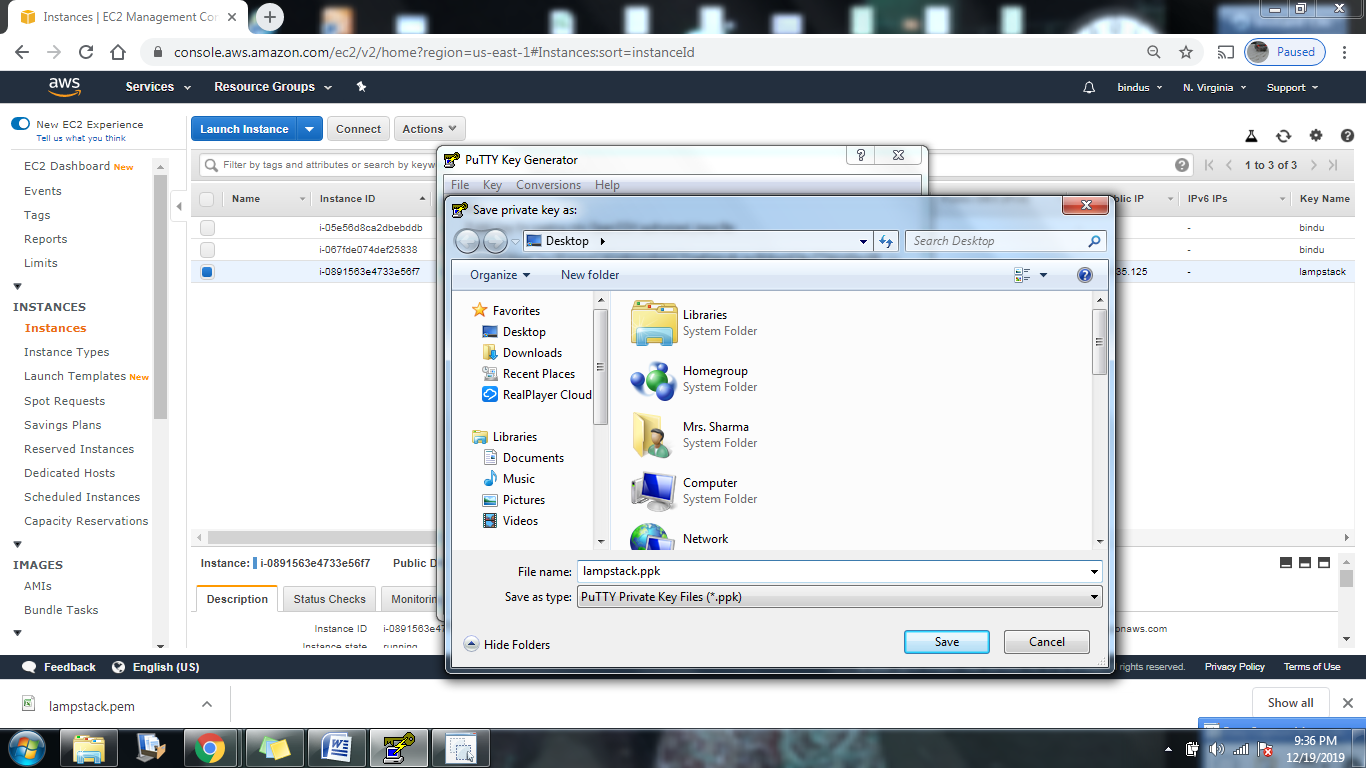


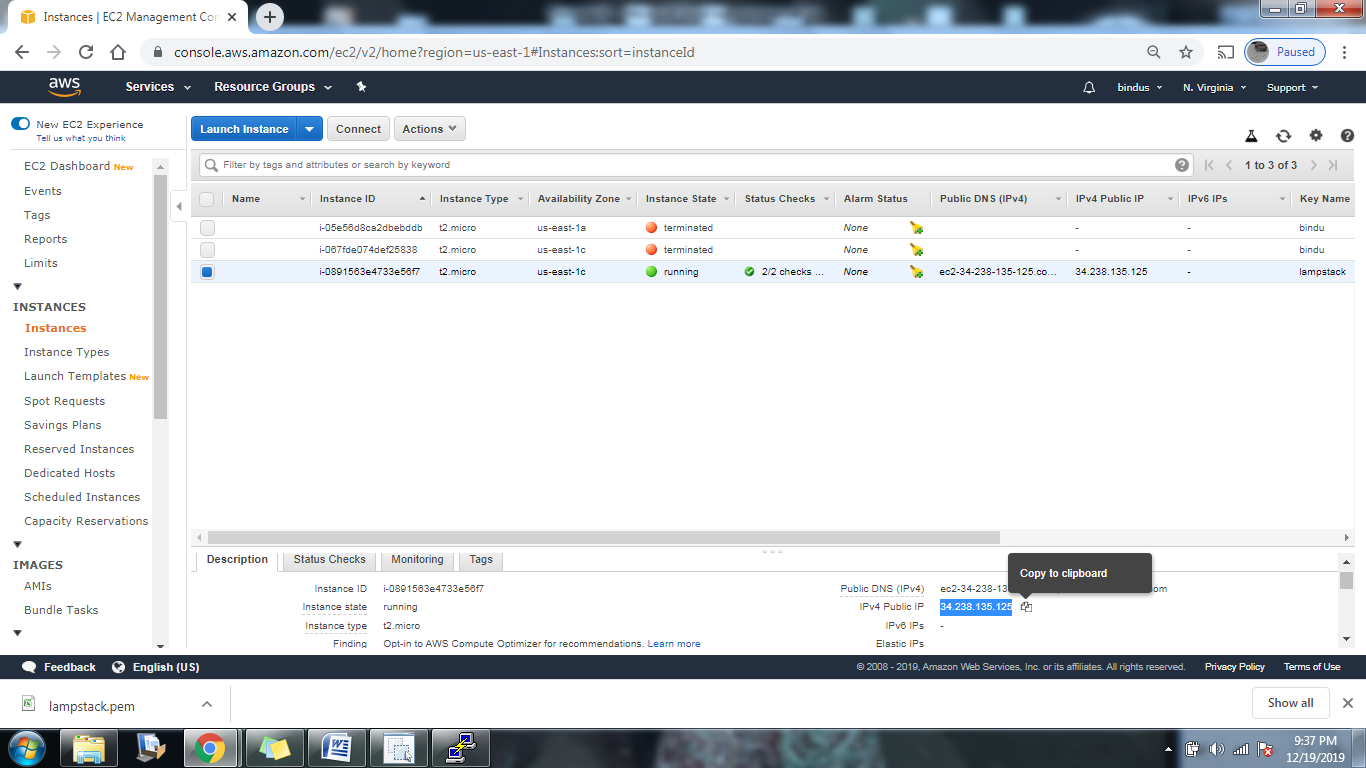


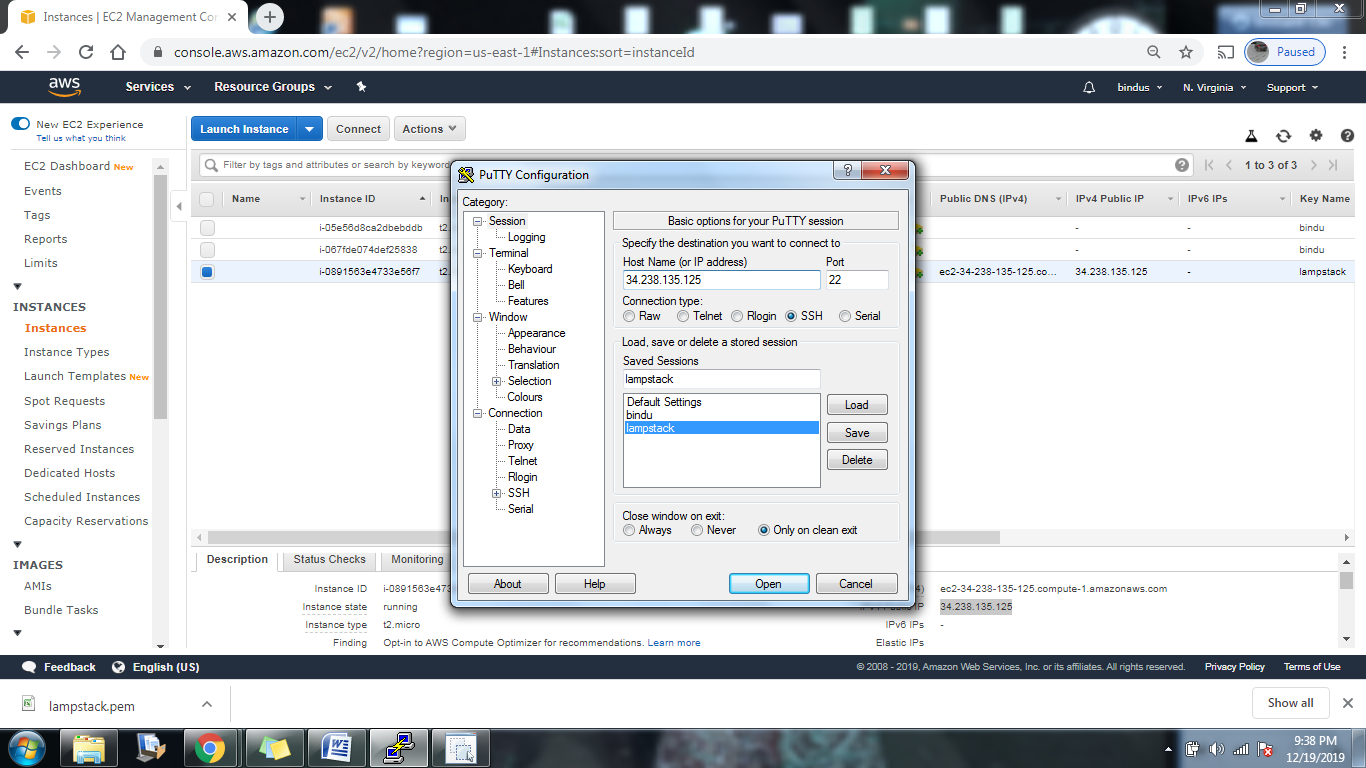


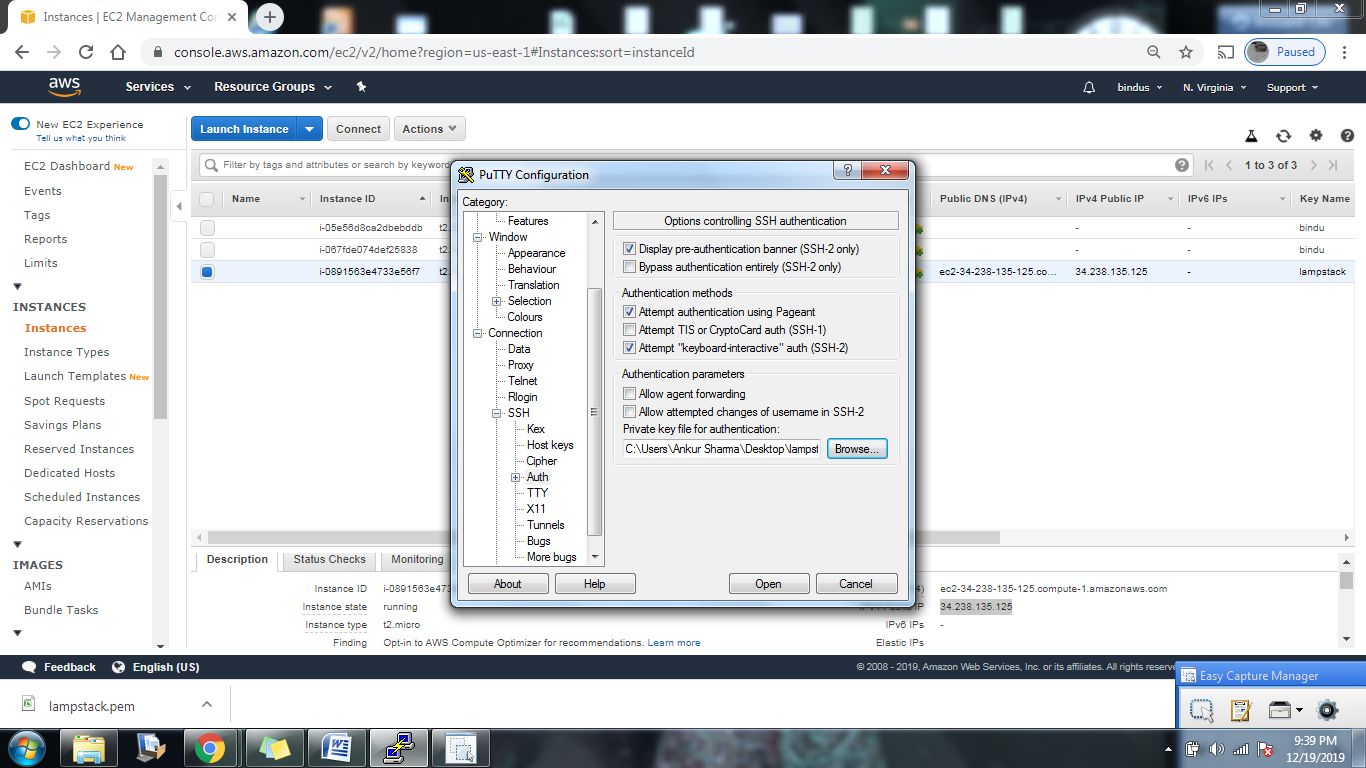


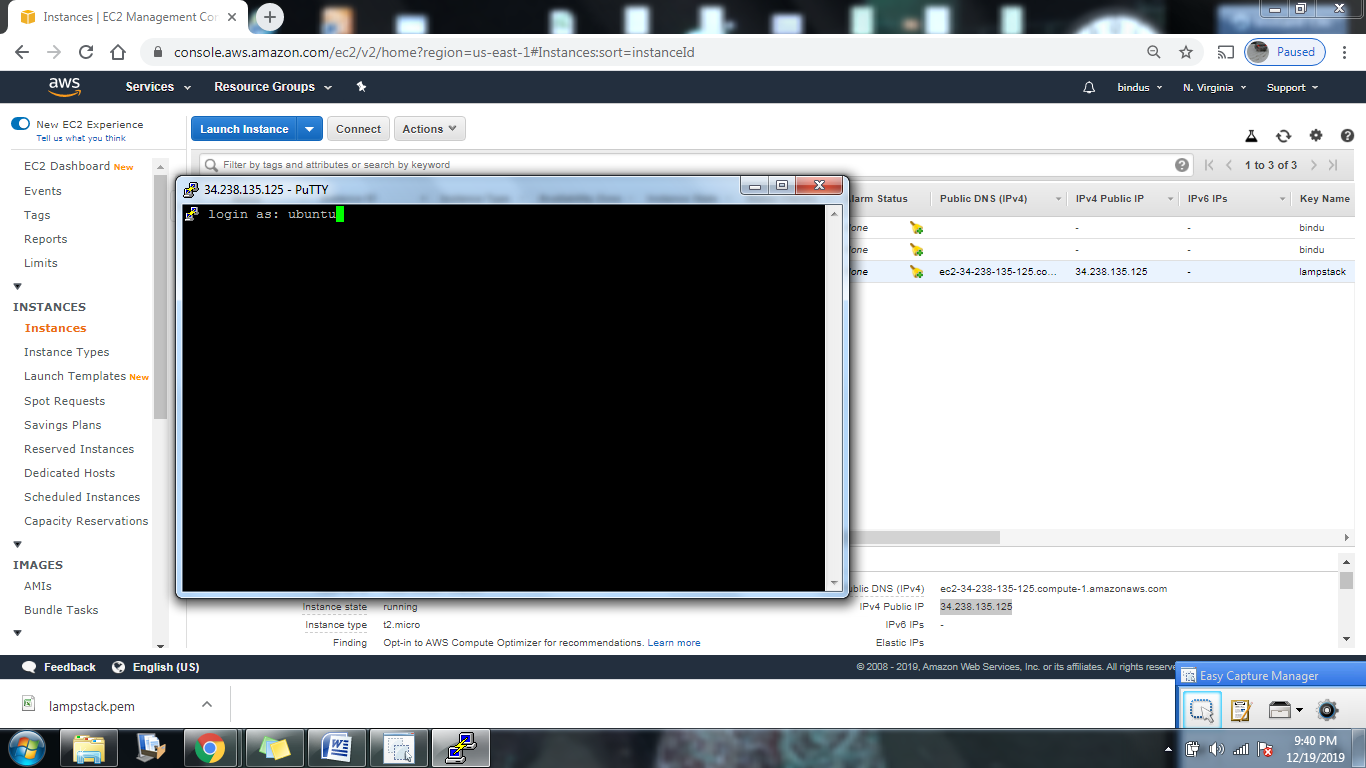




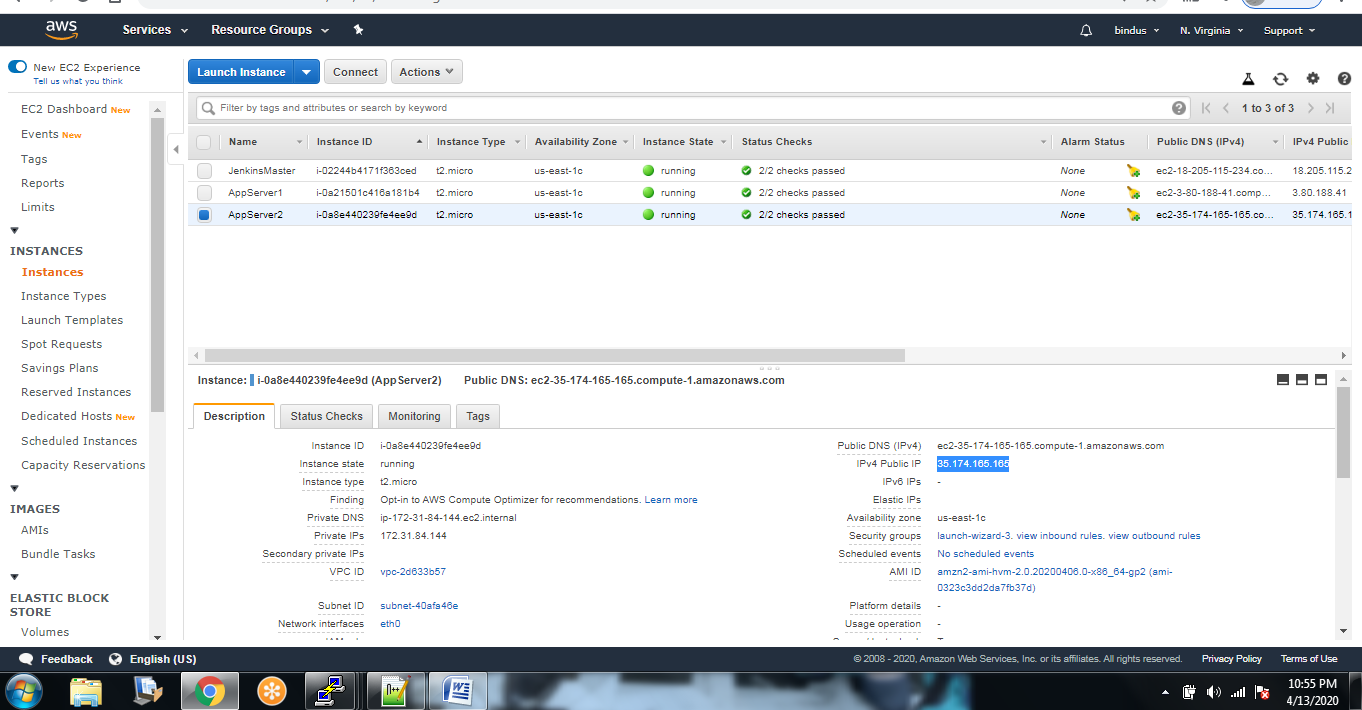








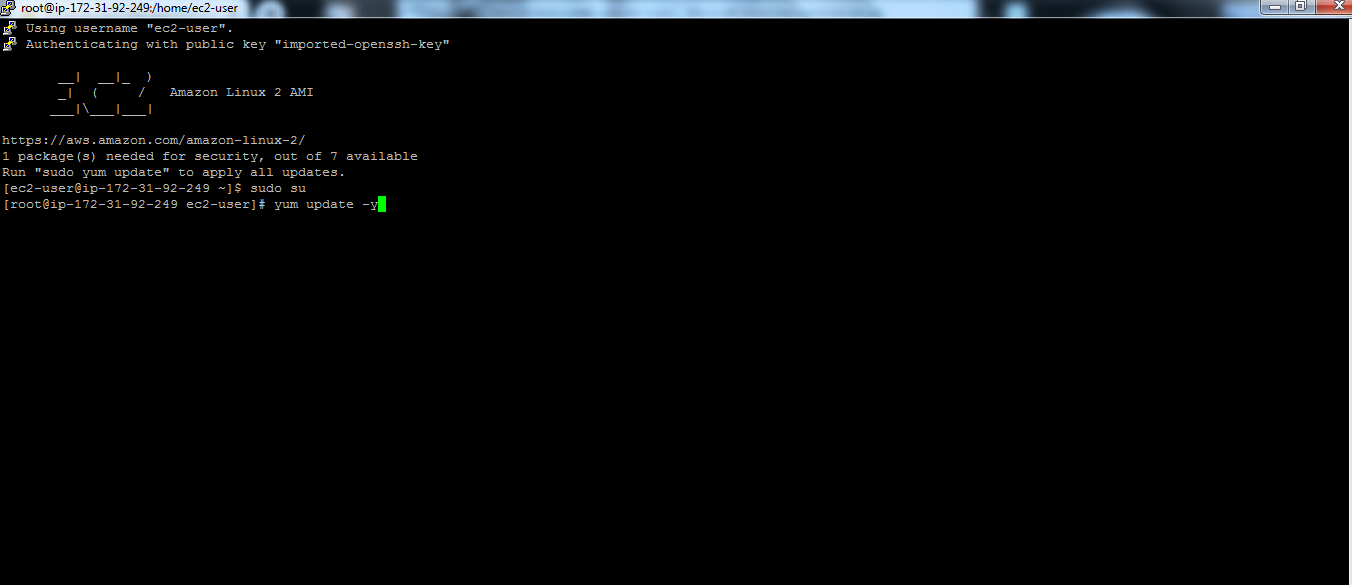
In the same way we’ll create three EC2 instances and name them accordingly.



We need to update the Linux repository.. To check that we’ll run command

Sudo su

Sudo yum update –y



Next step is to install Apatche Tomcat. First we need to change directory to opt.



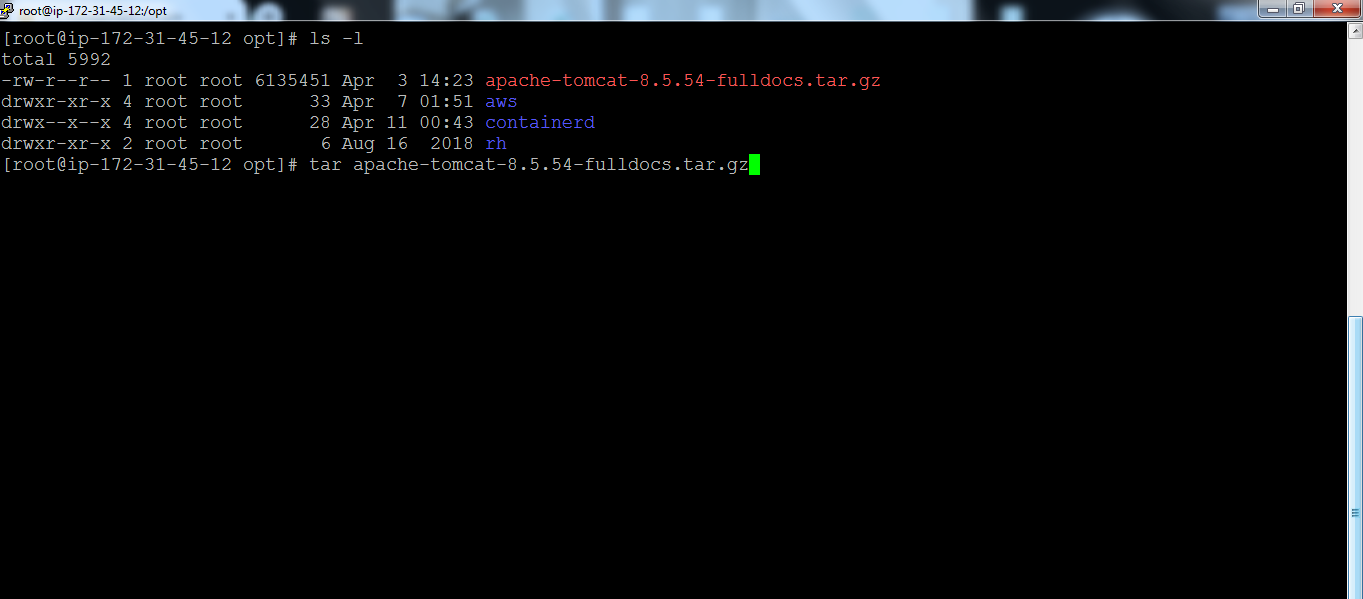
Command to download apatche tomcat:

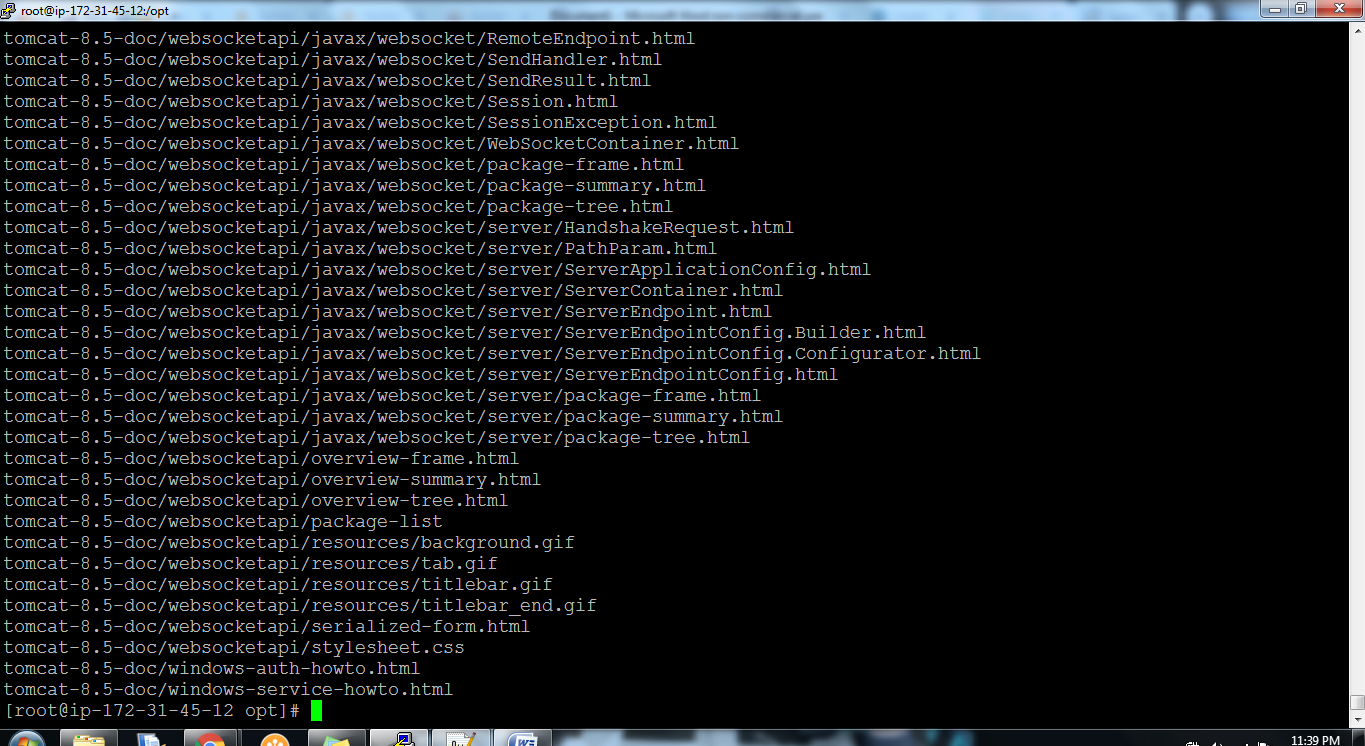
Wget <https://downloads.apache.org/tomcat/tomcat-9/v9.0.34/bin/apache-tomcat-9.0.34.tar.gz>.

Ls -l

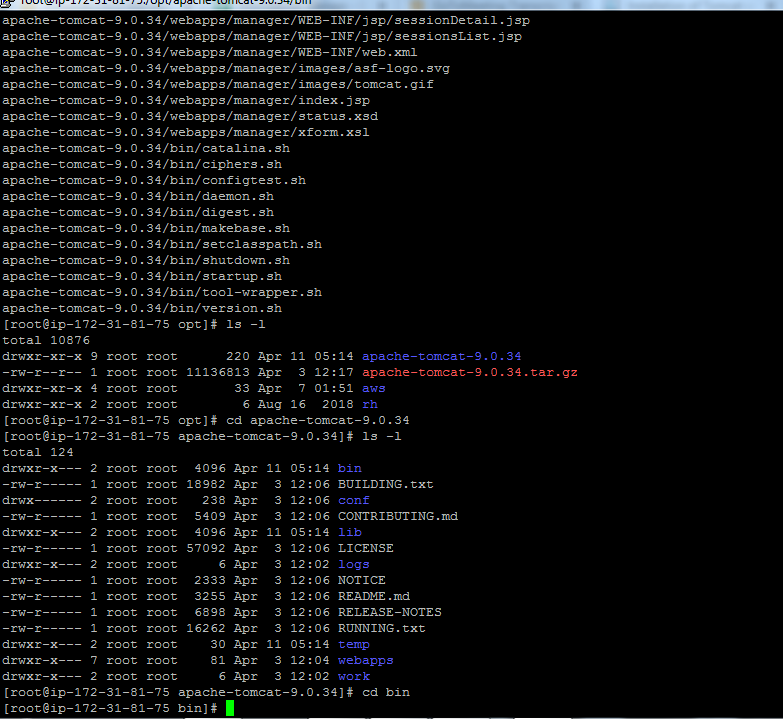


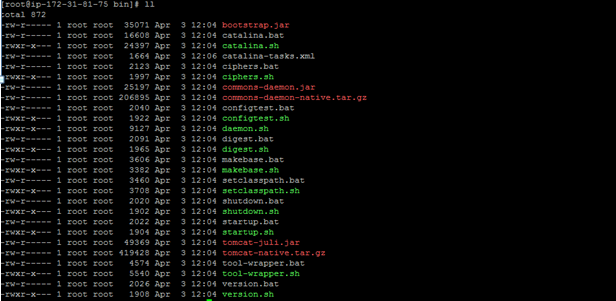
tar –zvxf [apache-tomcat-9.0.34.tar.gz](https://downloads.apache.org/tomcat/tomcat-9/v9.0.34/bin/apache-tomcat-9.0.34.tar.gz)





Then change directory to bin by using command cd apache-tomcat-9.0.34>ls –l>cd bin> ls -l

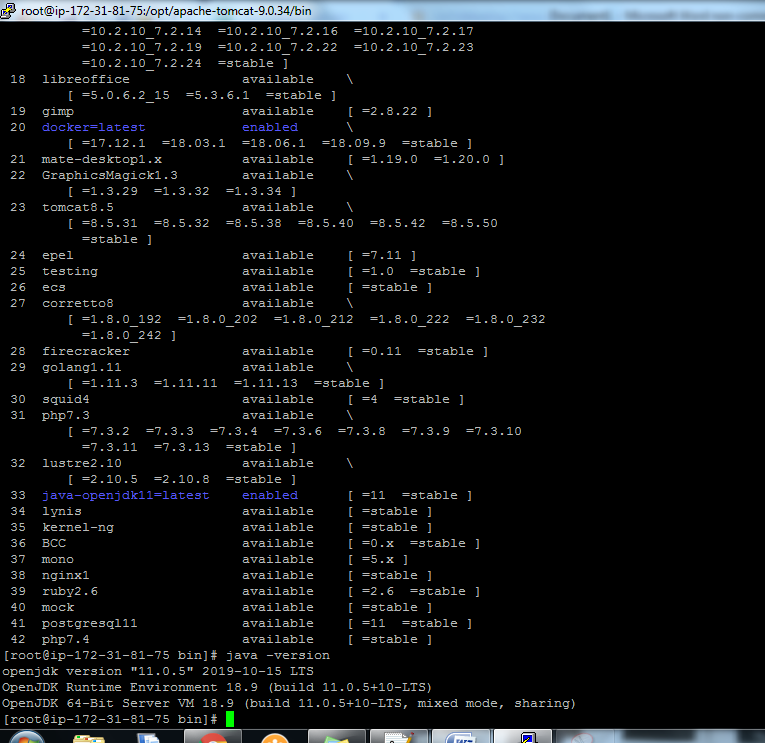




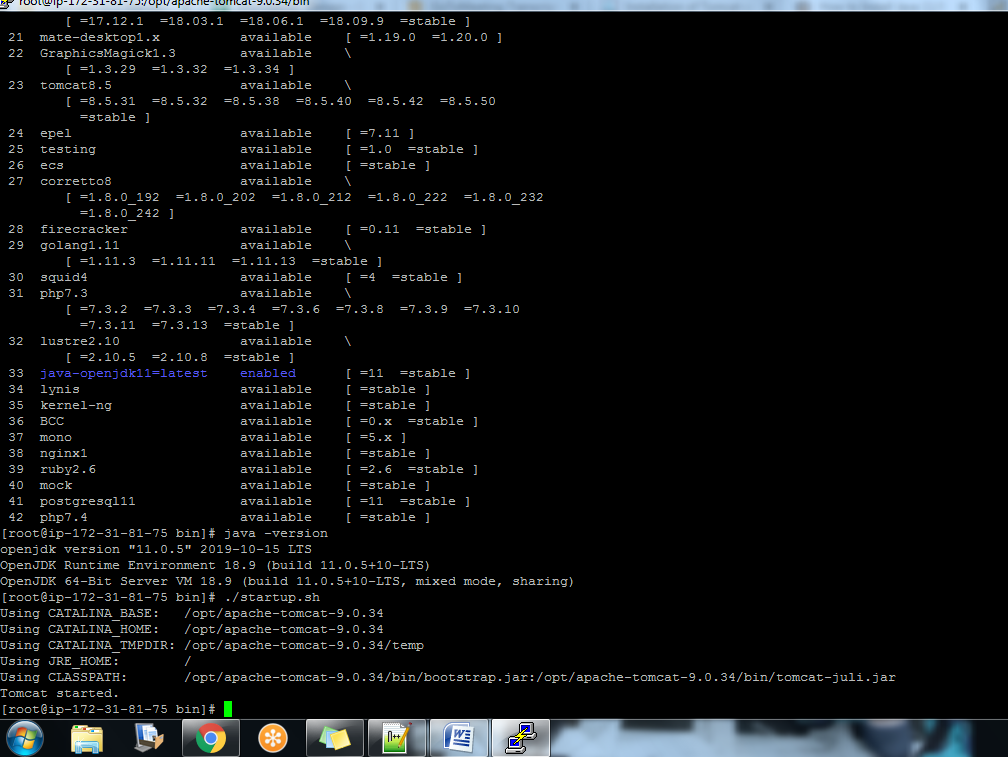
Before we install Tomcat we need to install java with command.

sudo amazon-linux-extras install java-openjdk11

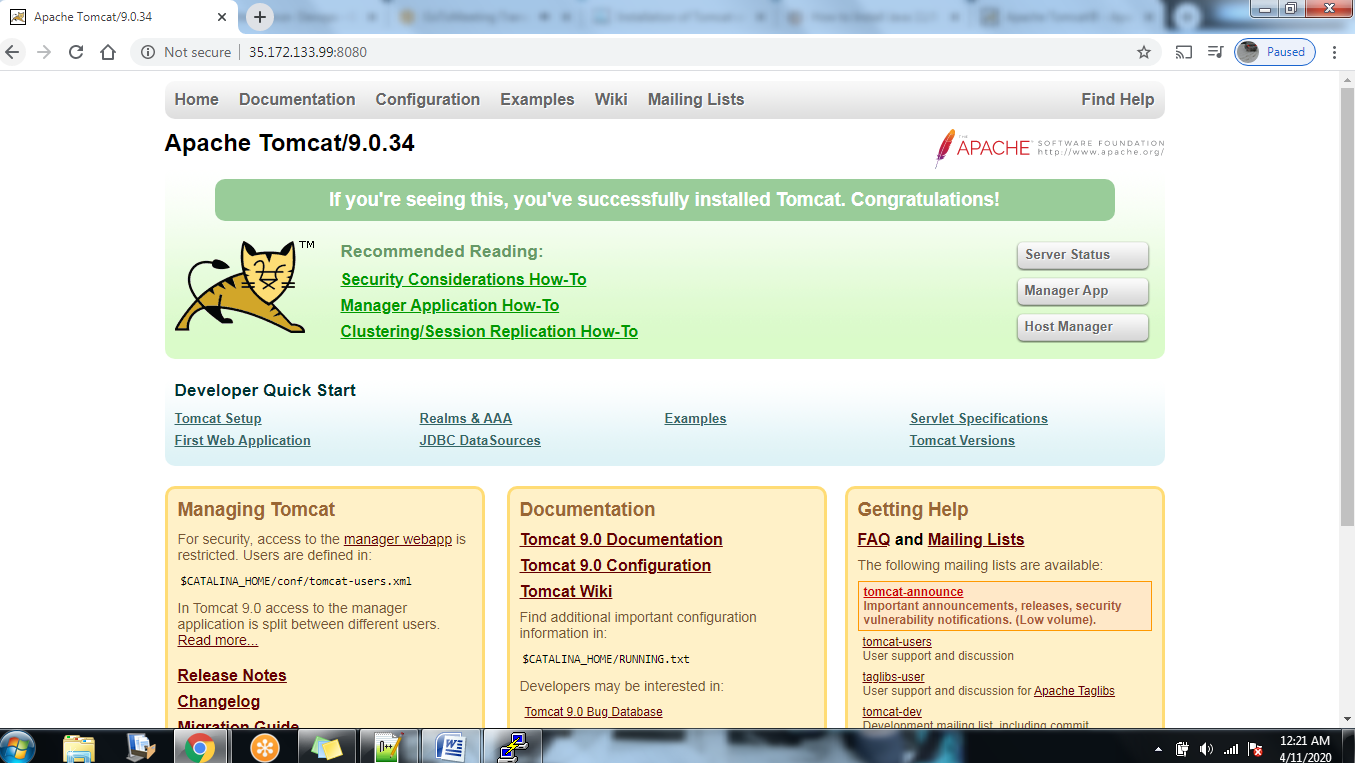




Once Java is installed we can install Tomcat by using command “./startup.sh”

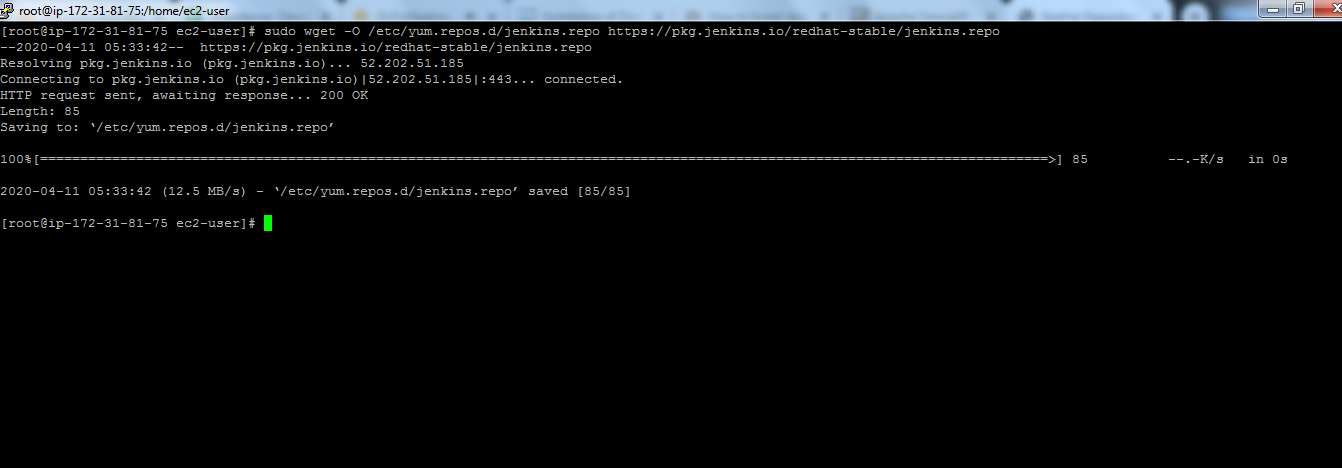


Then we’ll copy the IP Address with port 8080 and if we see following window that mean Apatche Tomcat is installed.

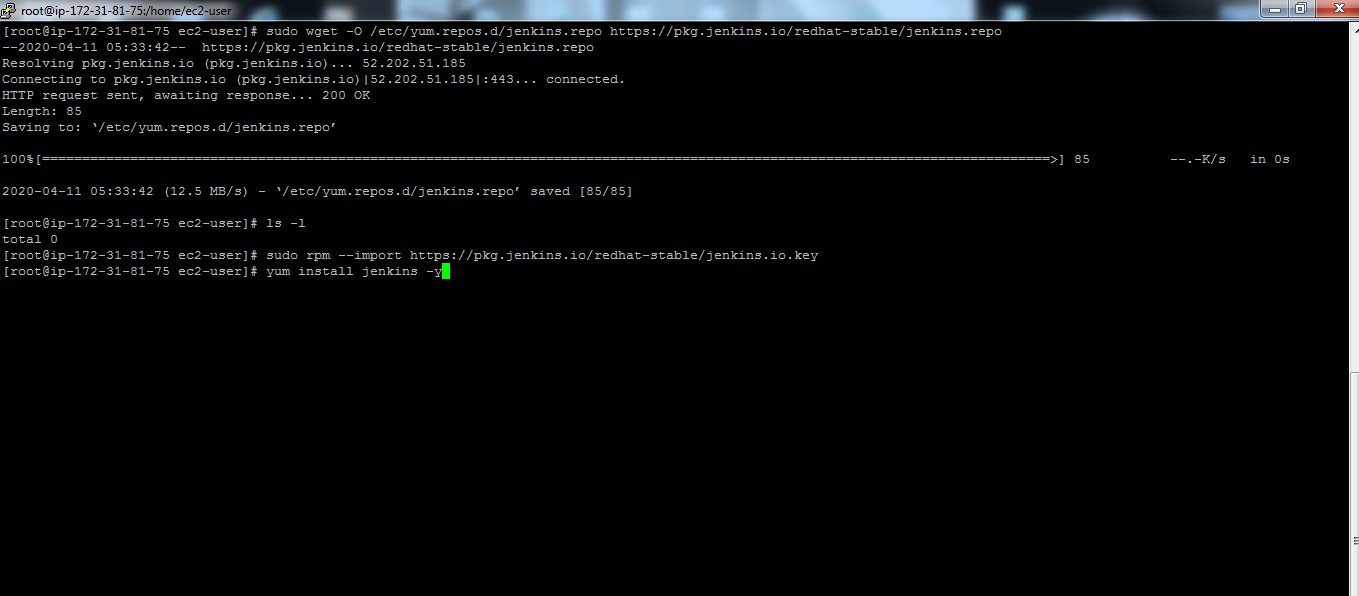


After installing Apatche Tomcat to both appservers. We’ll install Jenkins on Jenkins Master by using following commands.





Then sudo yum install Jenkins -y

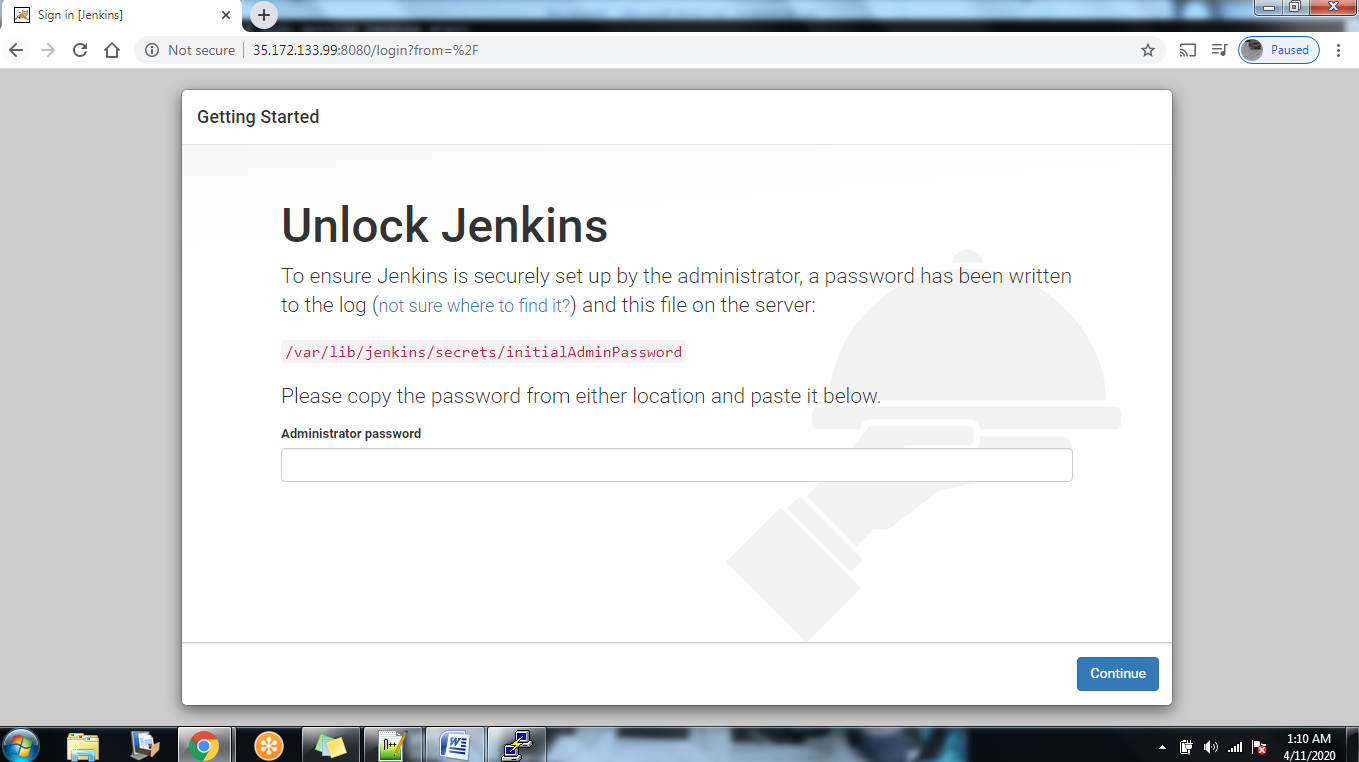


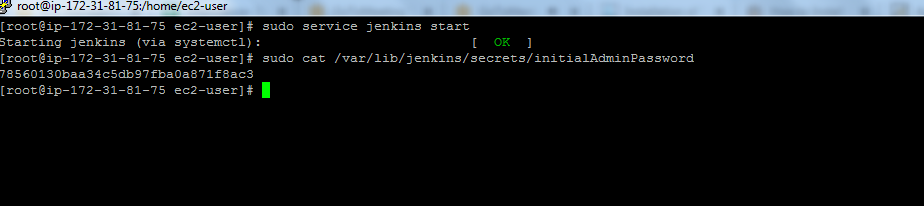
Once it’s installed check if it’s working properly with following command.

sudo service jenkins start

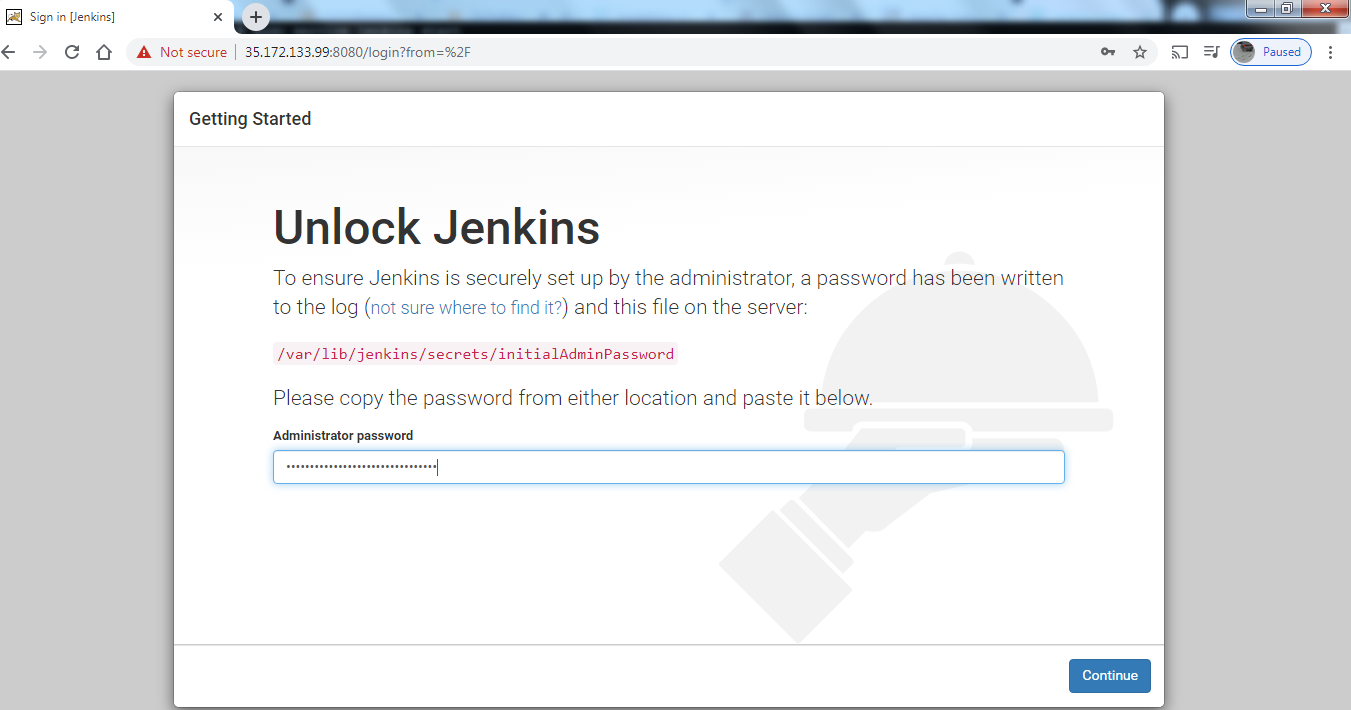


Next copy IP Address:8080 in browser and you’ll see following window then copy path “/var….”>cat “/var..”

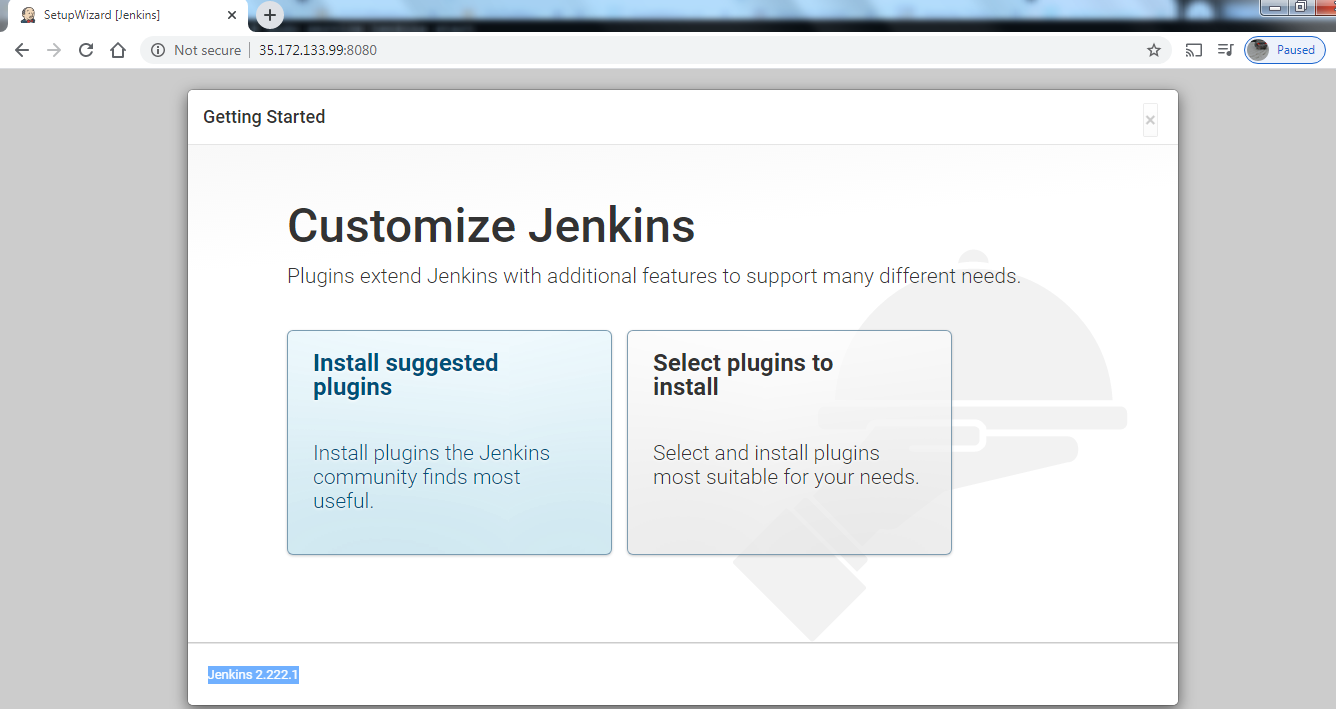


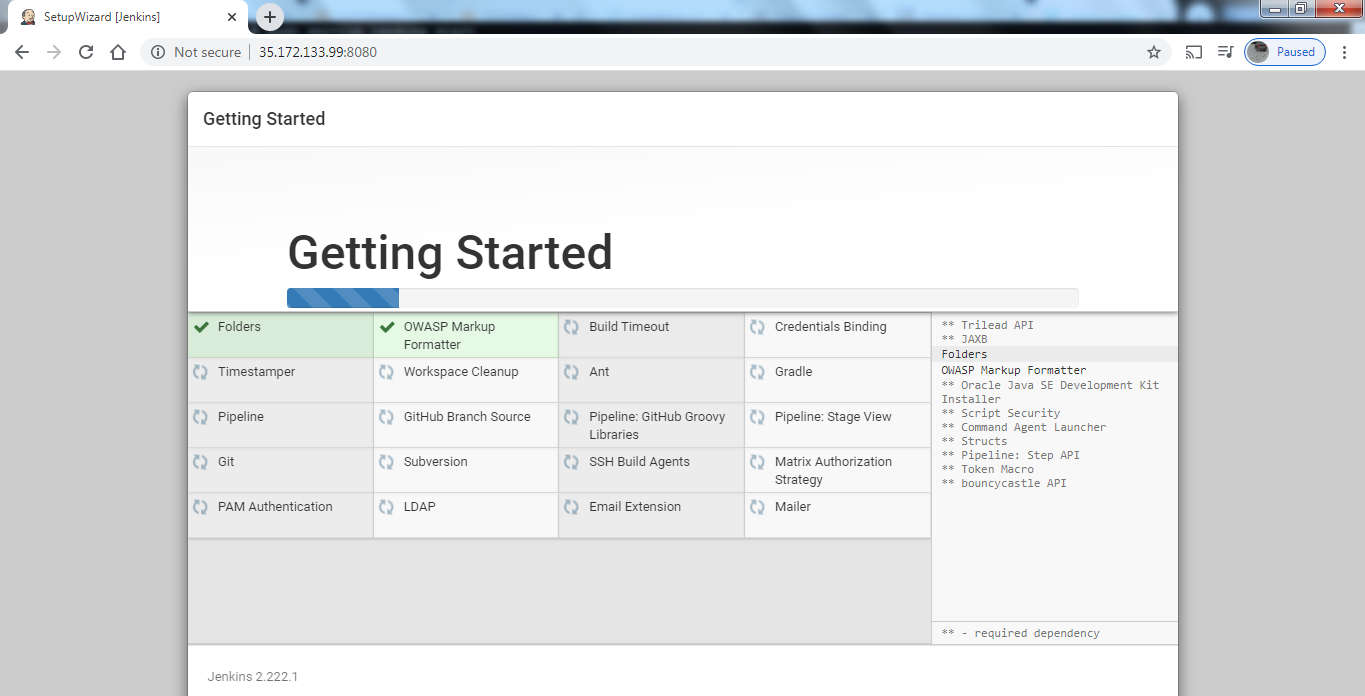


Copy and paste password to Jenkins.

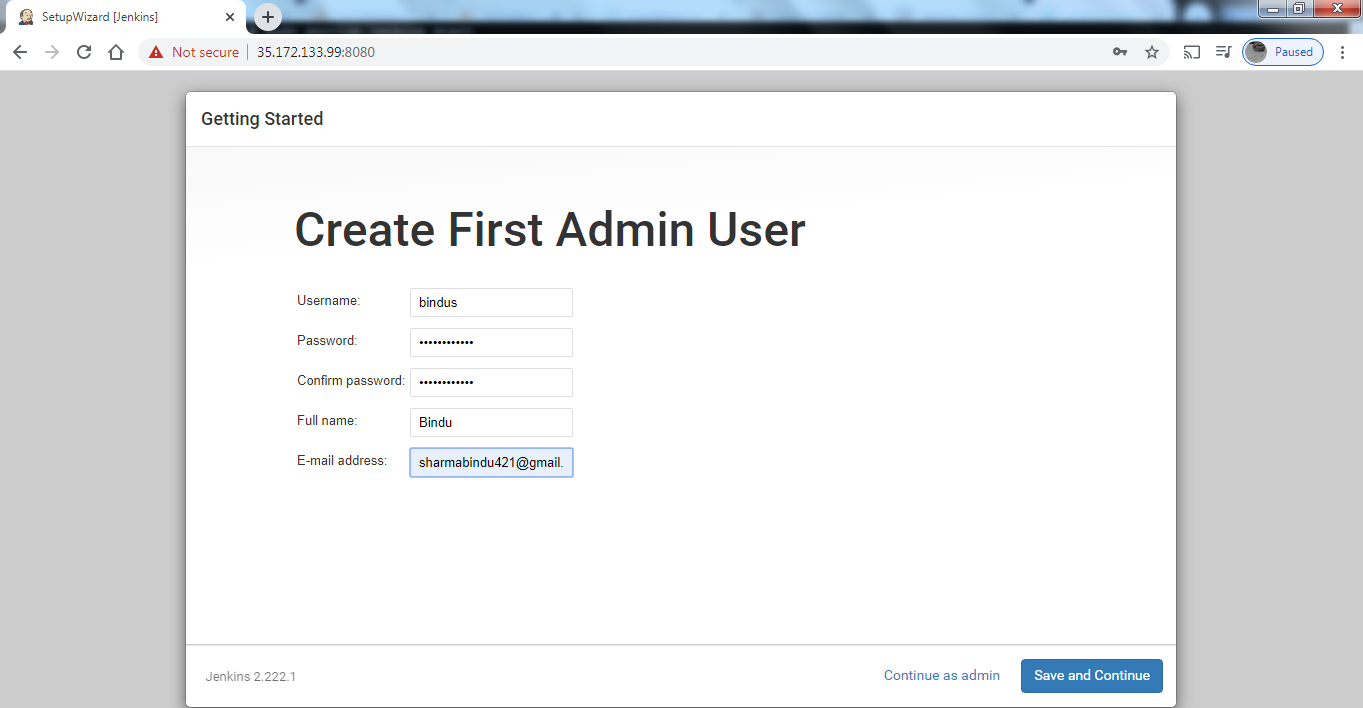


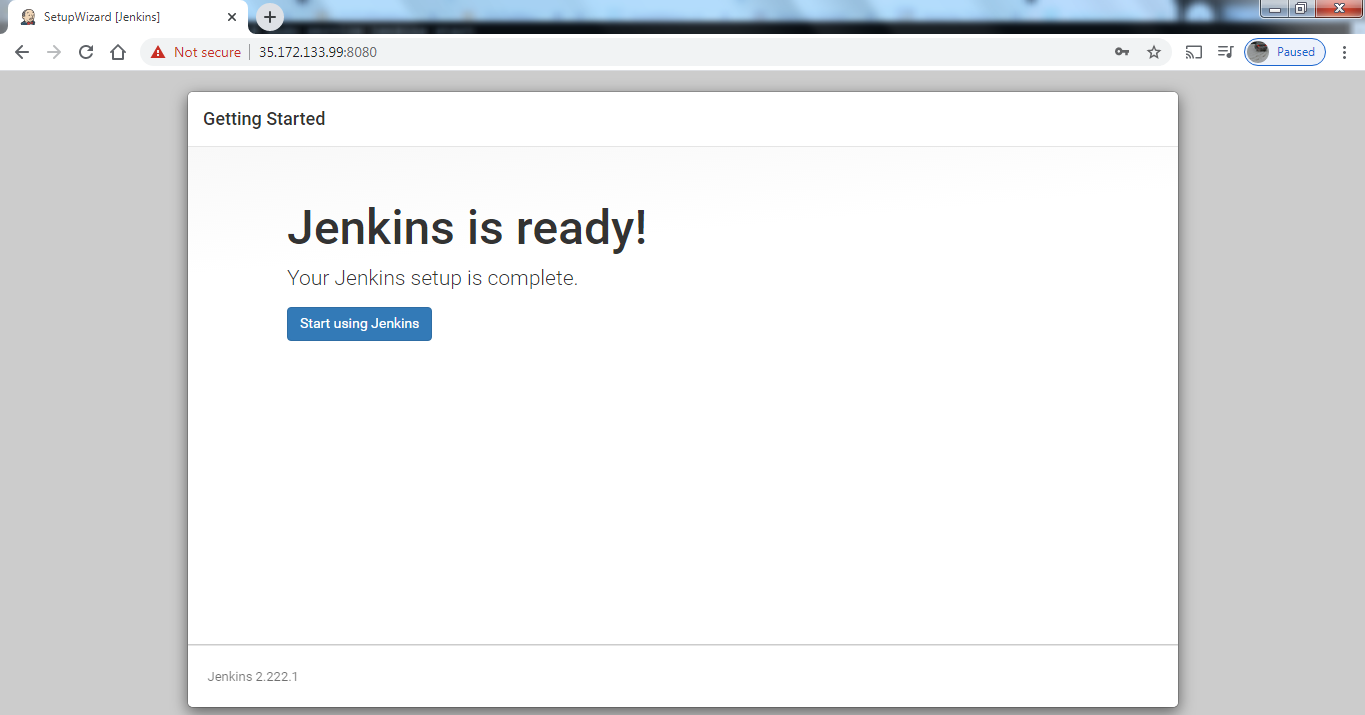
Click on install suggested plugins.





Now create username and password.

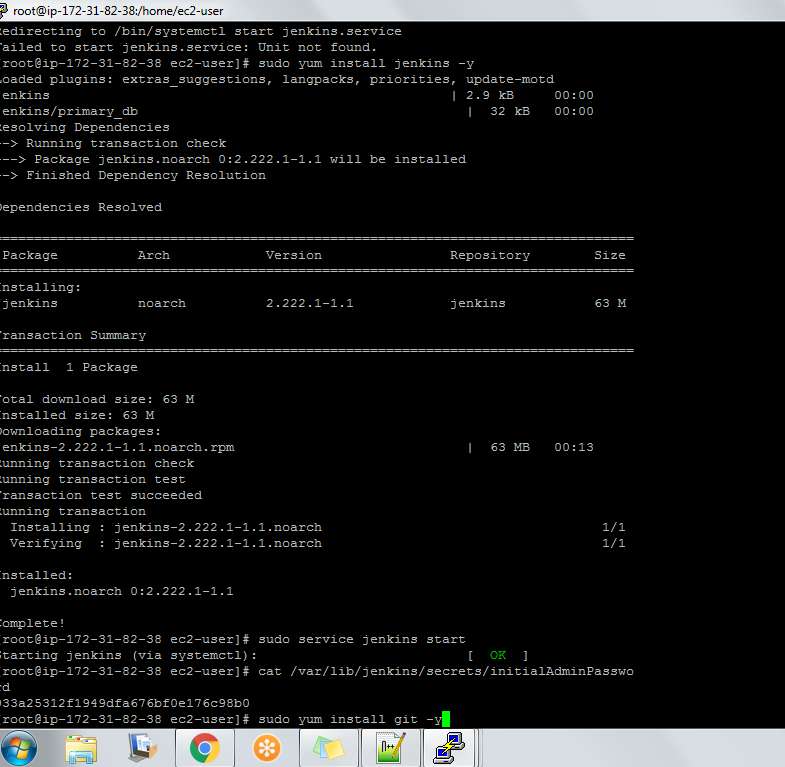


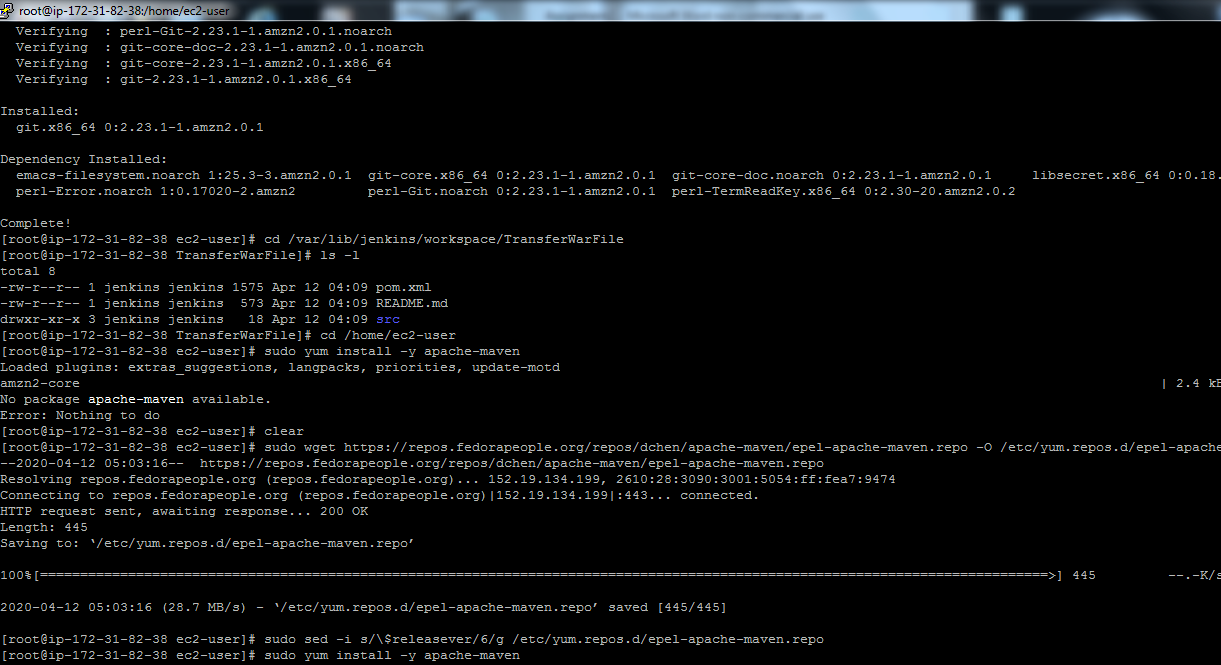




Now Jenkins is ready to use. Before we proceed with our pipeline job. First we need to download git and maven.

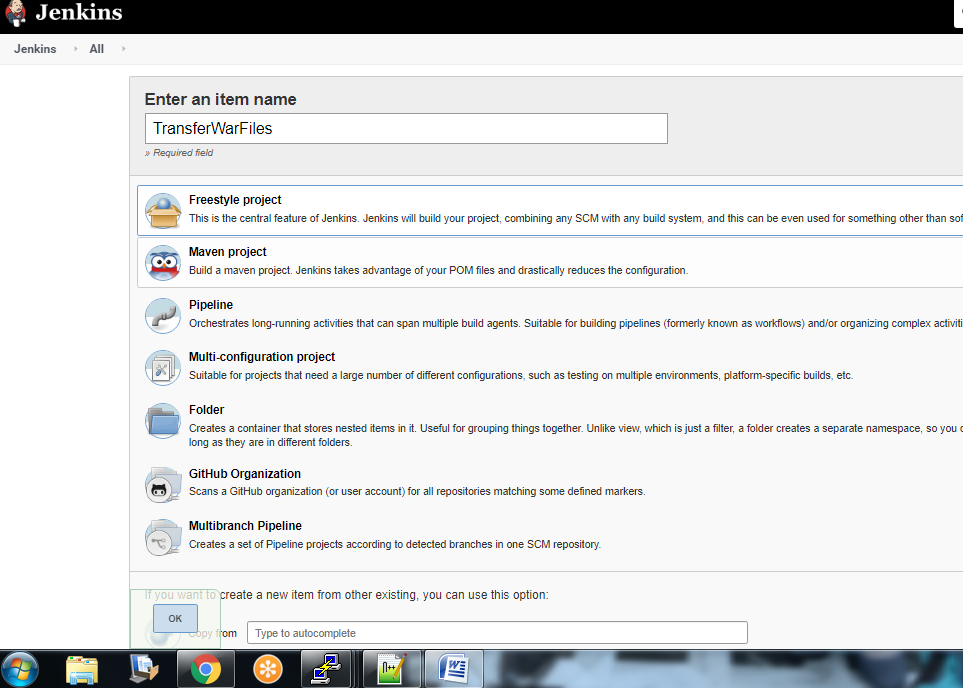
Sudo yum install maven –y

Sudo yum install git -y

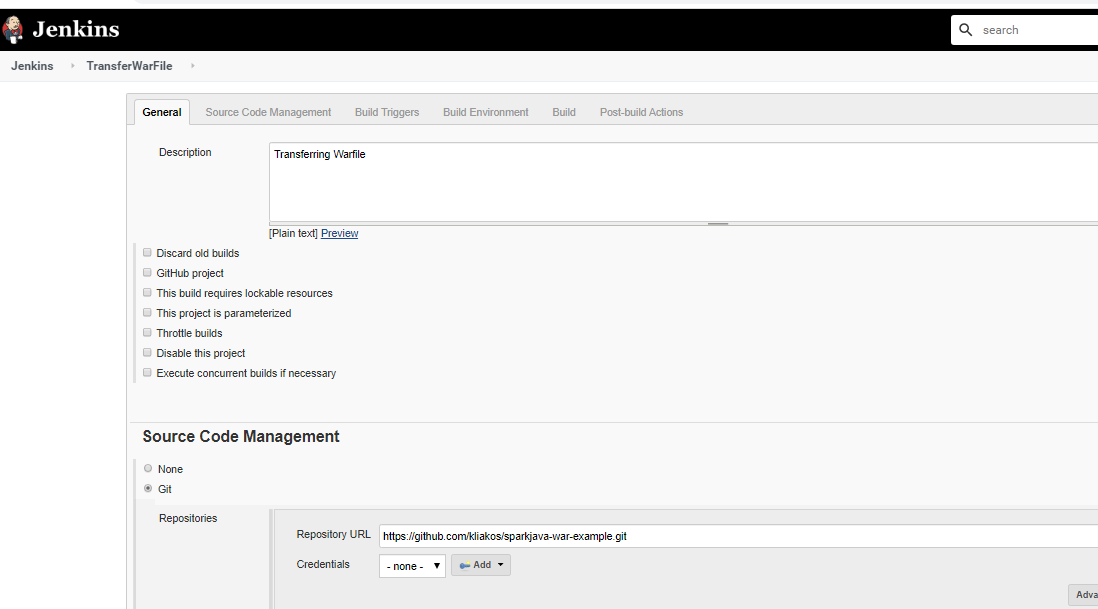


First we need to create a pipeline job and write pipeline grovy script. Before that we need to create a job to transfer files to both servers.

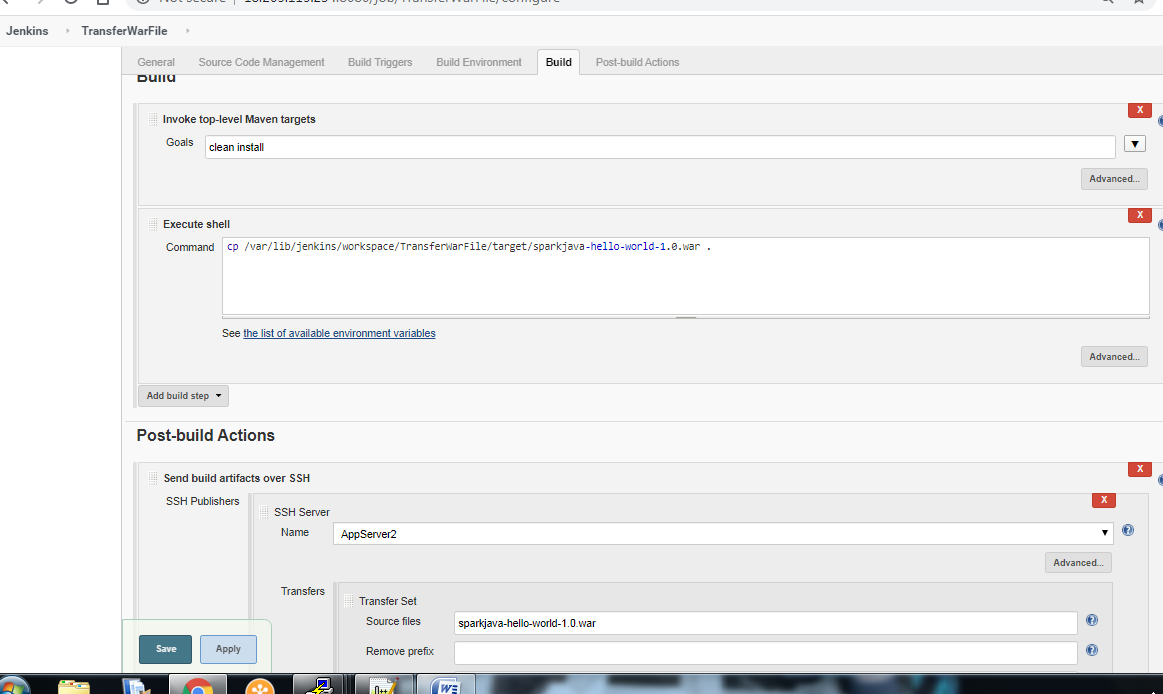
Click on new item> Give a name to file> freestyle project>ok



Then enter gitrepo url and description under description tab.



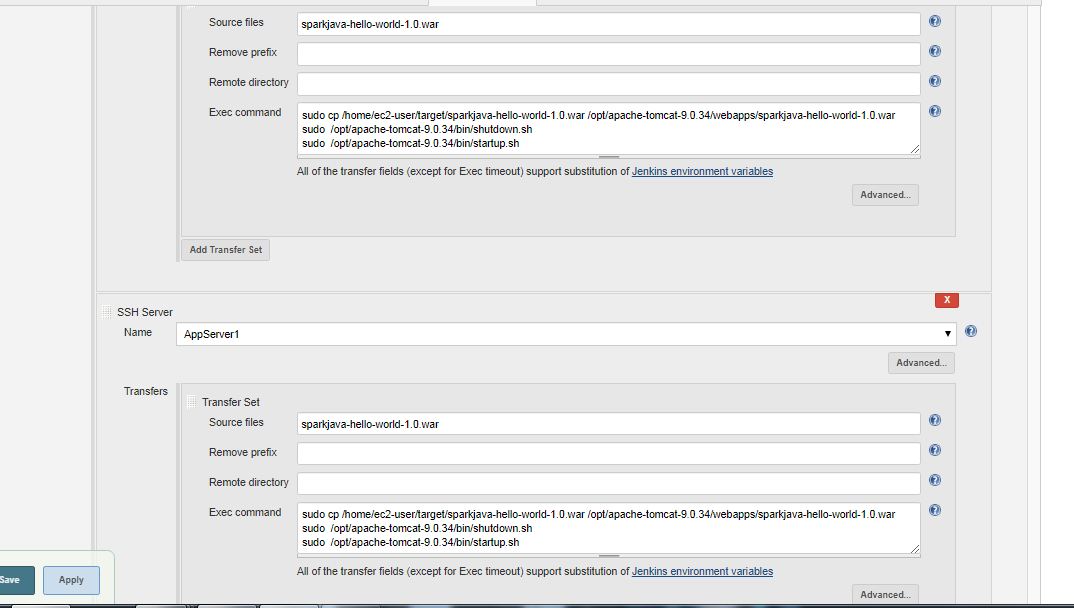
Next we’ll use maven clean install command to create war file. And we’ll use send build artifacts over SSH to copy files to app servers.



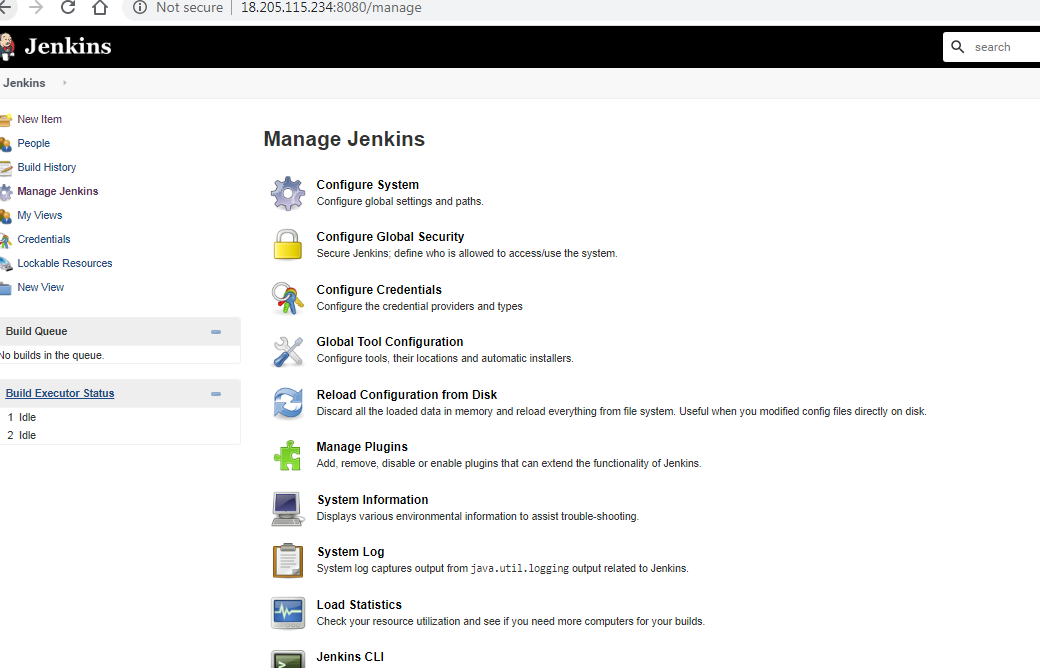
Also we need to use “sudo cp /home/ec2-user/target/sparkjava-hello-world-1.0.war /opt/apache-tomcat-9.0.34/webapps/sparkjava-hello-world-1.0.war

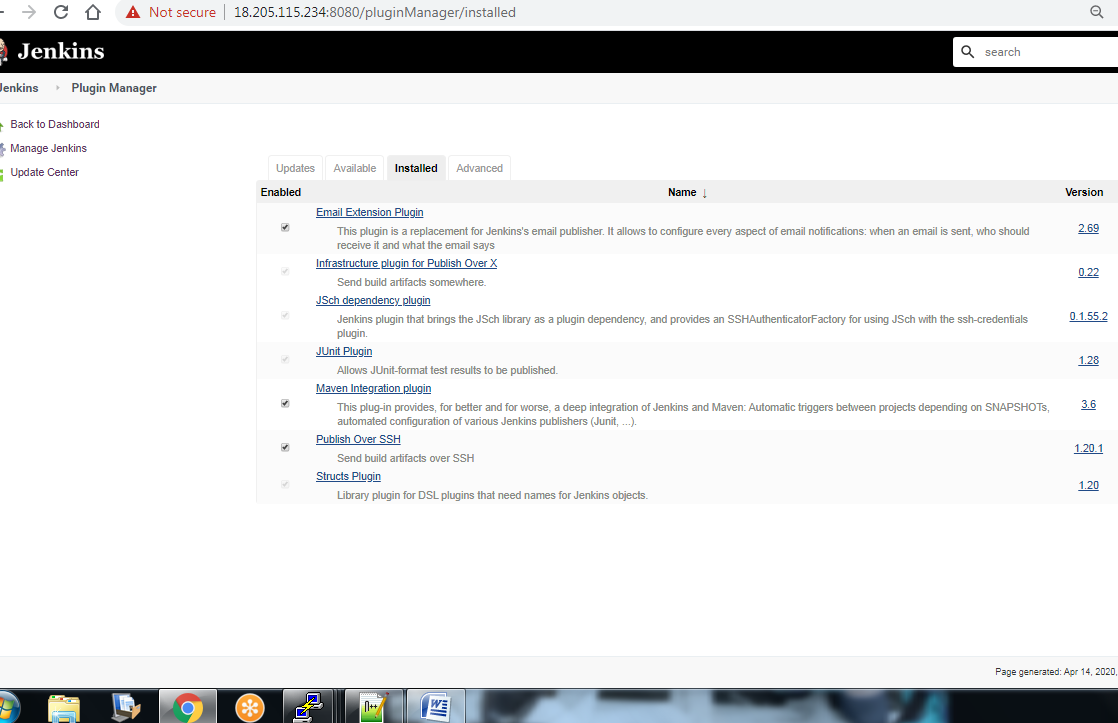
sudo /opt/apache-tomcat-9.0.34/bin/shutdown.sh

sudo /opt/apache-tomcat-9.0.34/bin/startup.sh” to restart apatche and copy files.

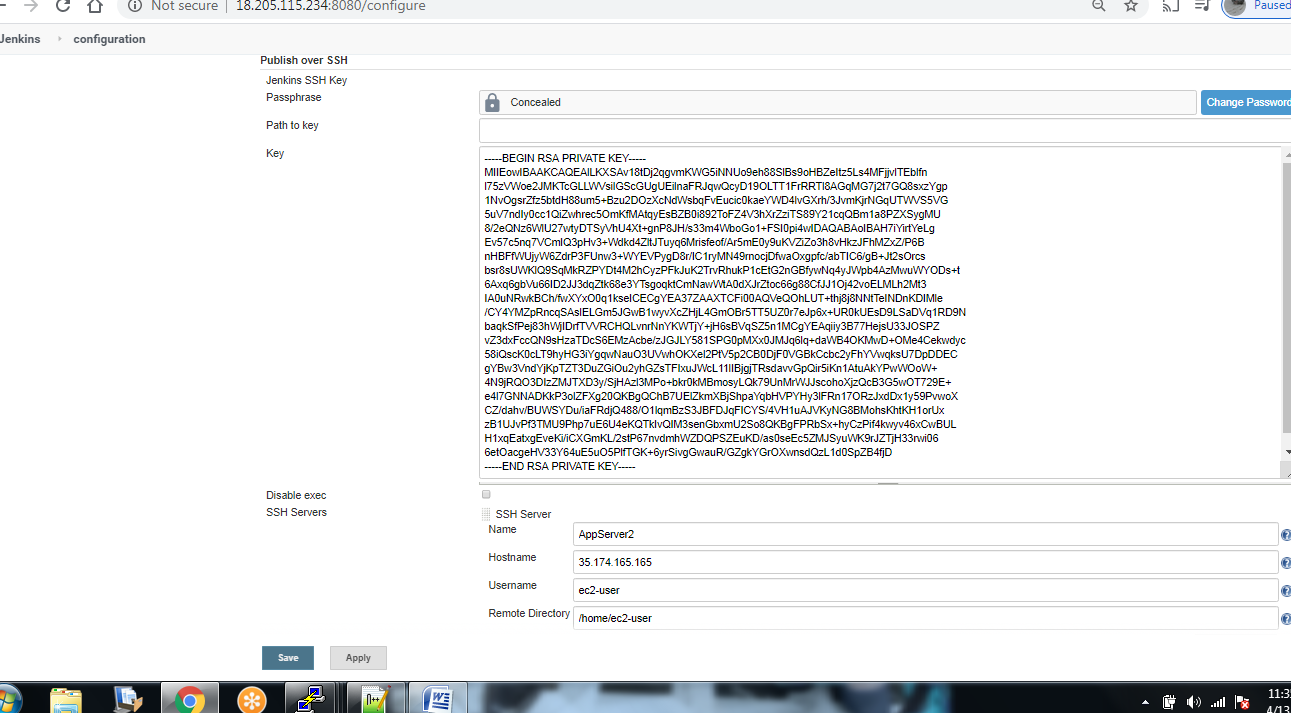


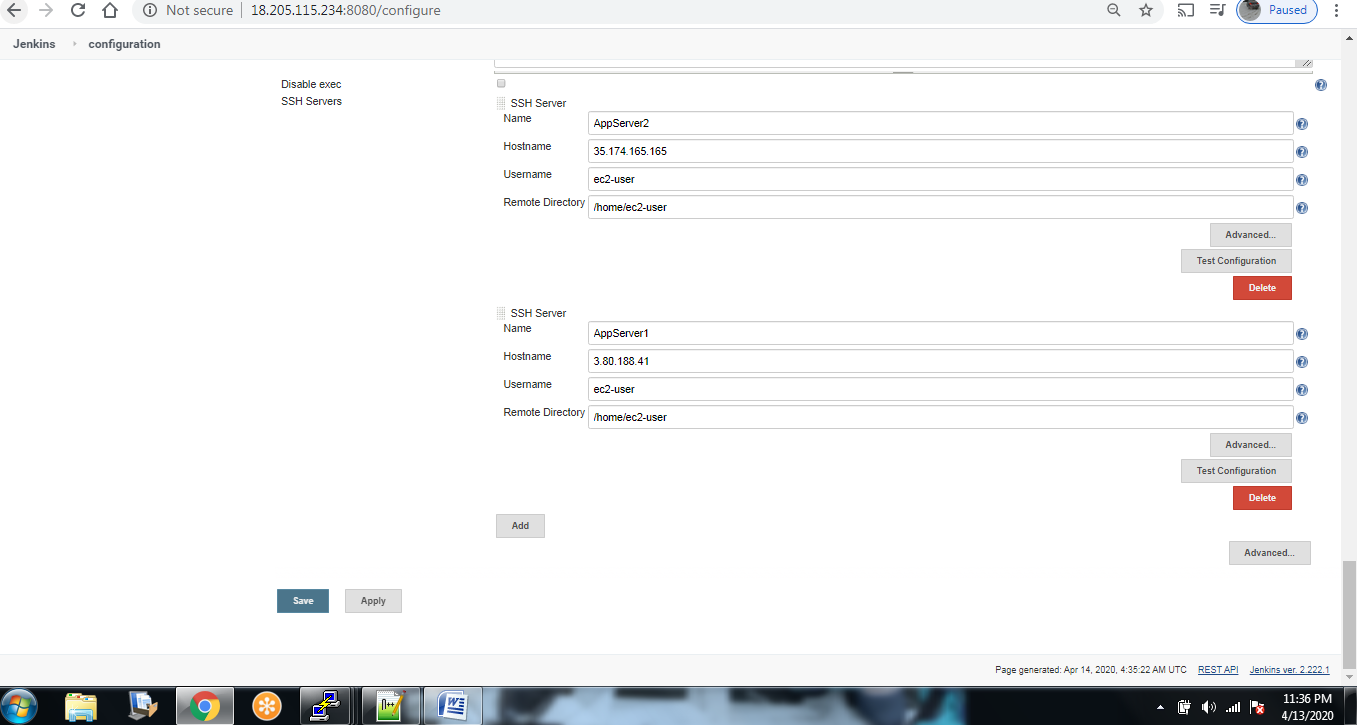
To use send build artifacts over SSH to copy files to app servers. First we need to install plugin PUBLISH OVER SSH. Manage Jenkins>Manage Plugins>Available>Publish over SSH.



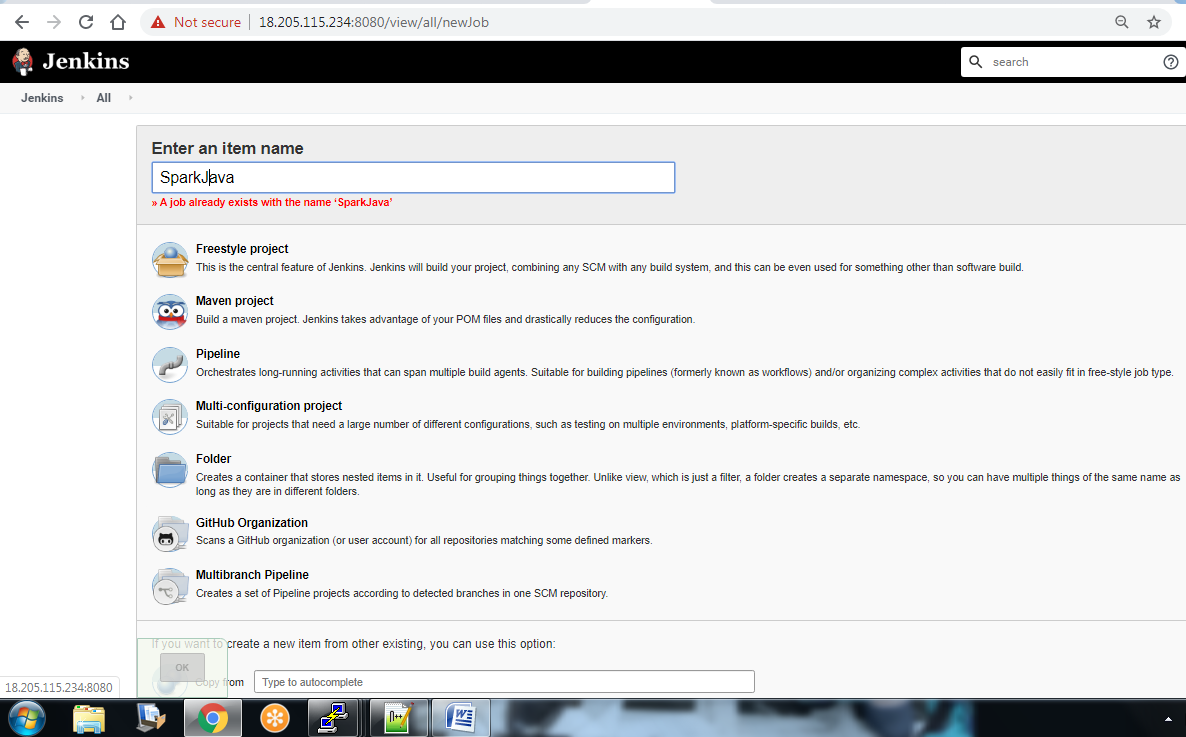


Next we need to configure both servers under Manage Jenkins>Configure System>Publish over SSH>Add key>Server Name> host Name> username>remote directory.





Now we are ready to build our pipeline. Click on new item> Give a name to file> pipeline>ok



Then we’ll enter pipeline script under pipeline.Groovy script:

pipeline {

agent any

stages {

stage ('Copying Repository')

{

steps {

echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

echo 'Copying Files to Jenkins Server'

sh label: '', script: 'rm -rf sparkjava-war-example '

sh label: '', script: 'git clone https://github.com/kliakos/sparkjava-war-example.git'

echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

}

}

stage ('Maven Build')

{

steps

{

echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

echo 'Running a Maven Build'

sh label: '', script: 'cd sparkjava-war-example && mvn package'

echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

}

}

stage ('Transfer War Files') {

steps{

echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

echo 'Transferring War Files to AppServer'

build 'TransferWarFile'

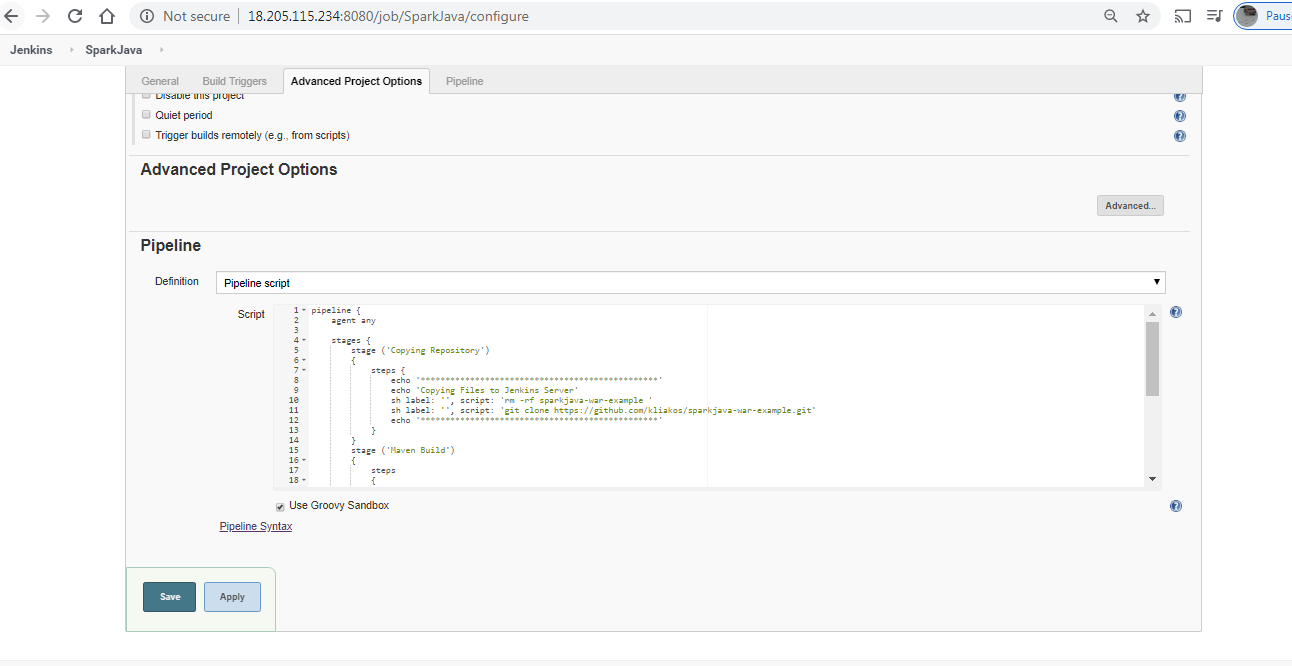
echo '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

}

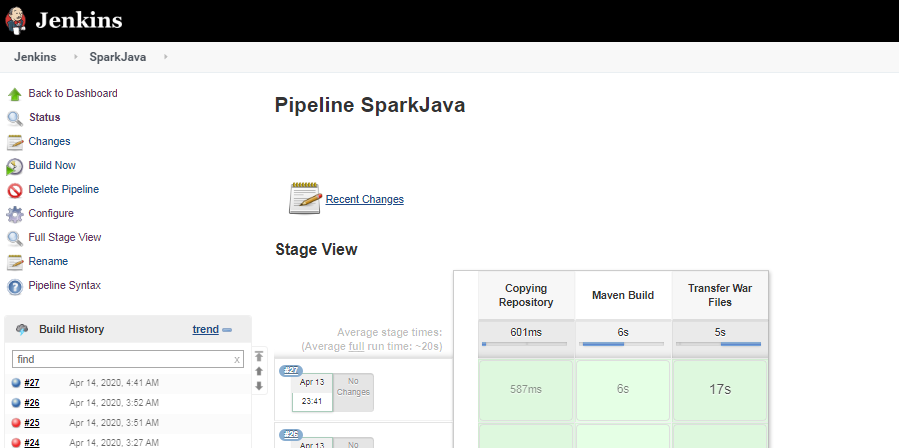
}

}

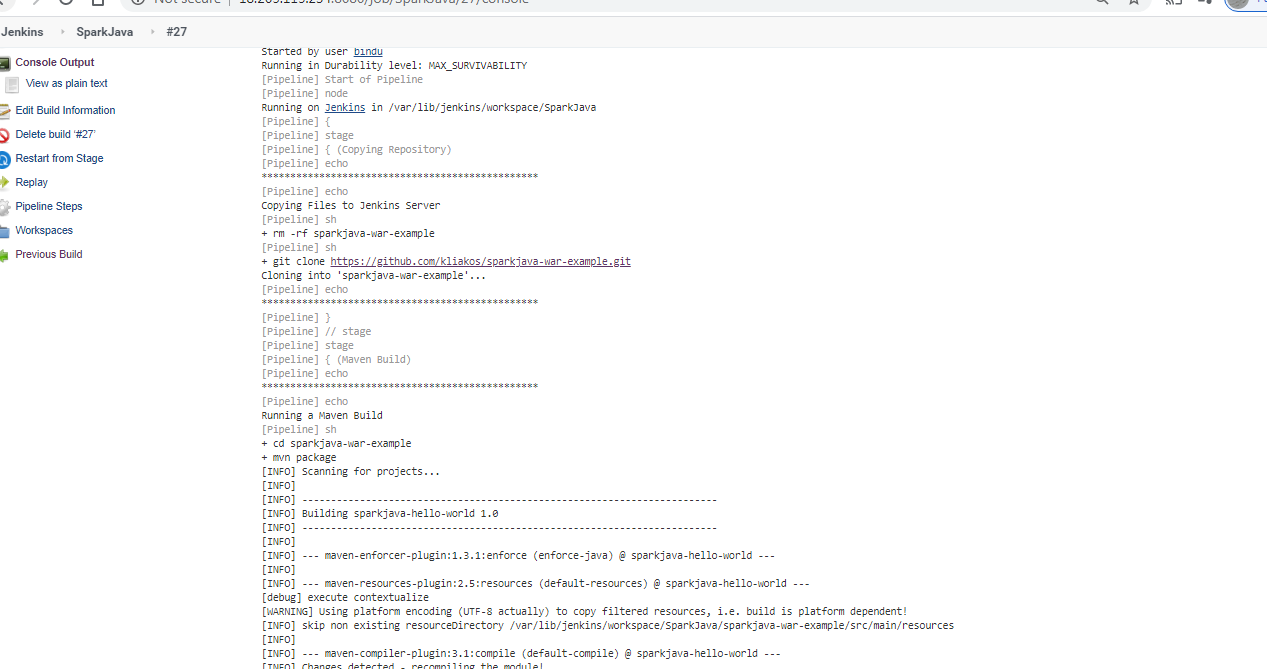
}

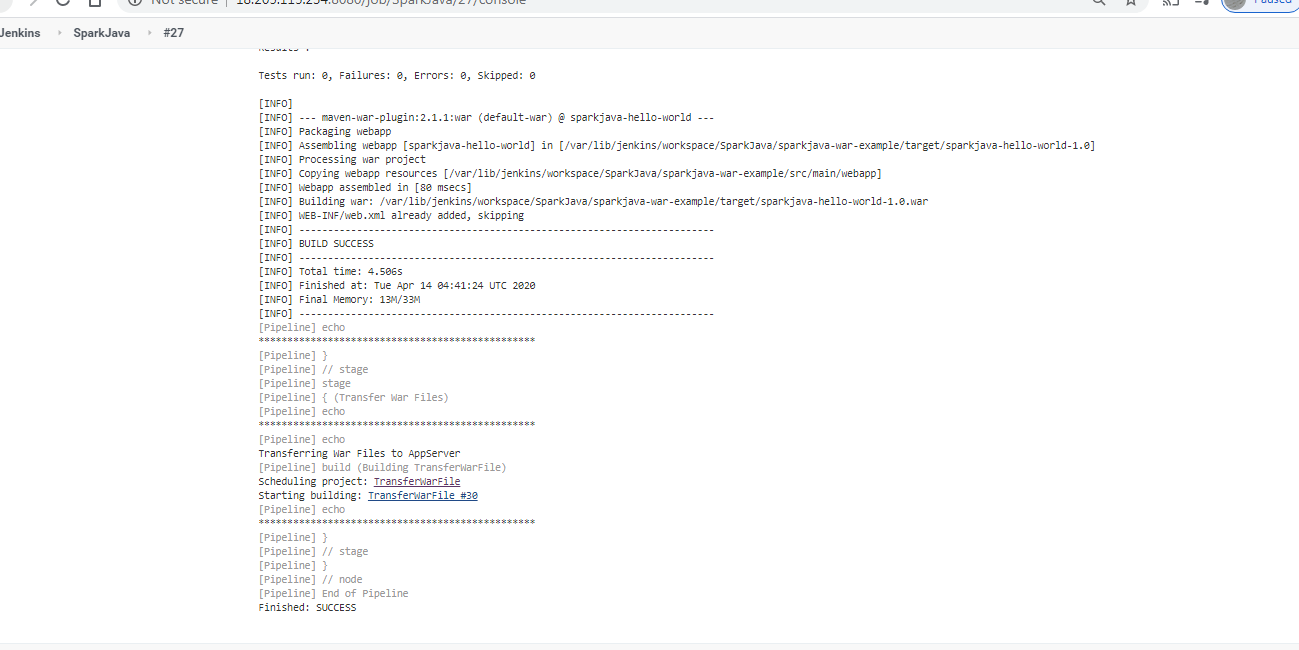


Once we hit build we’ll see following:



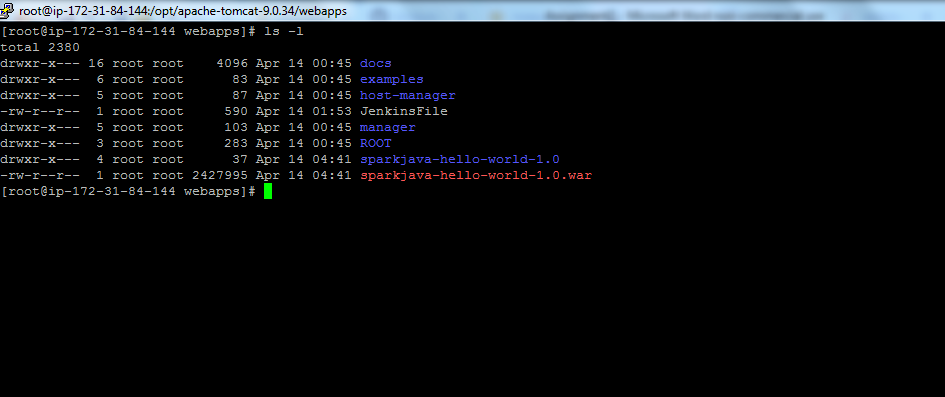
Console Output:



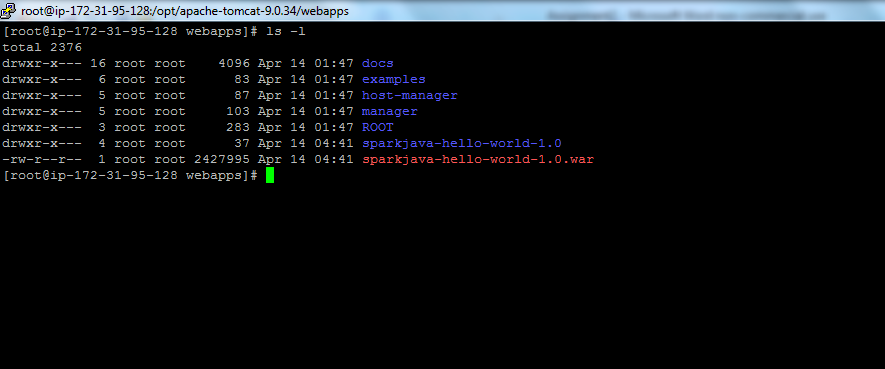


Once it runs successfully we can check all appservers had file saved and Java application is running in Tomcat server.

App server1:



Appserver2:



Java application is running.

