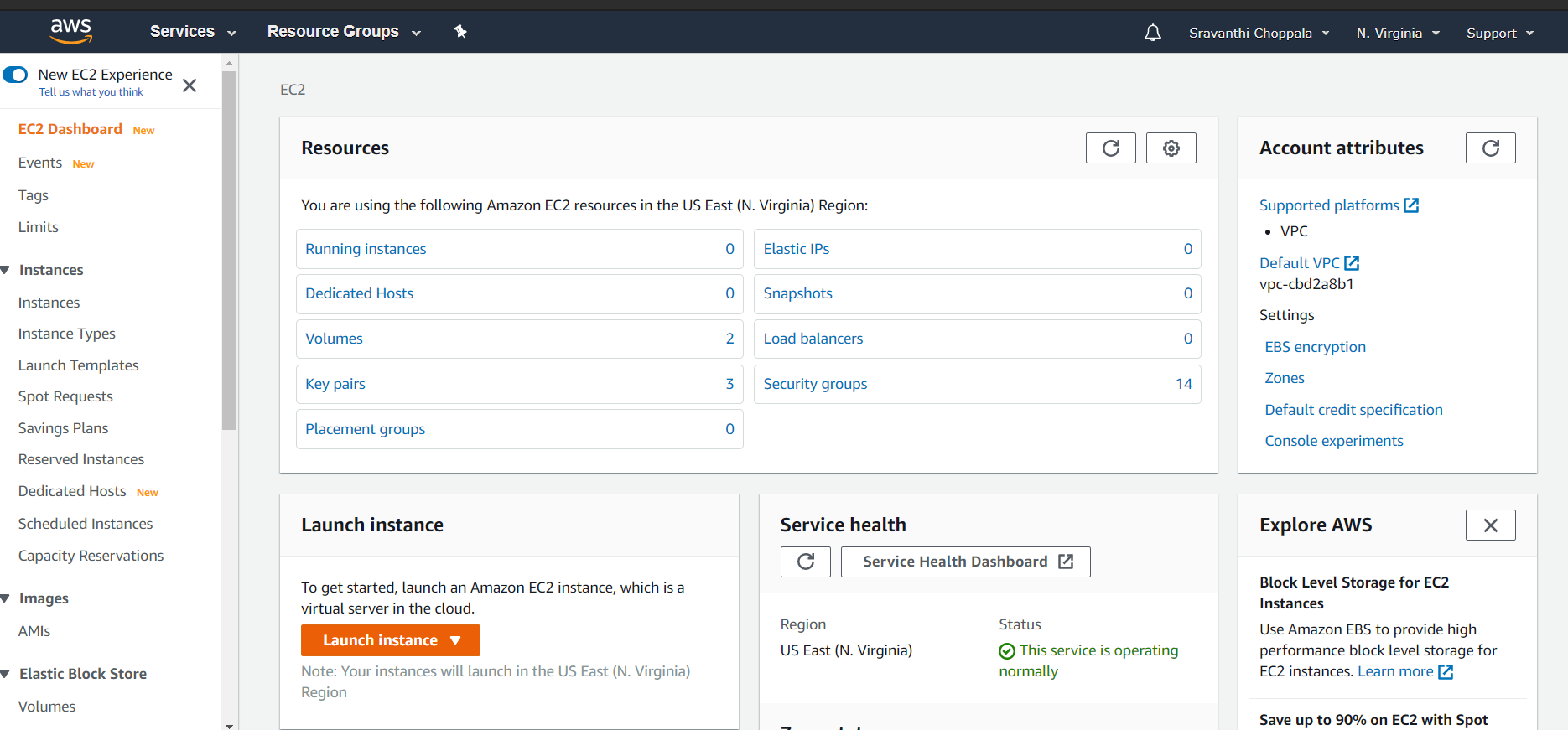
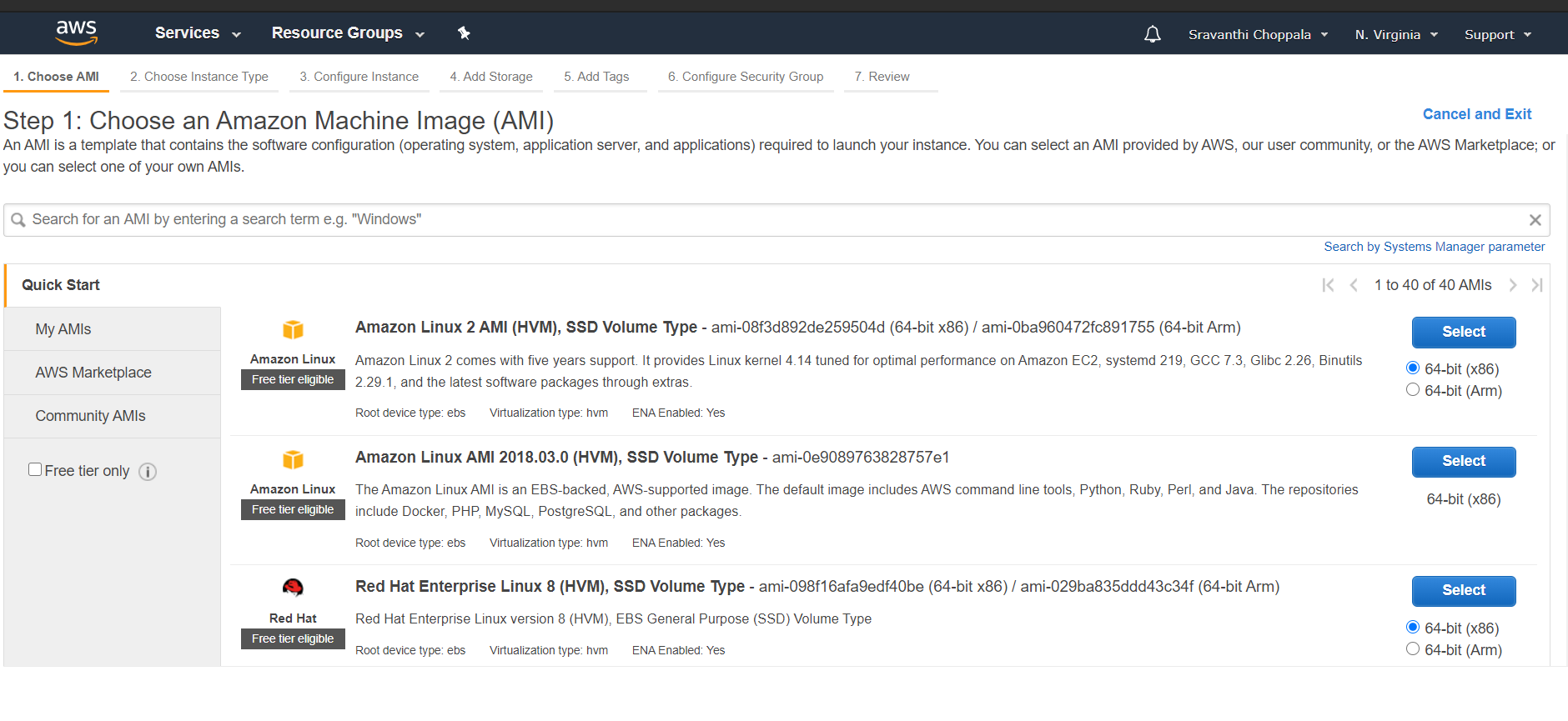
# Deployment of Application in to cloud

To create EC2 instance follow the below steps

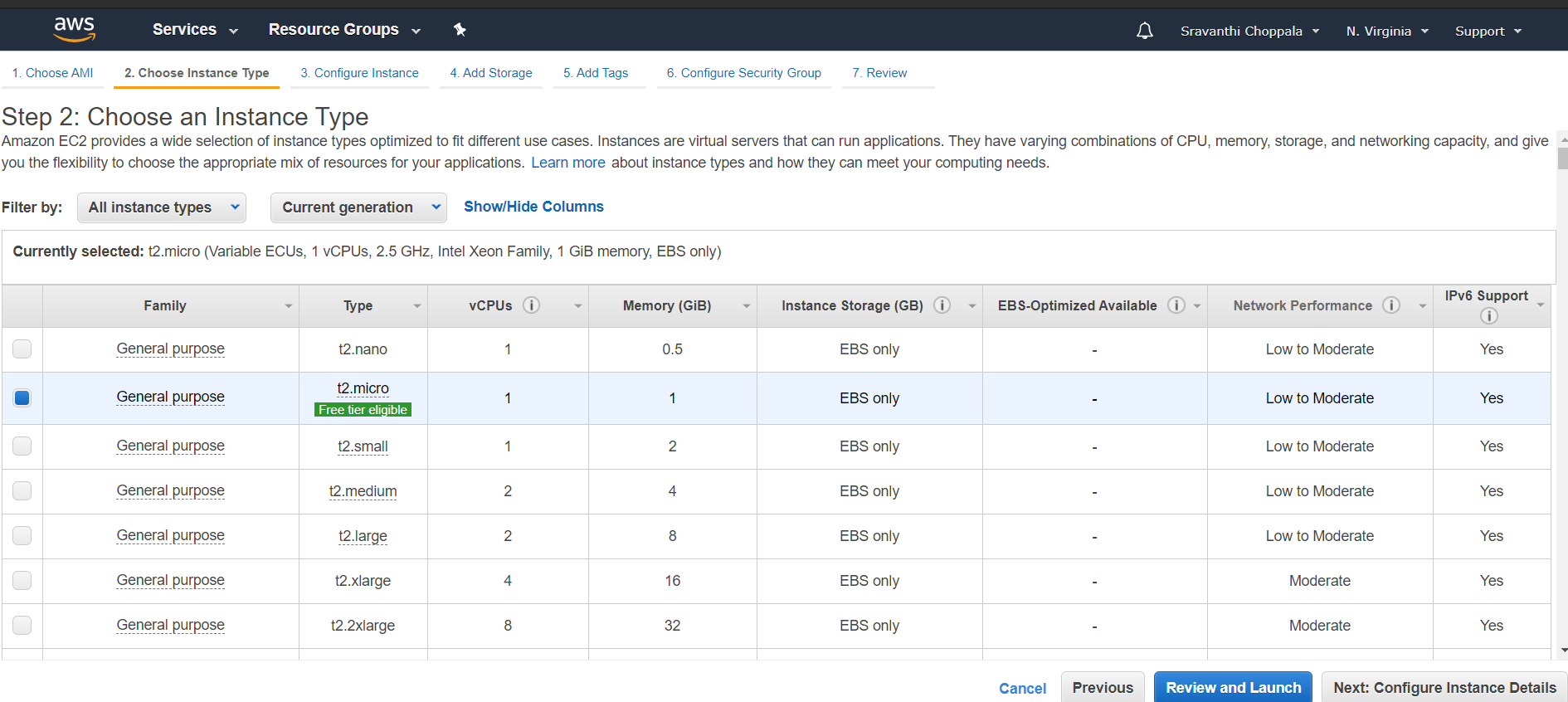
* From **services** select **EC2** instance and click on **Launch instance**



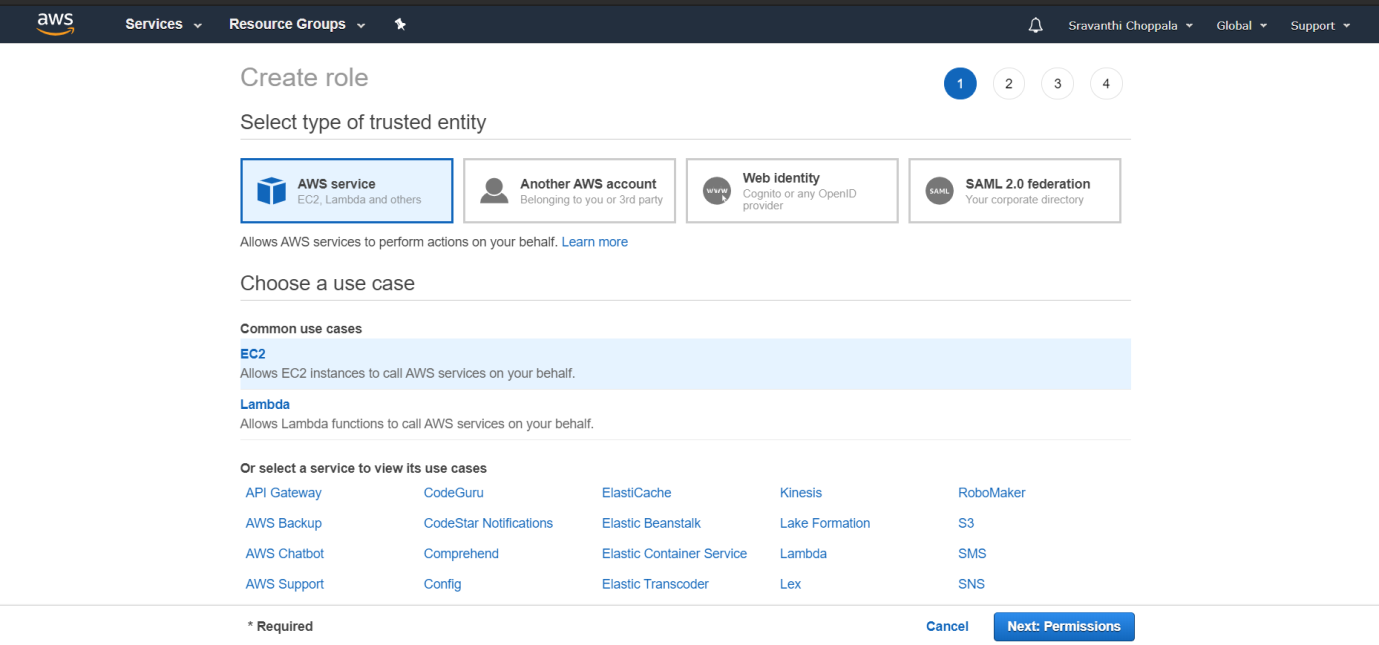
* Select Linux Amazon Machine Image(AMI)



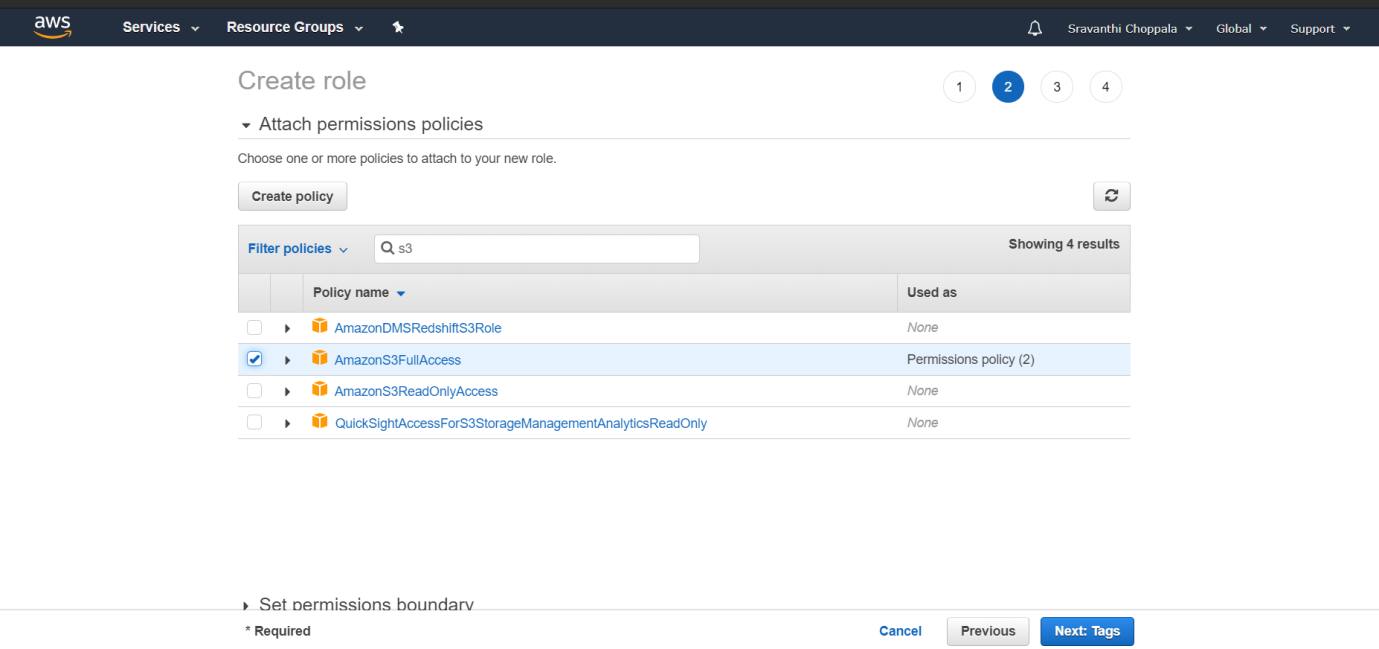
* Select General purpose instance type

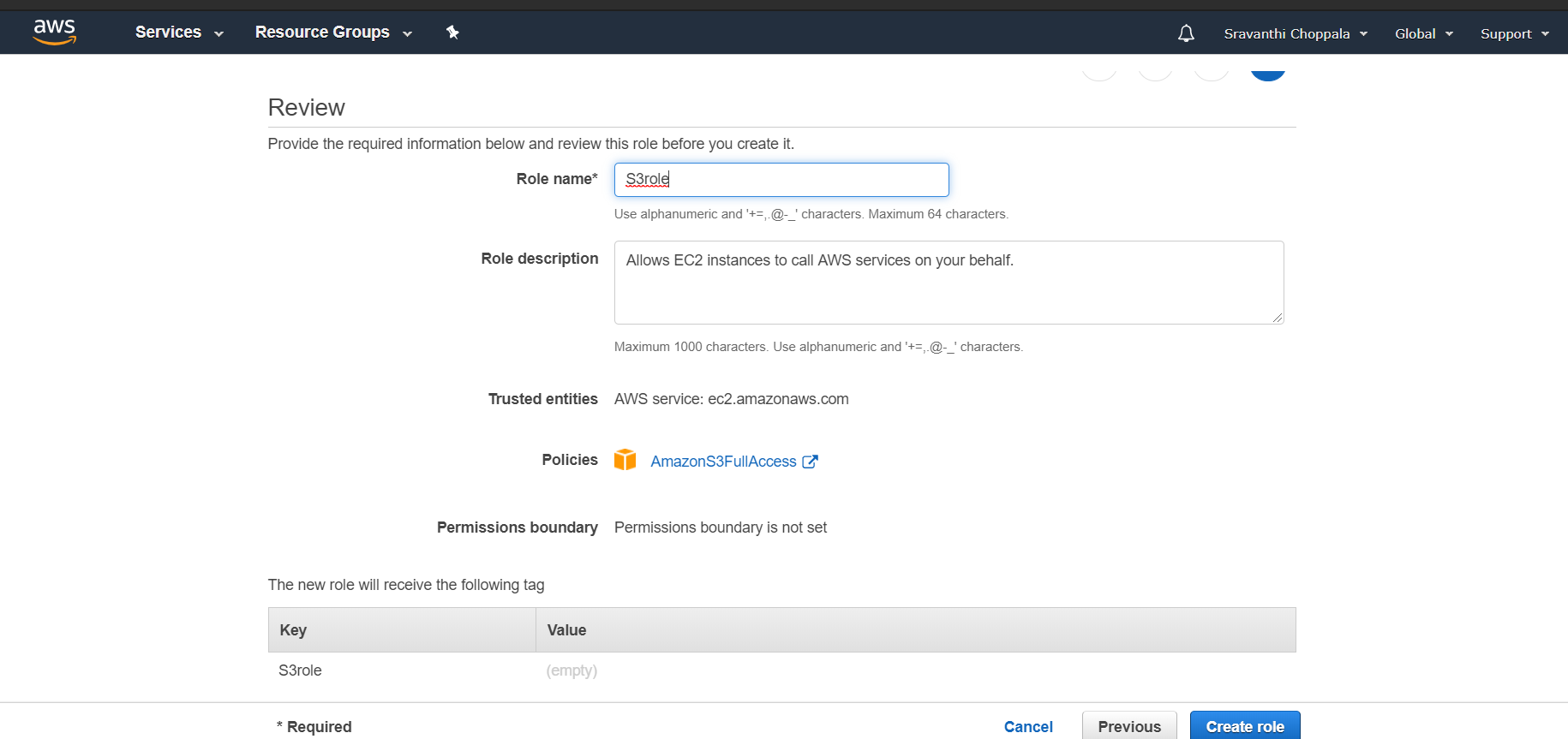


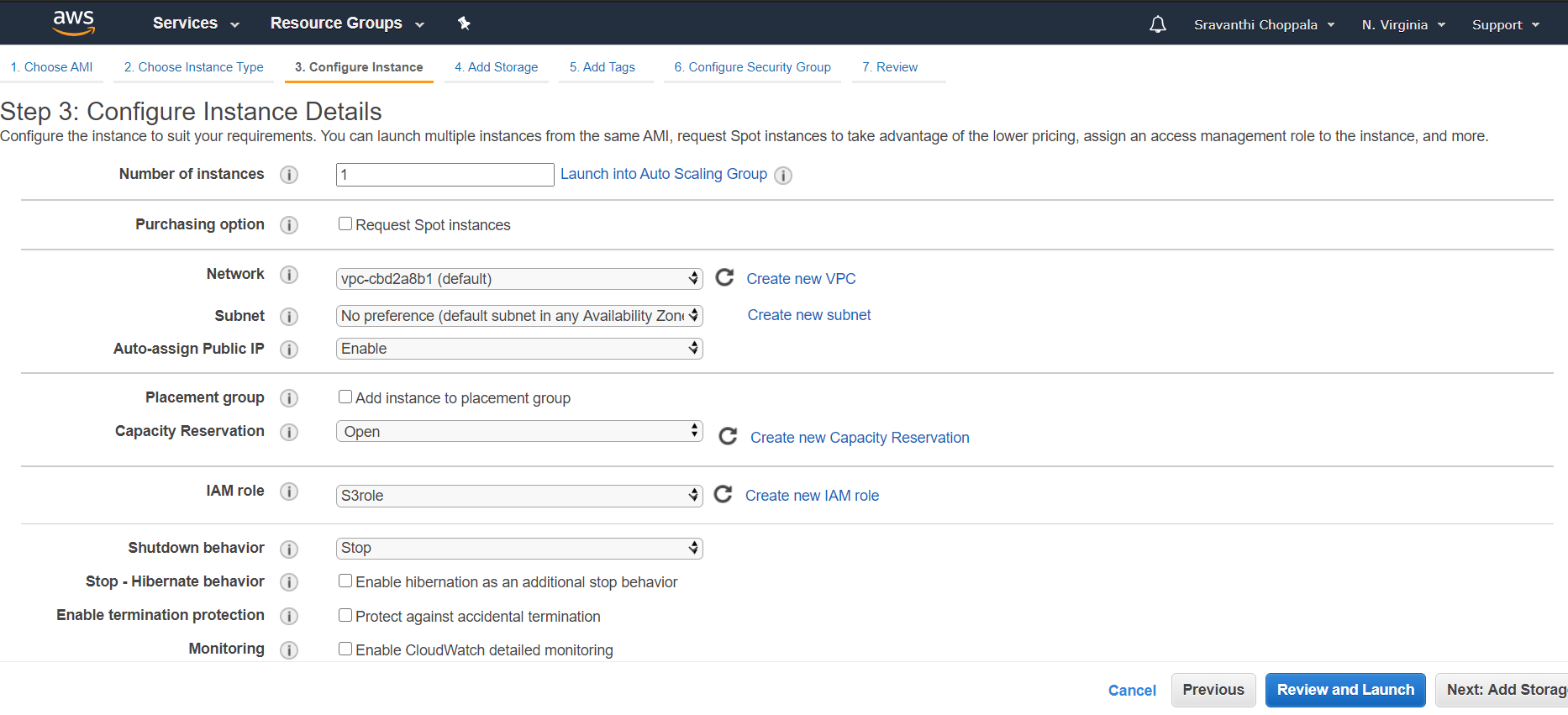
* Create role select **AWS Service**



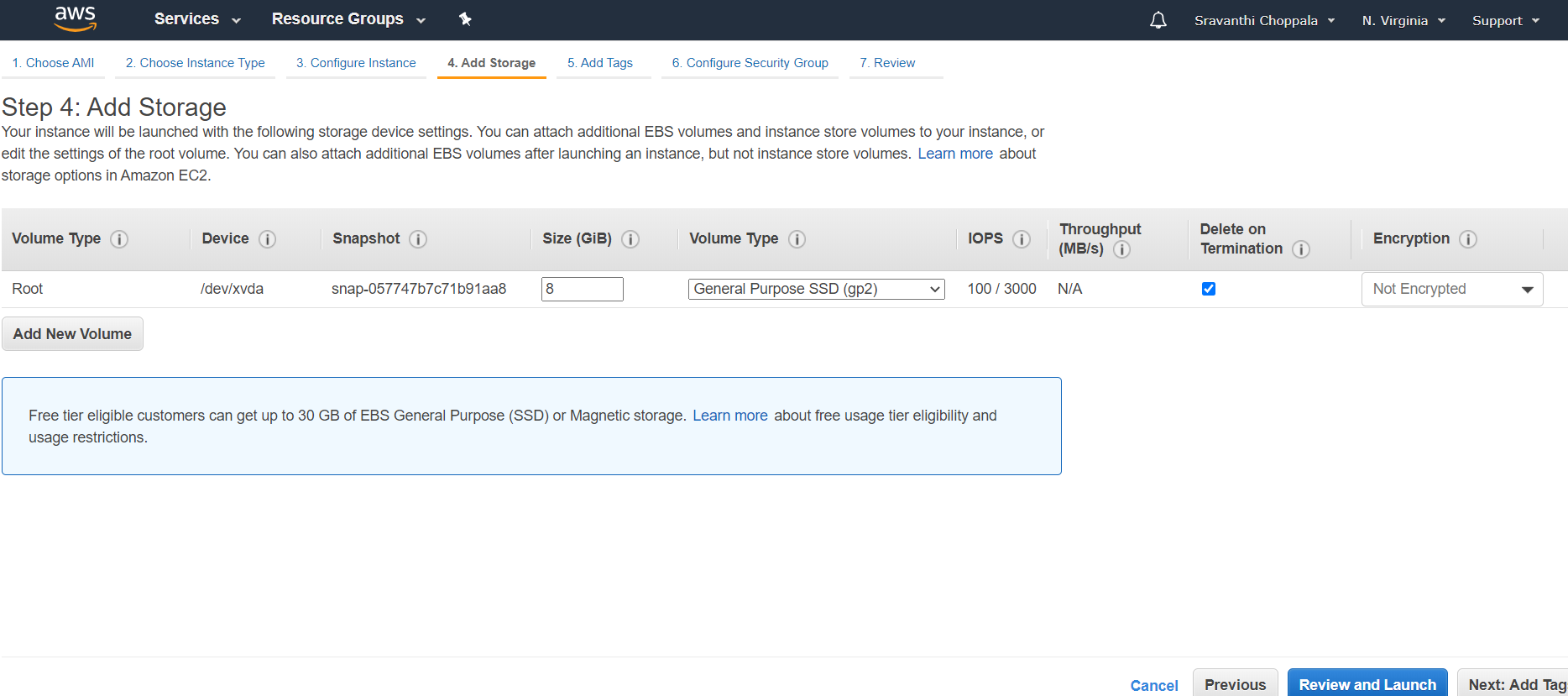
* Give permission to AmozonS3Access



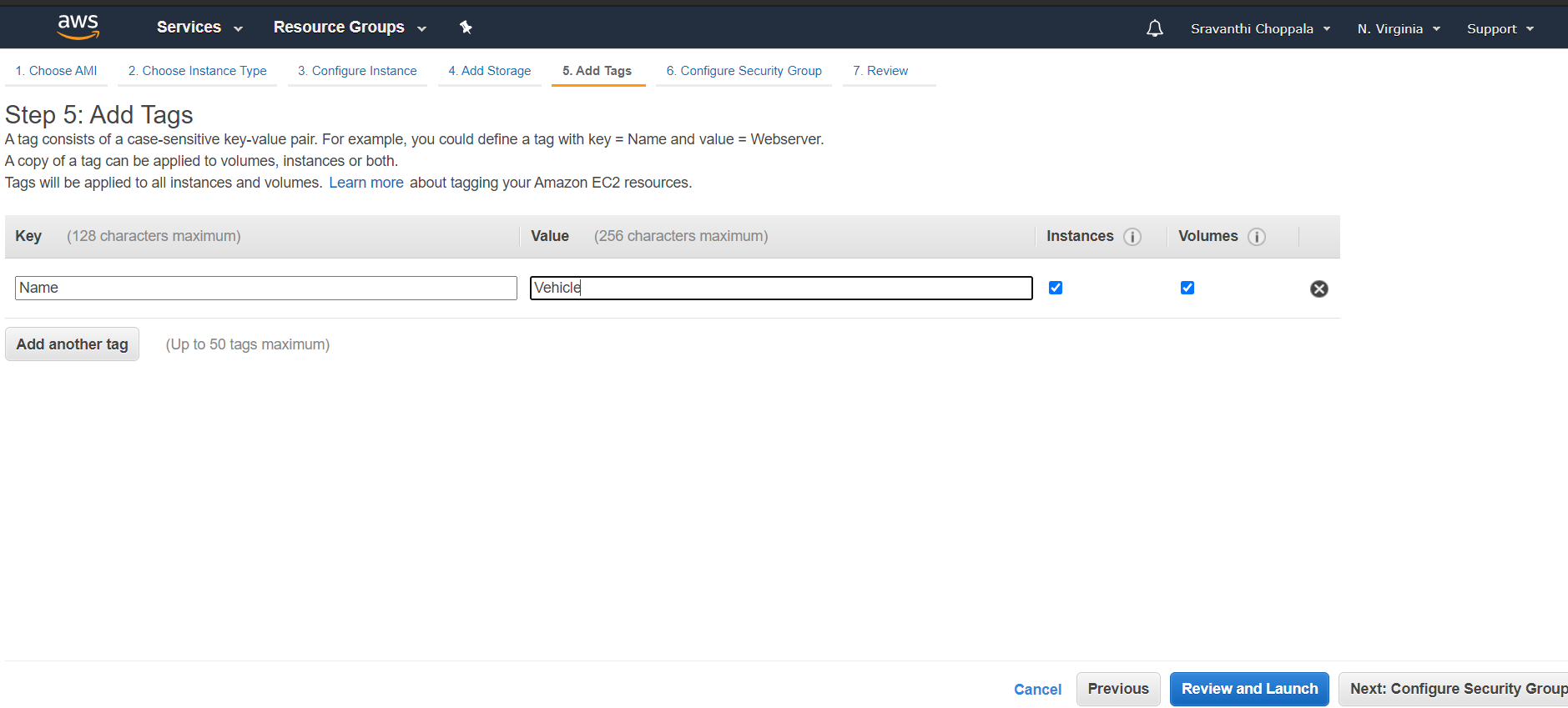
* Give Role Name which you want to give
* Enable Auto assign public IP in configuration instance details



* Add storage to instance



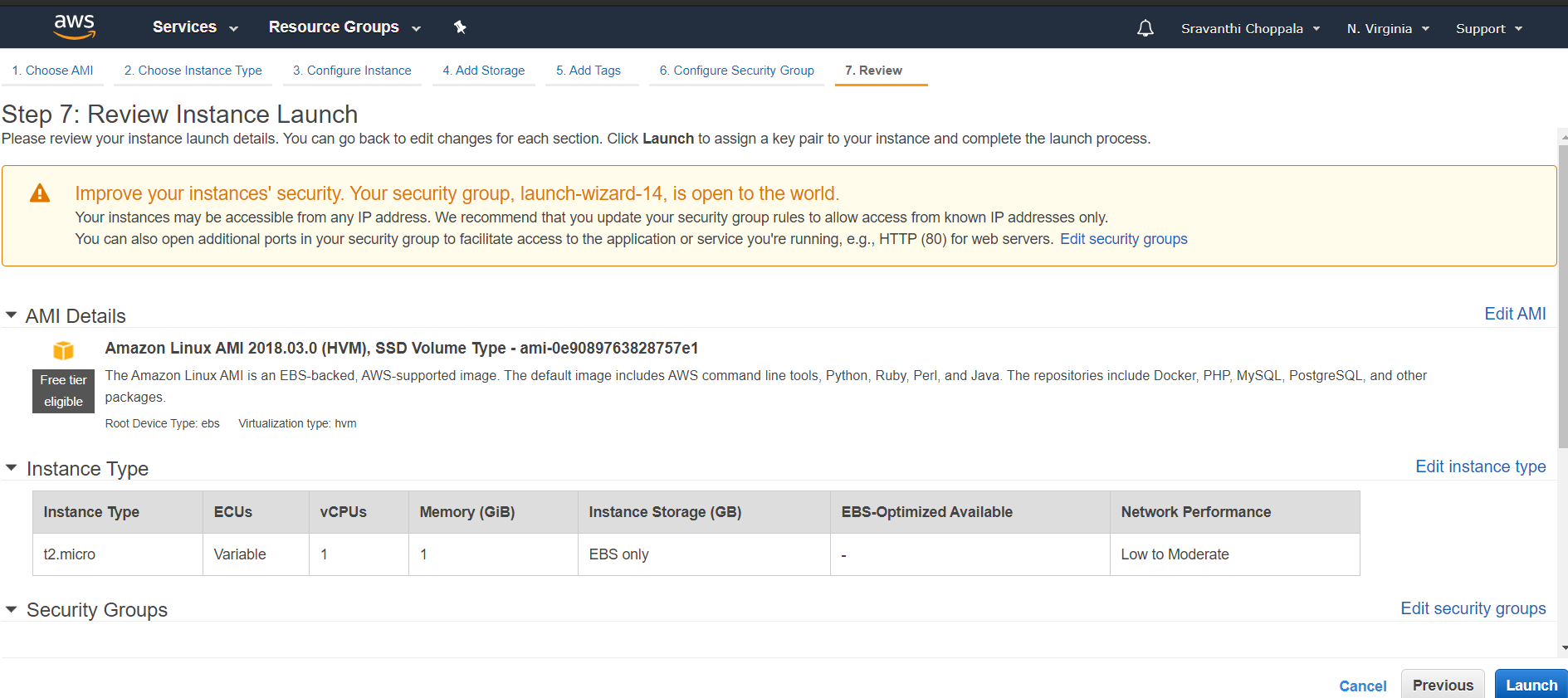
* Add tags and key pair value



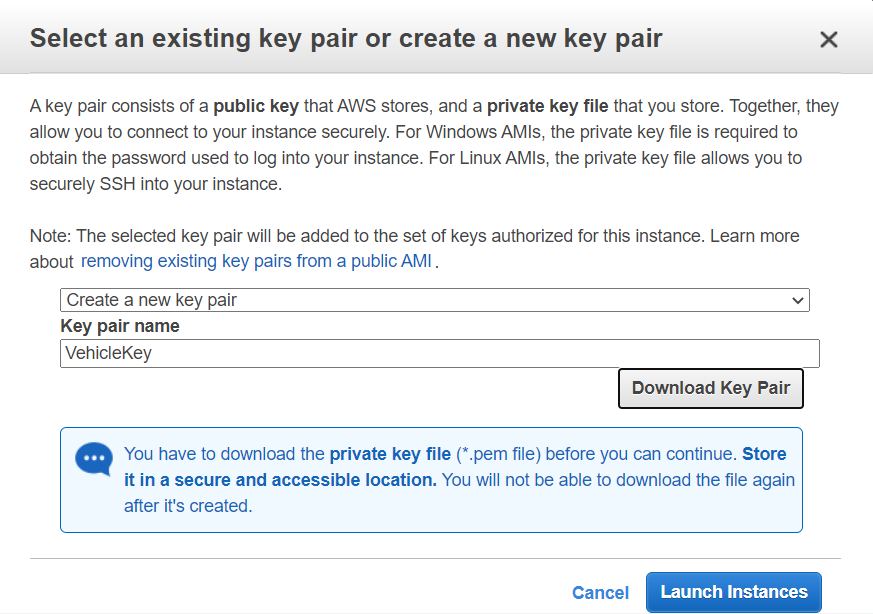
* Add a rule and give all traffic and provide custom as anywhere



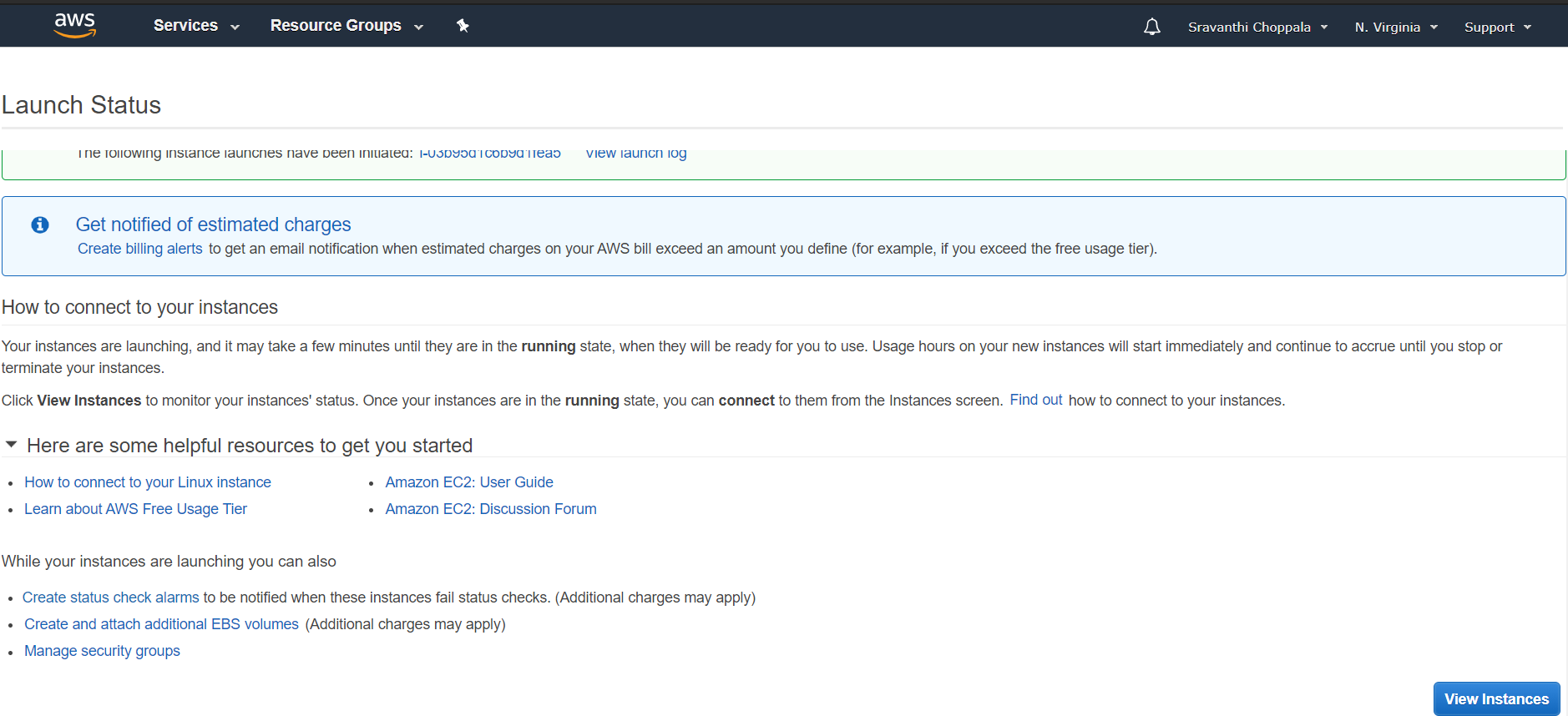
* Review and launch the instance



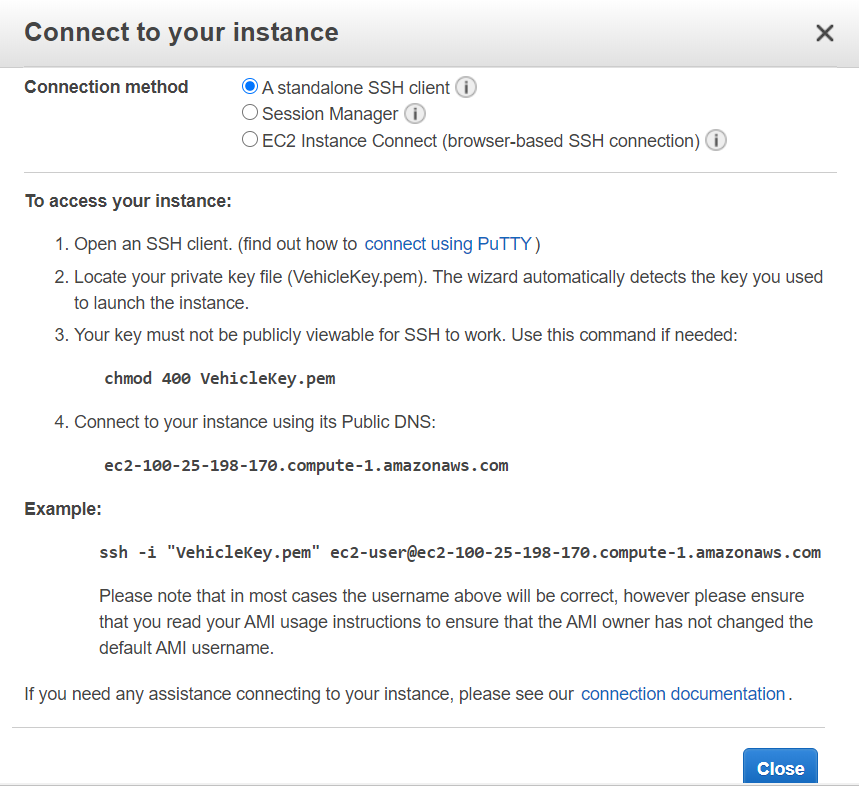
* Select Create a new key pair and download the keypair and launch instance



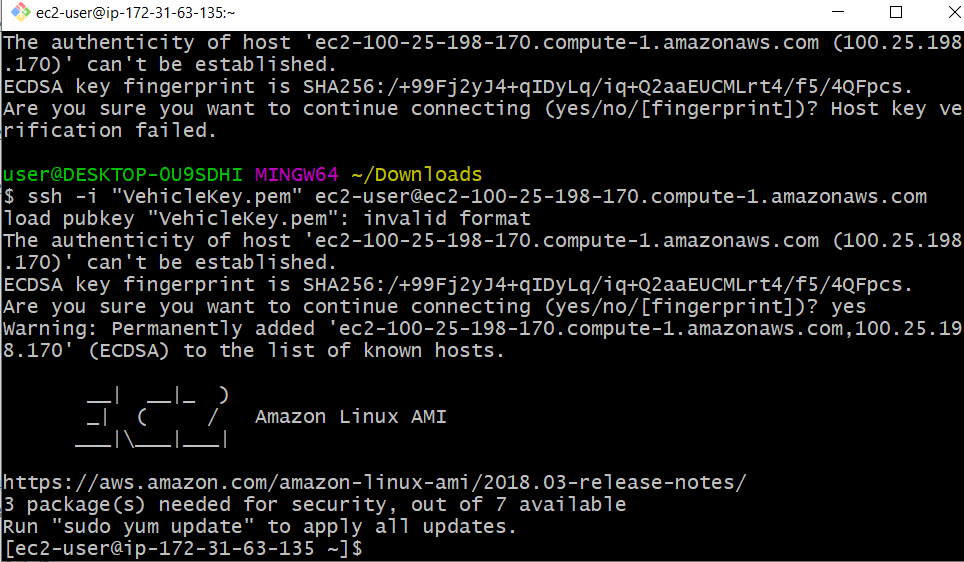
* Now instance is created and click on **View Instance**



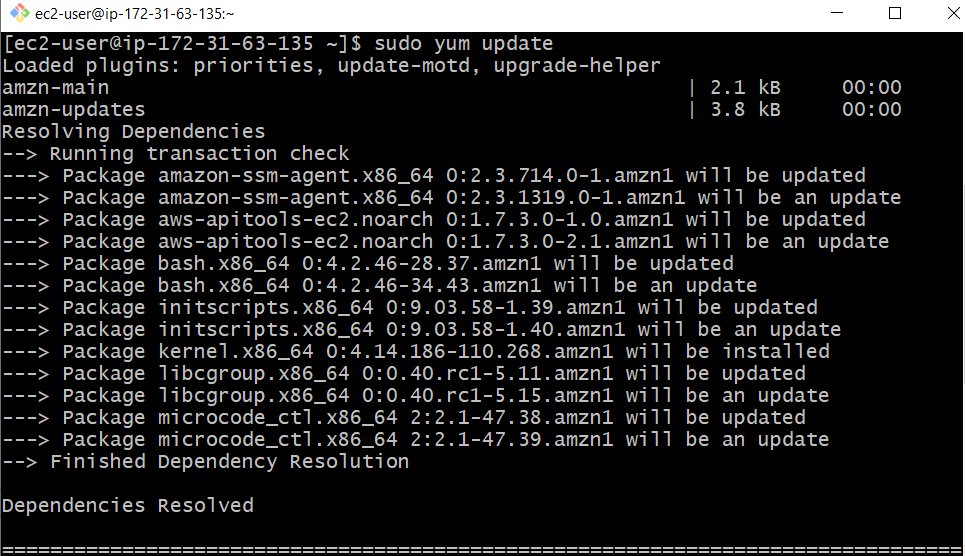
* Connect to instance and copy the ssh link



* Copy the ssh link in Git bash and connect to the Instance



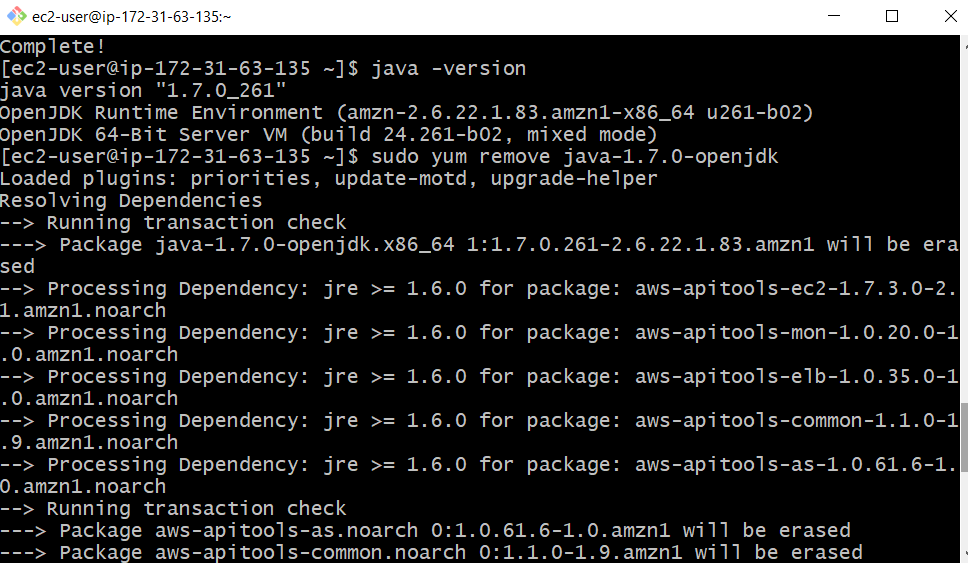
* Update the instance by giving sudo yum update



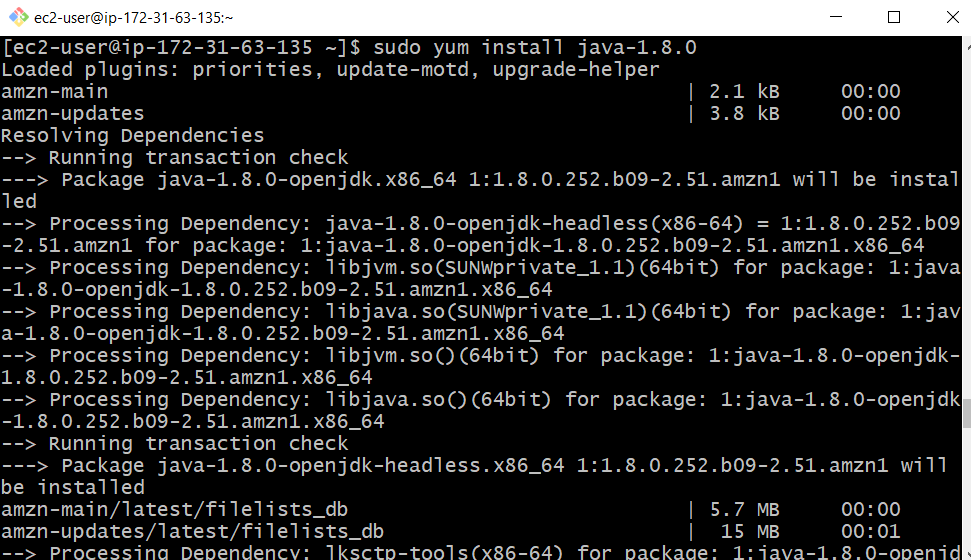
* Check java version and remove older version

To check java version🡪java –version

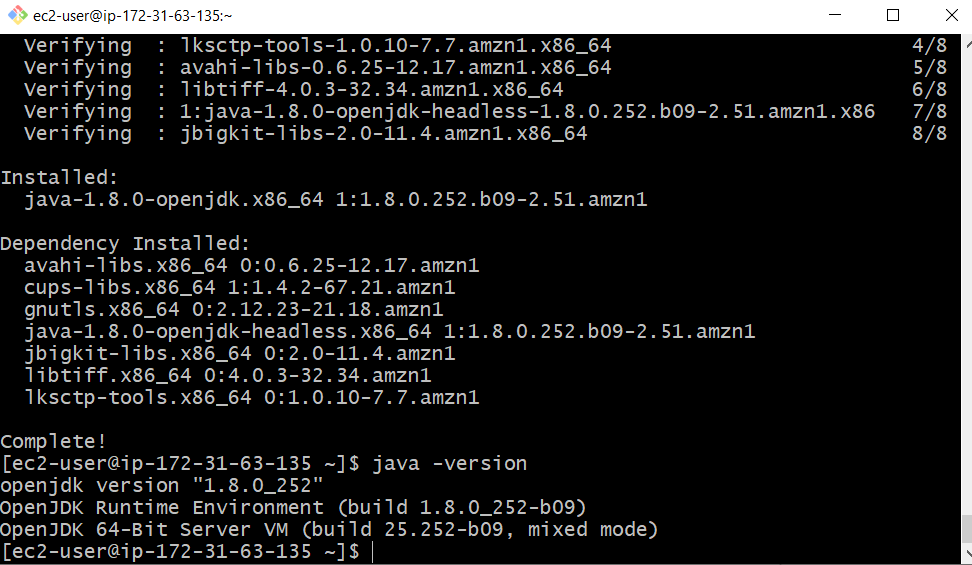
To remove old version🡪sudo yum remove java-1.7.0-openjdk



* Install java latest version sudo yum install java-1.8.0

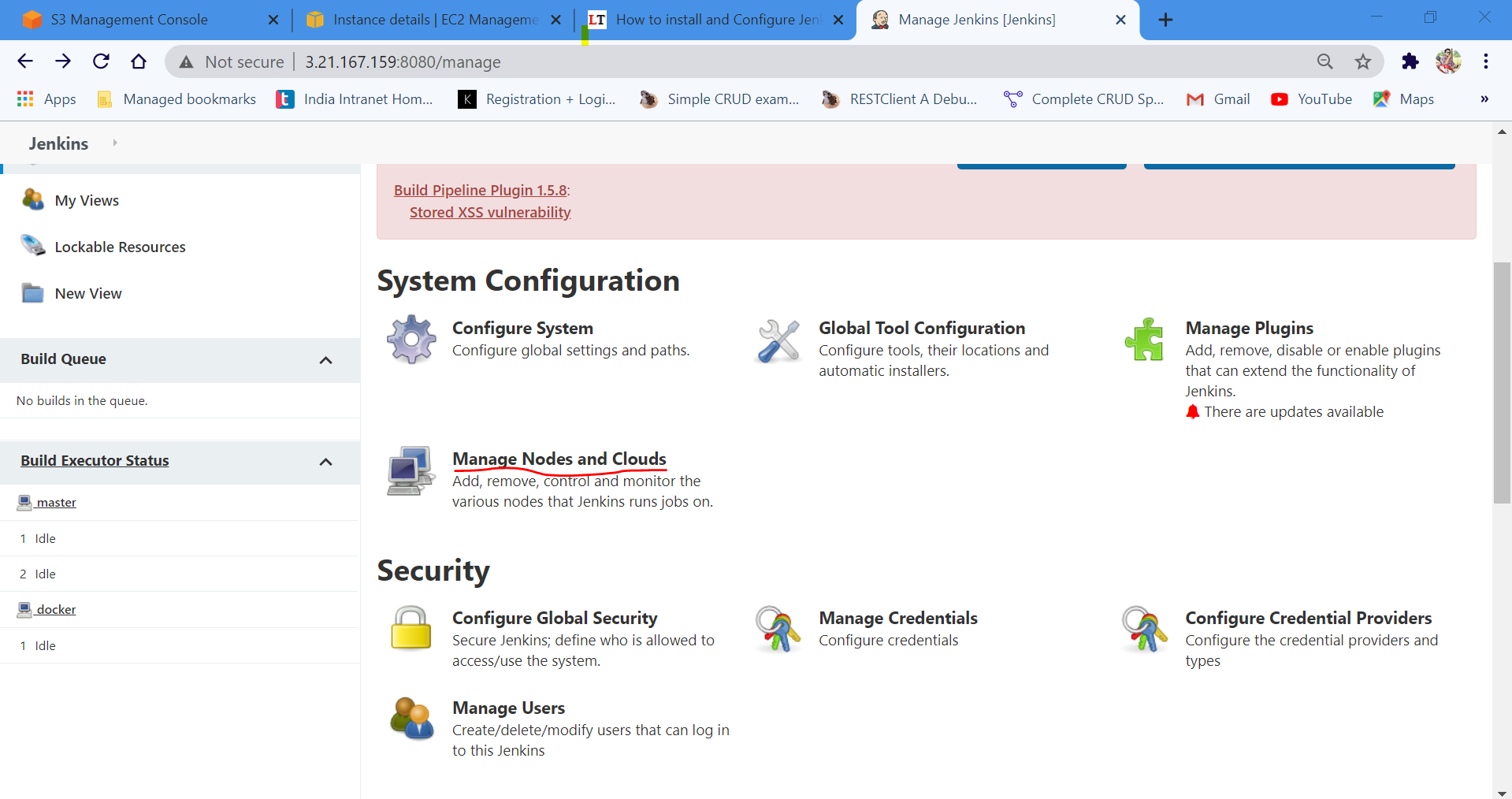


* You can check the version after the installation

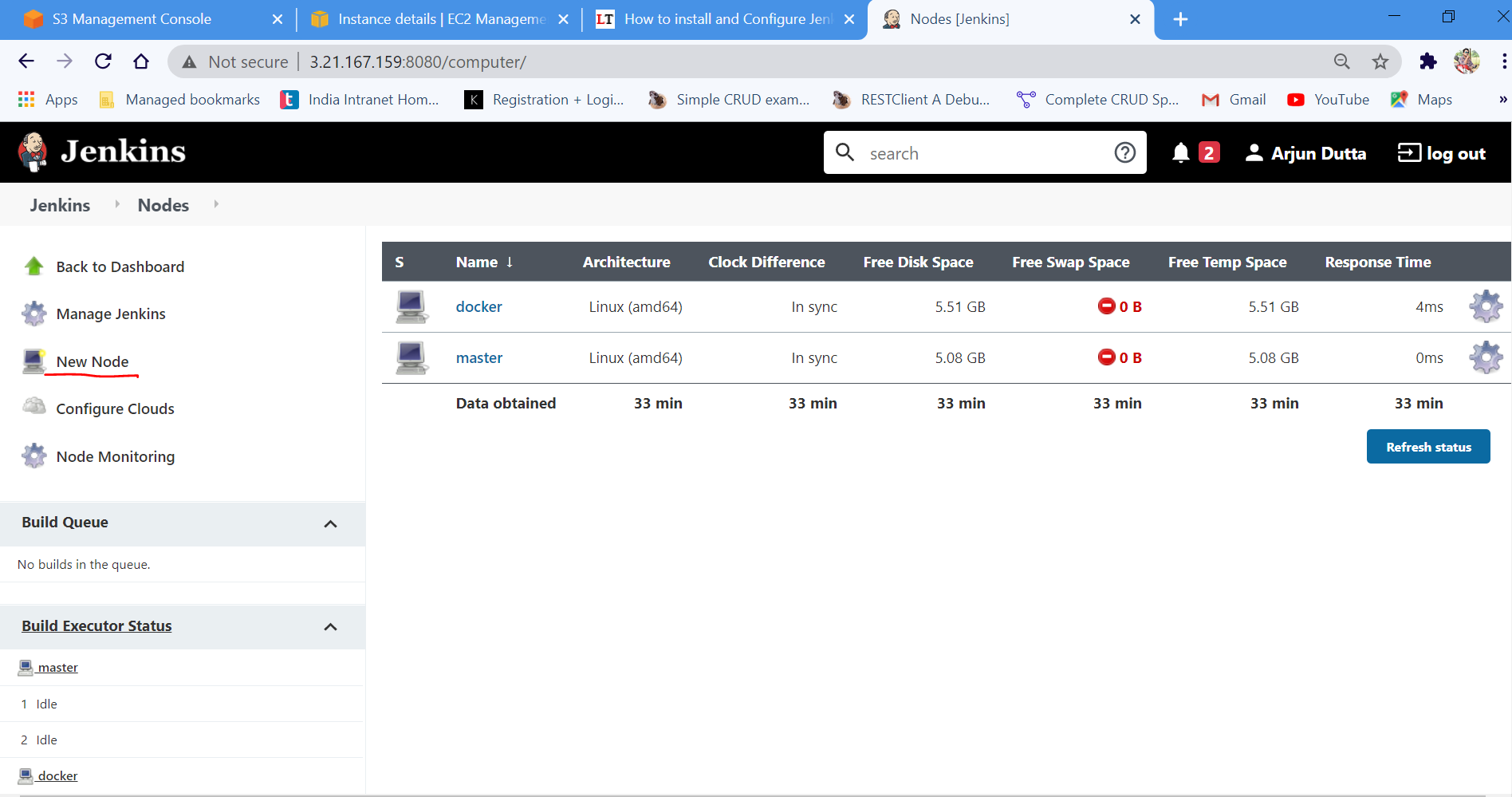


Node Creation:-

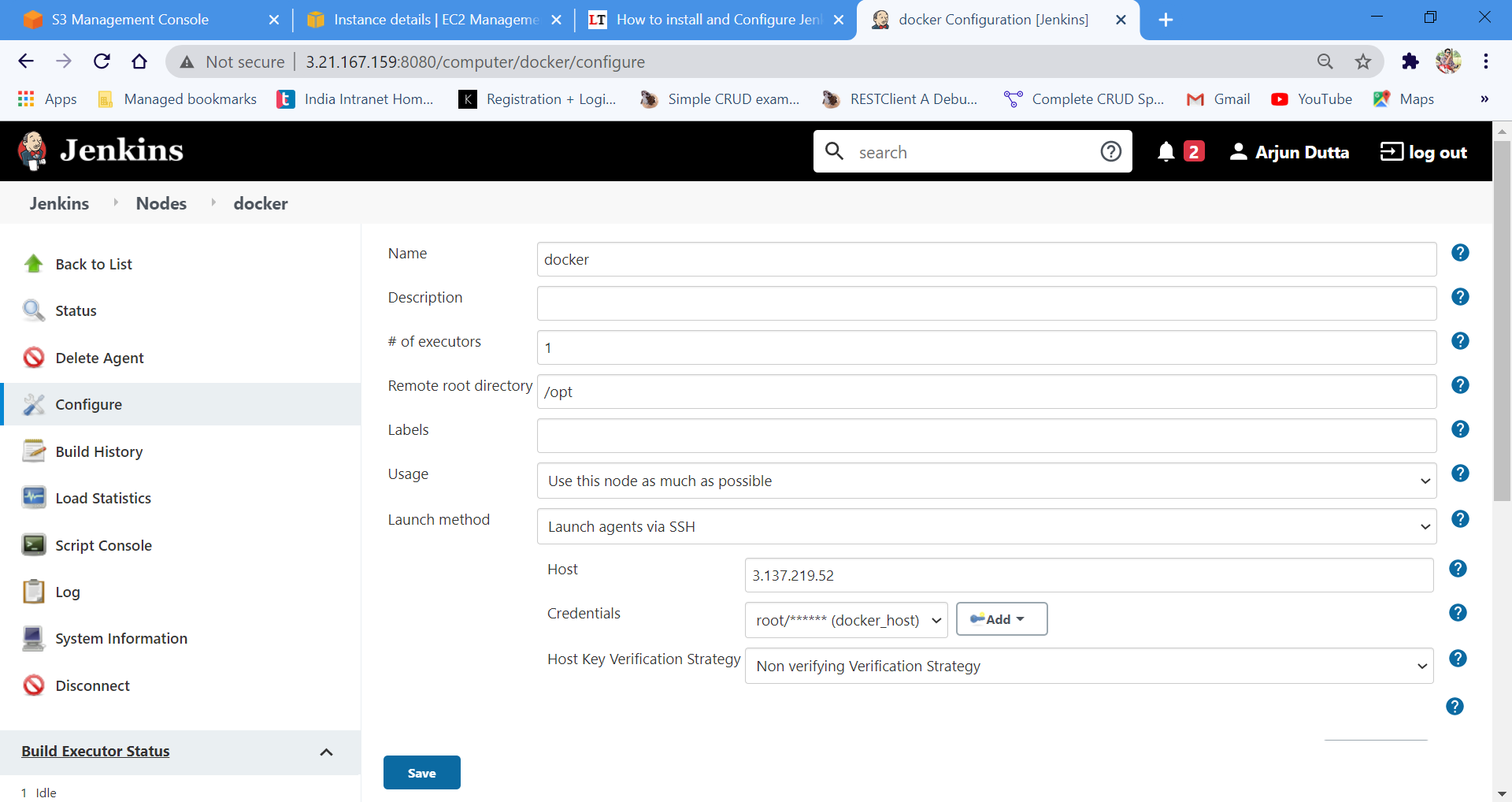
Step:-1



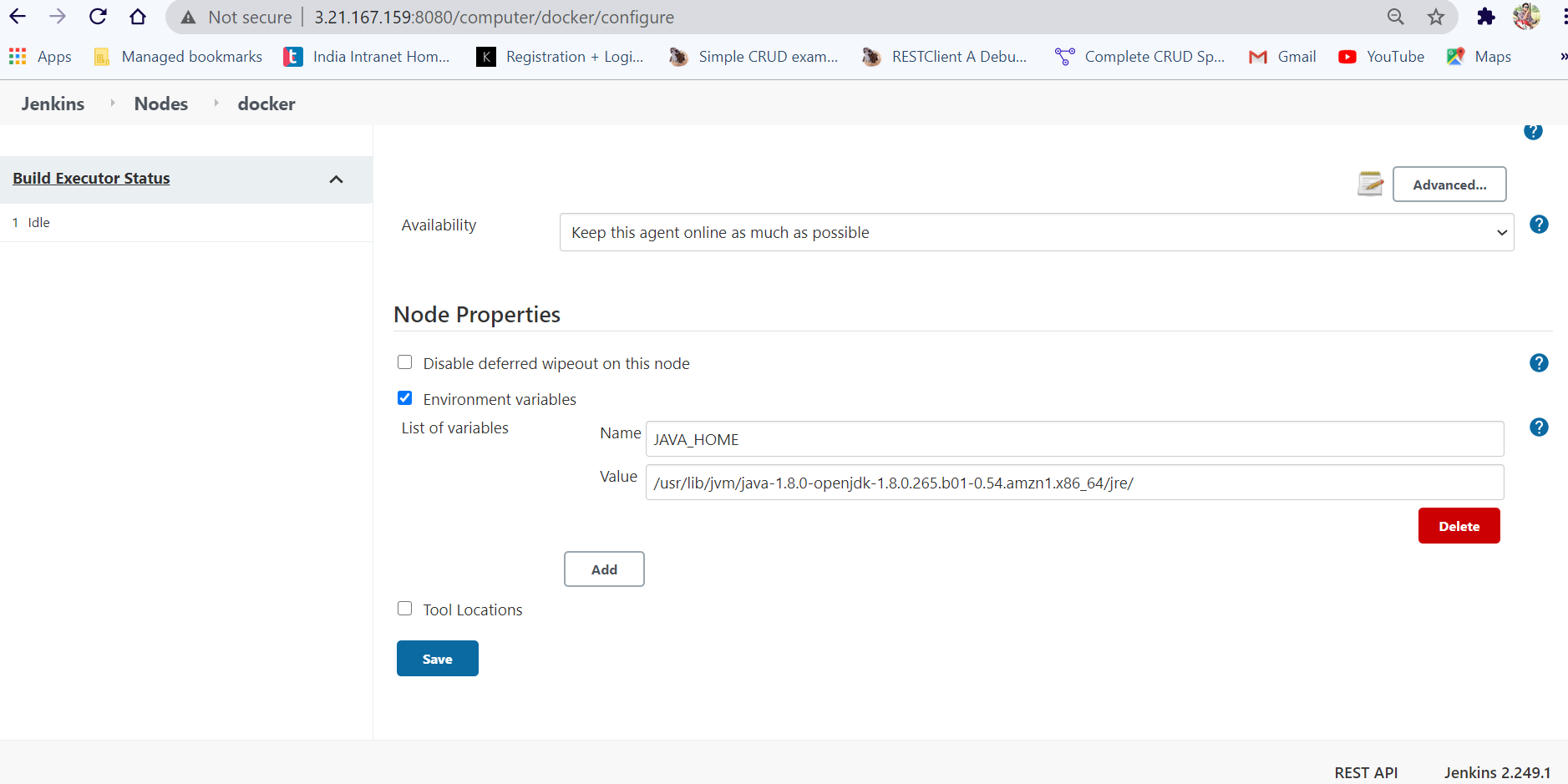
Step:-2



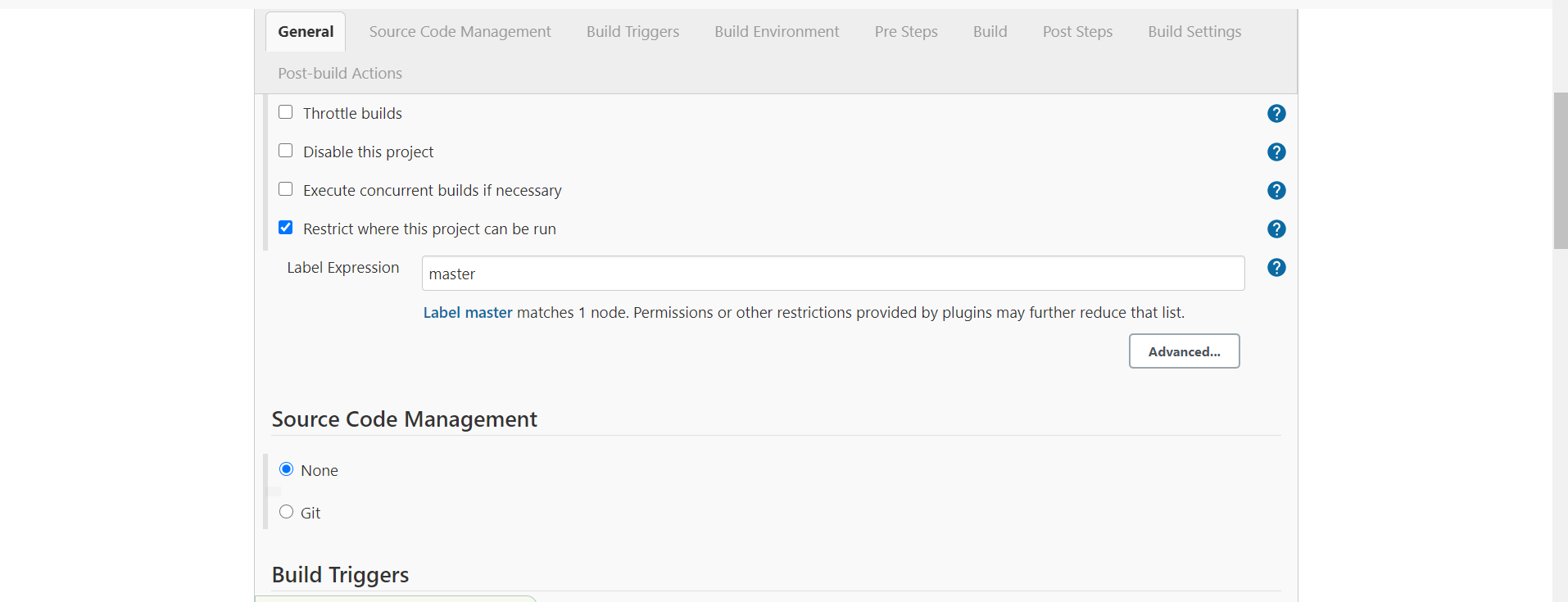
Step:3

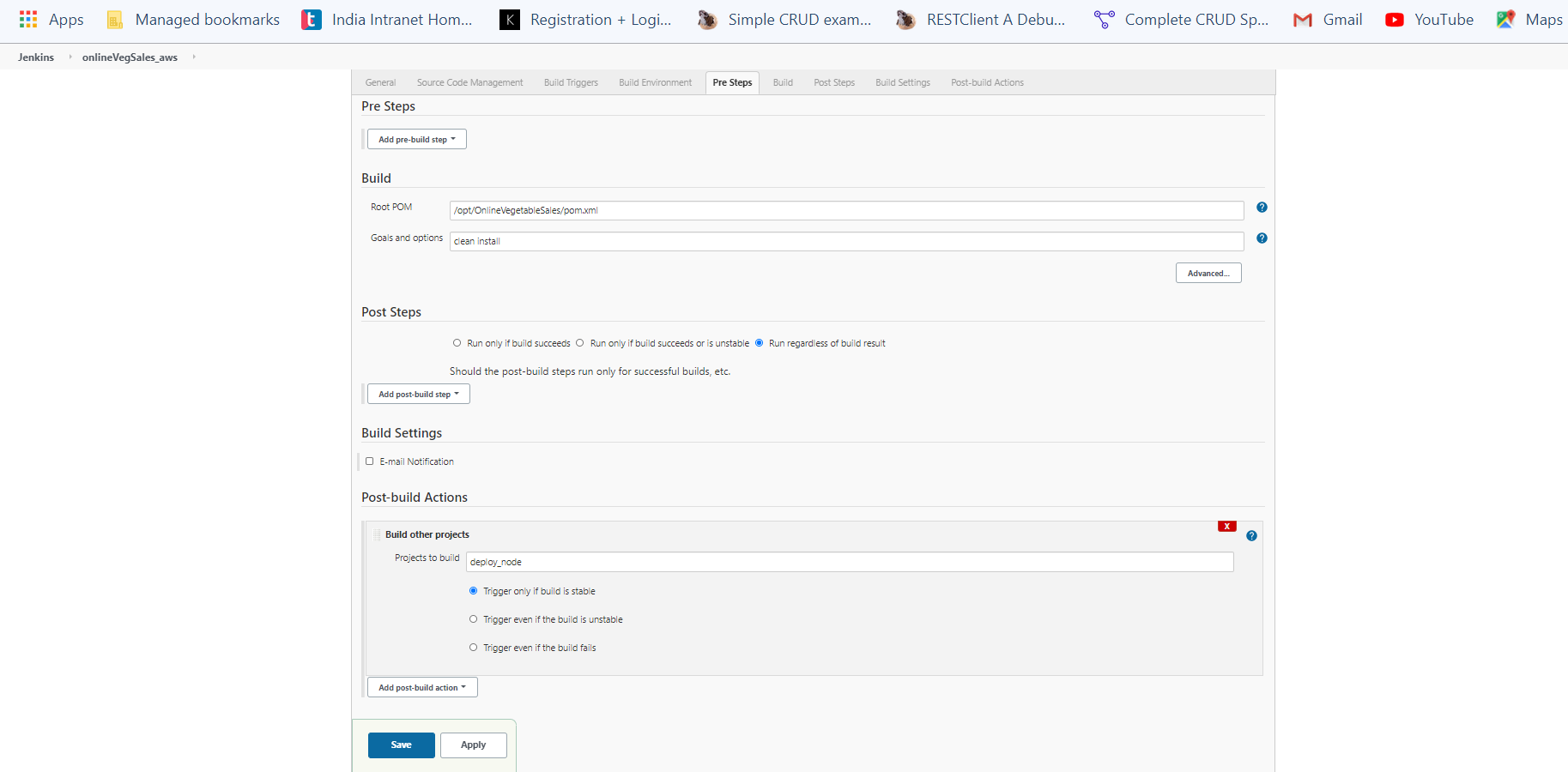


Step:4

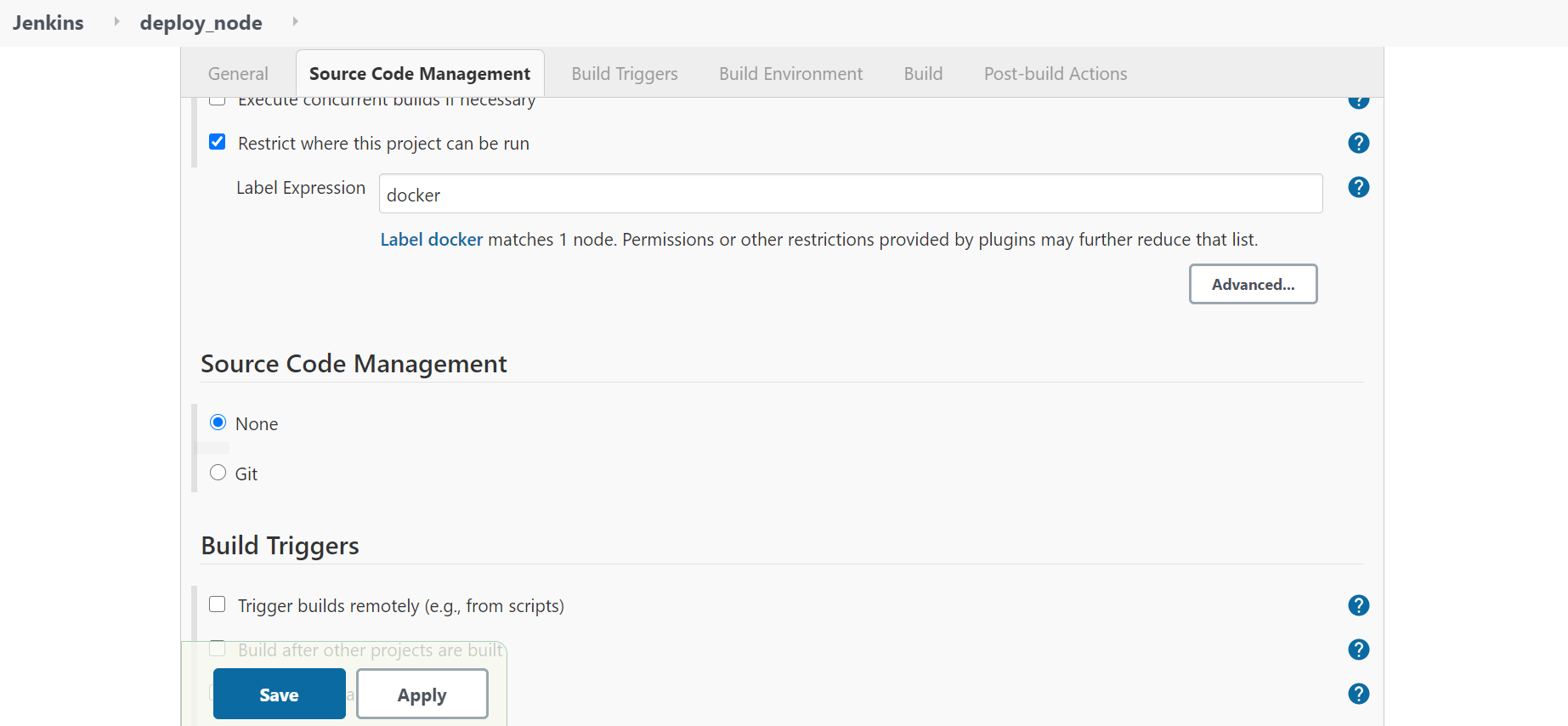


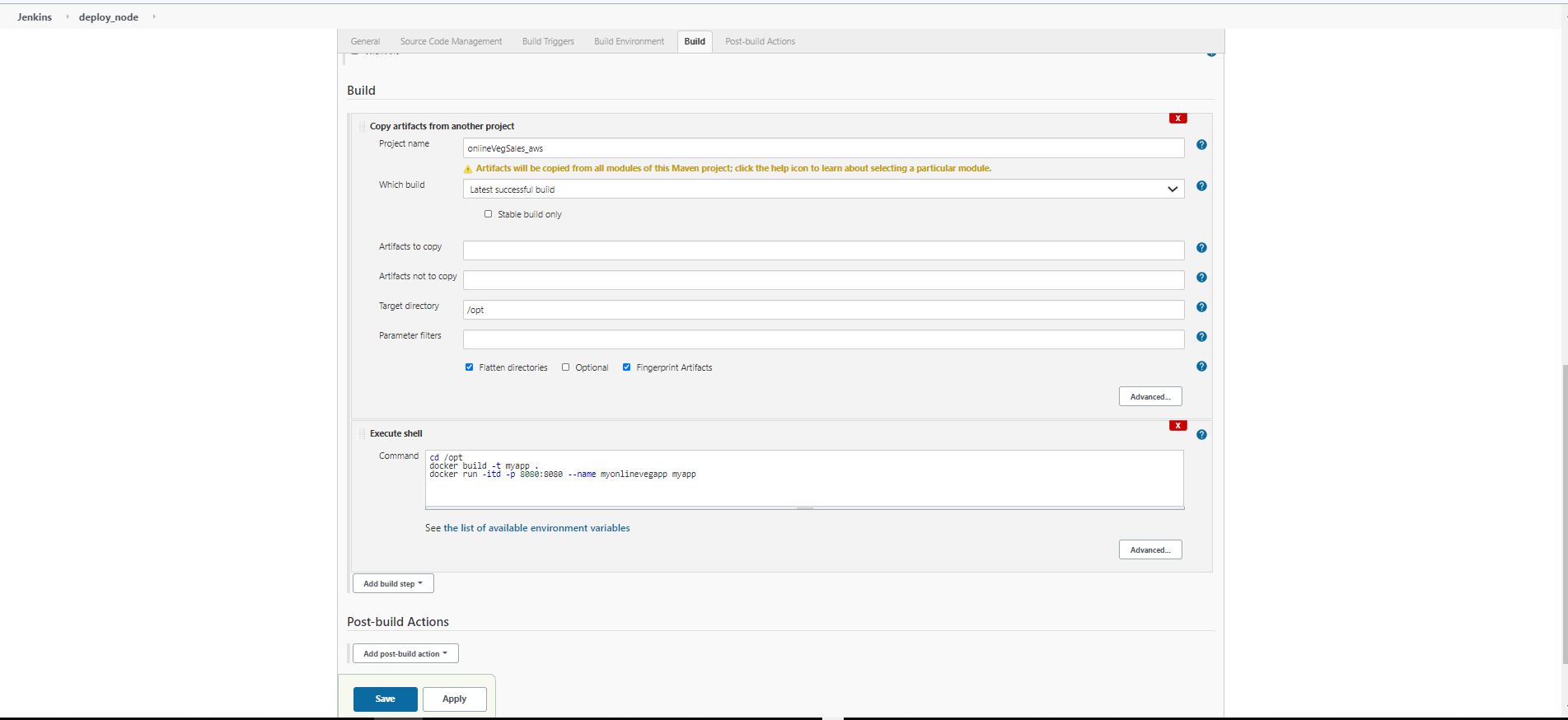
Maven Job (ShoppingAWS):





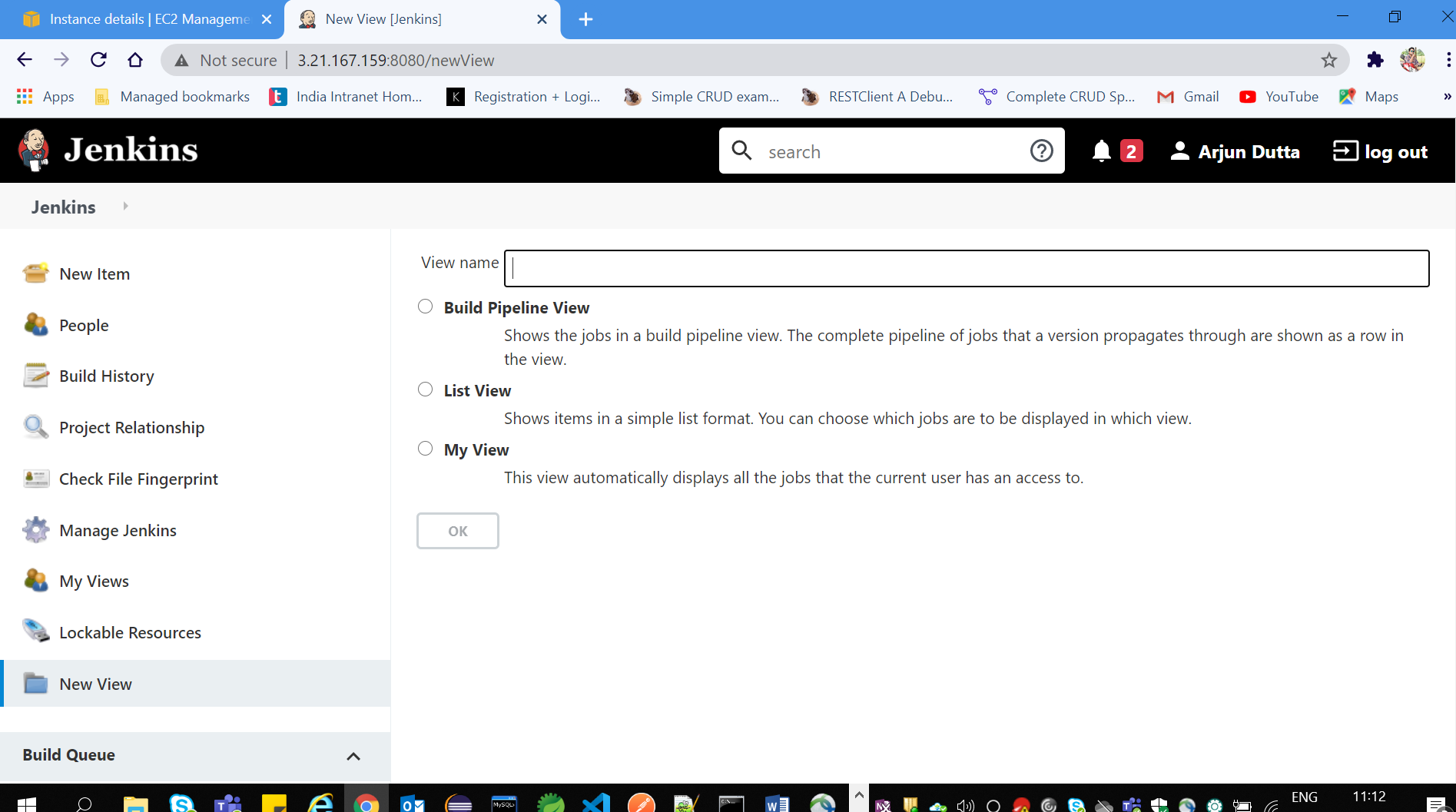
FreeStyle job (DeployNode):

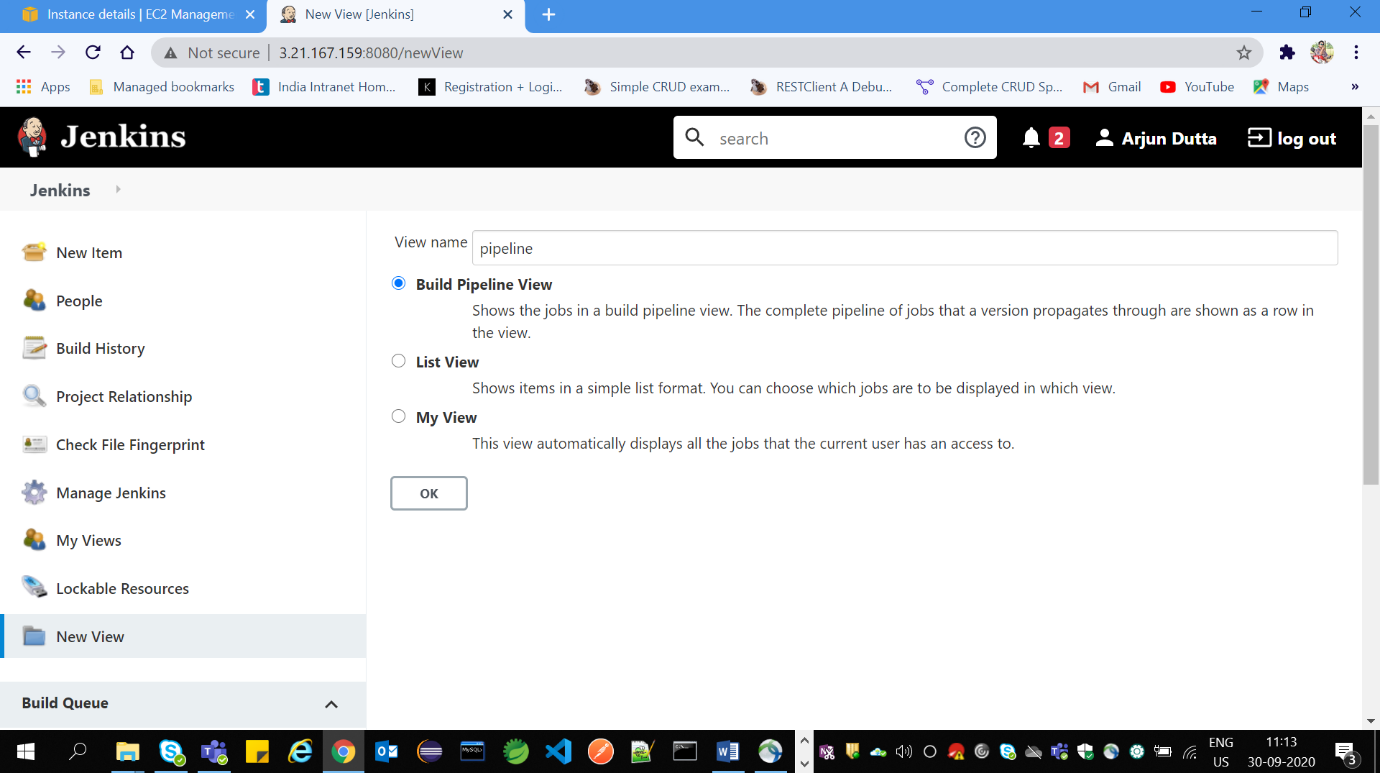


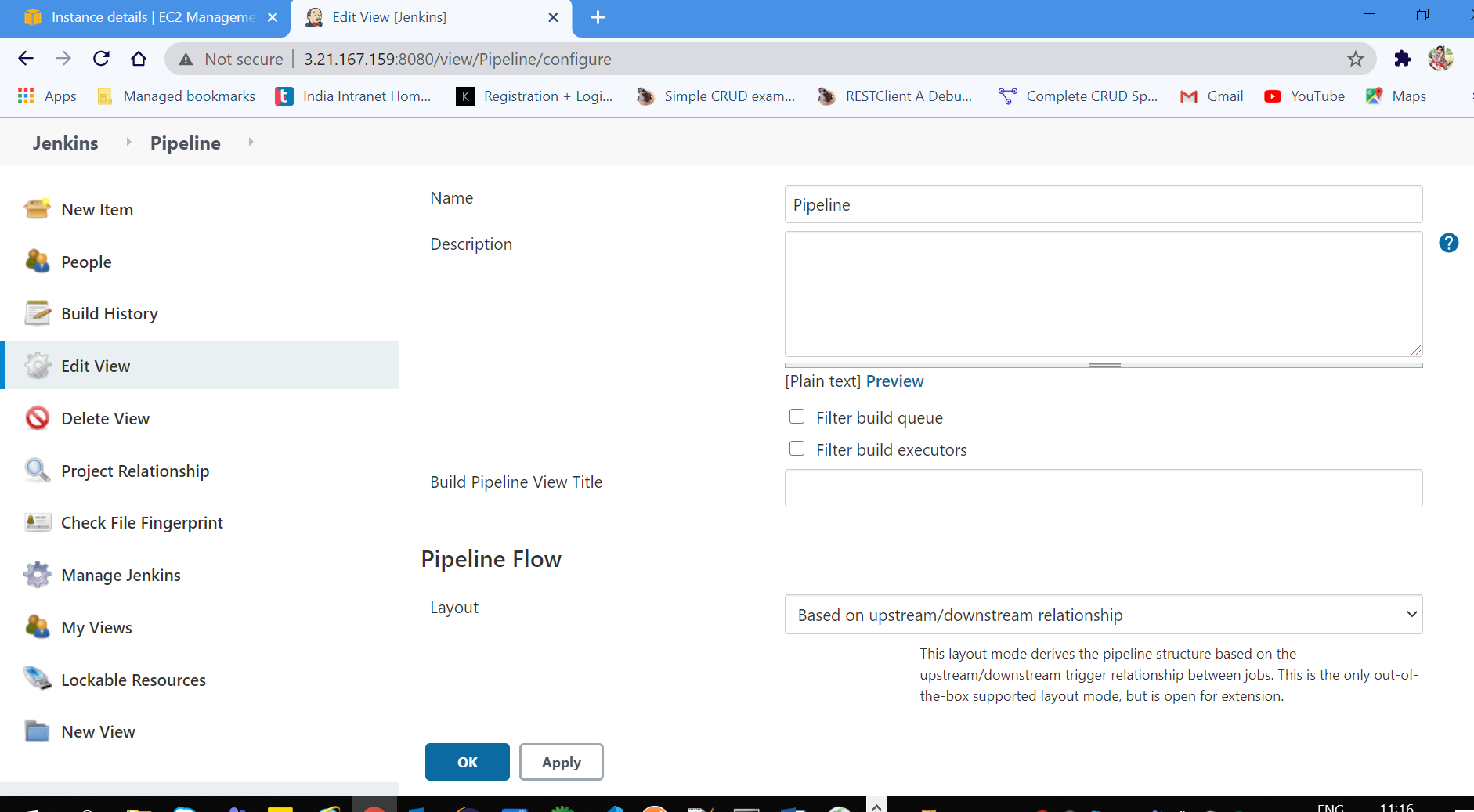


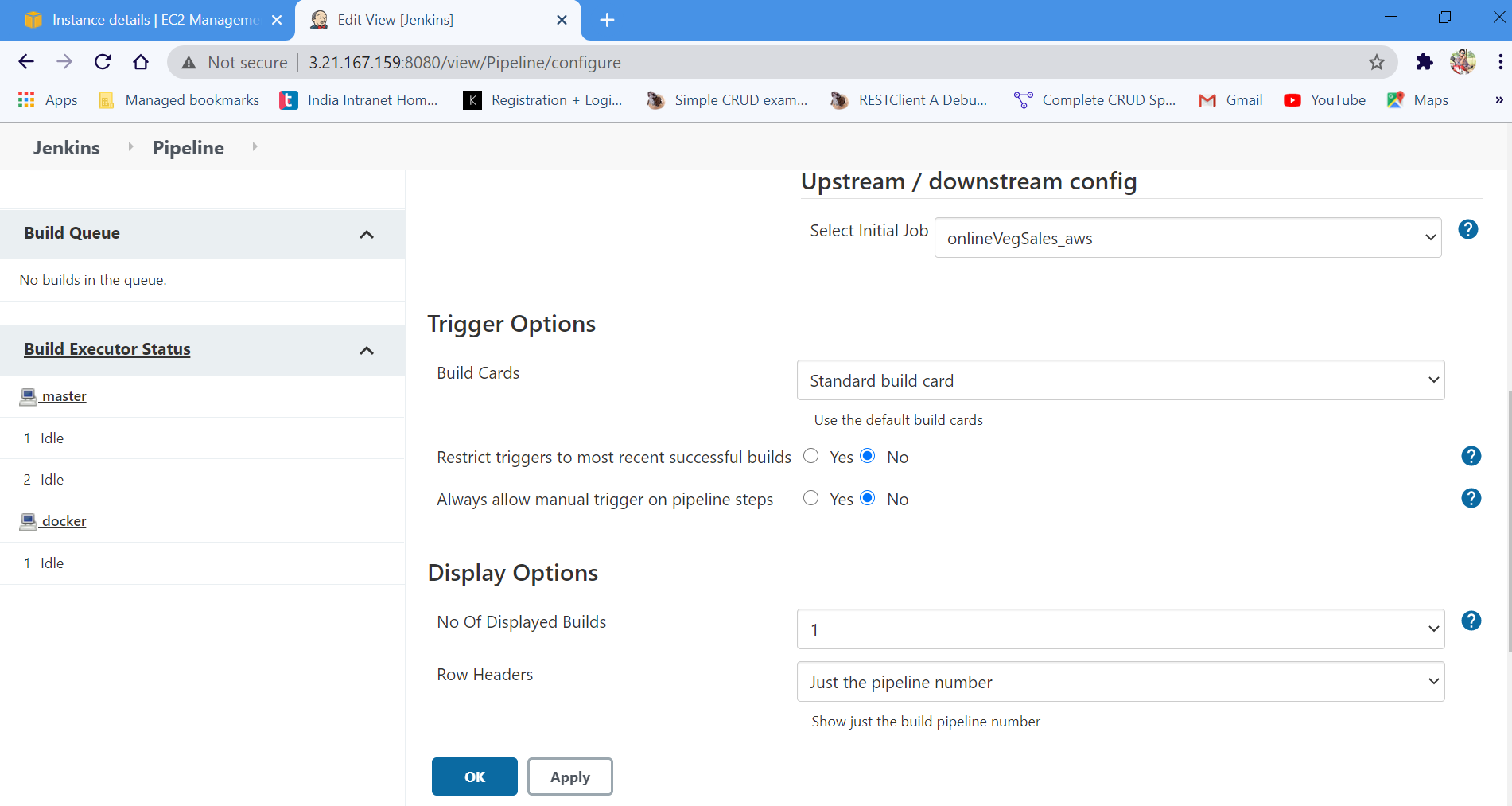
Pipeline:

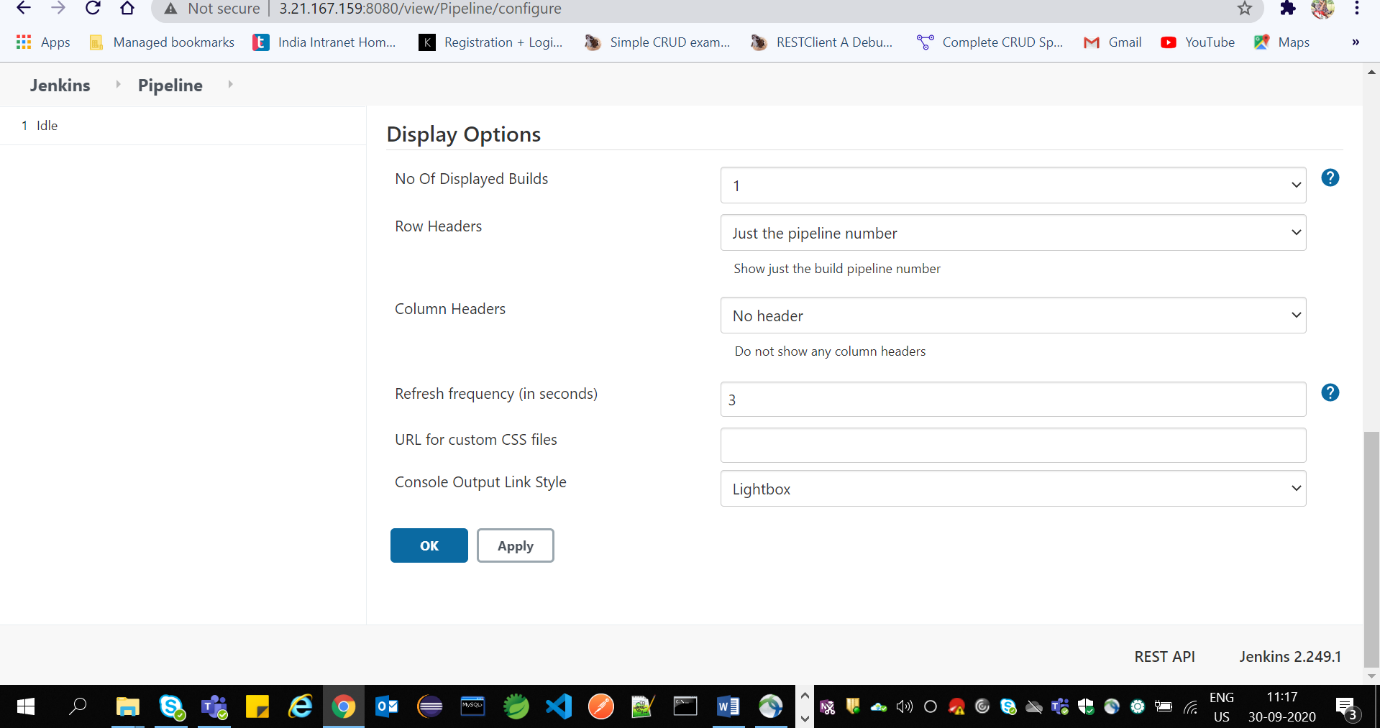
* Firstly download the ”**Build pipeline”** plugin from manage Jenkins.

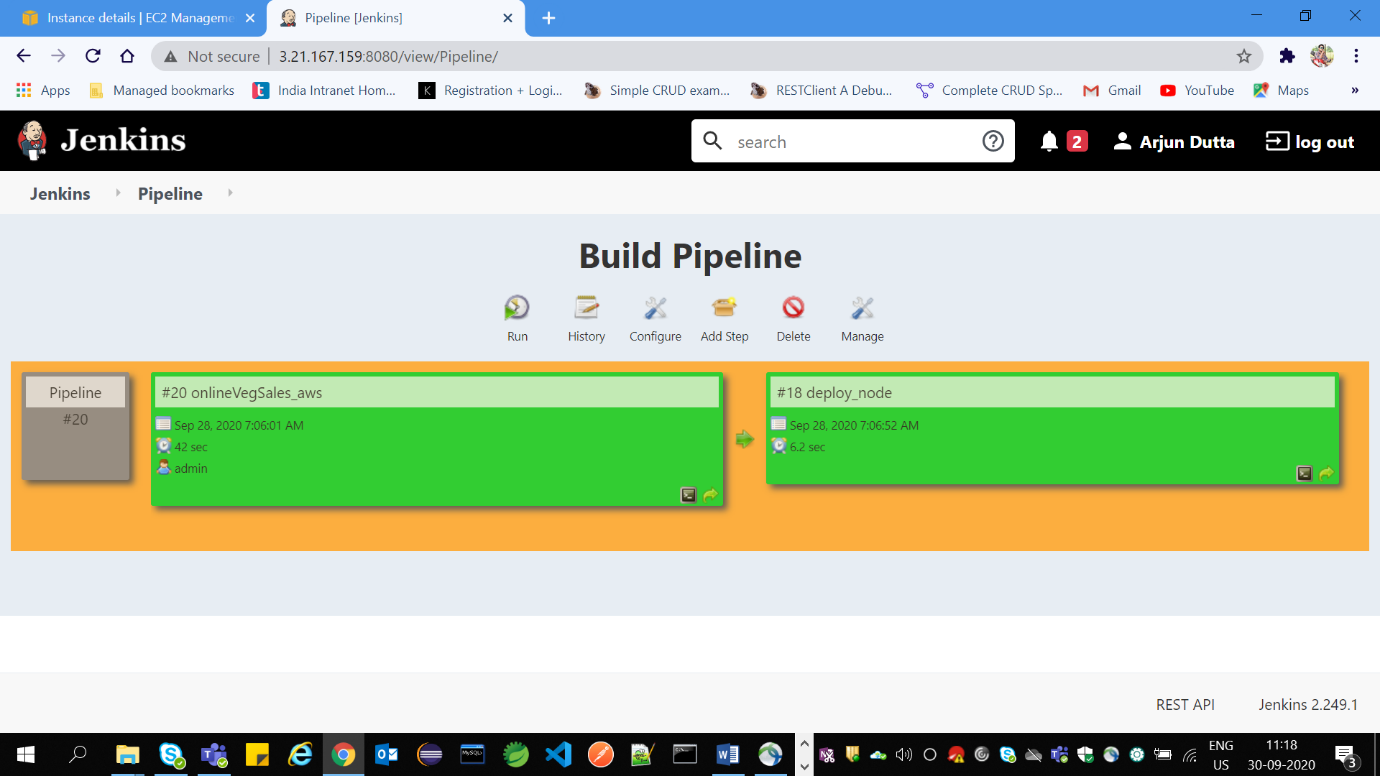




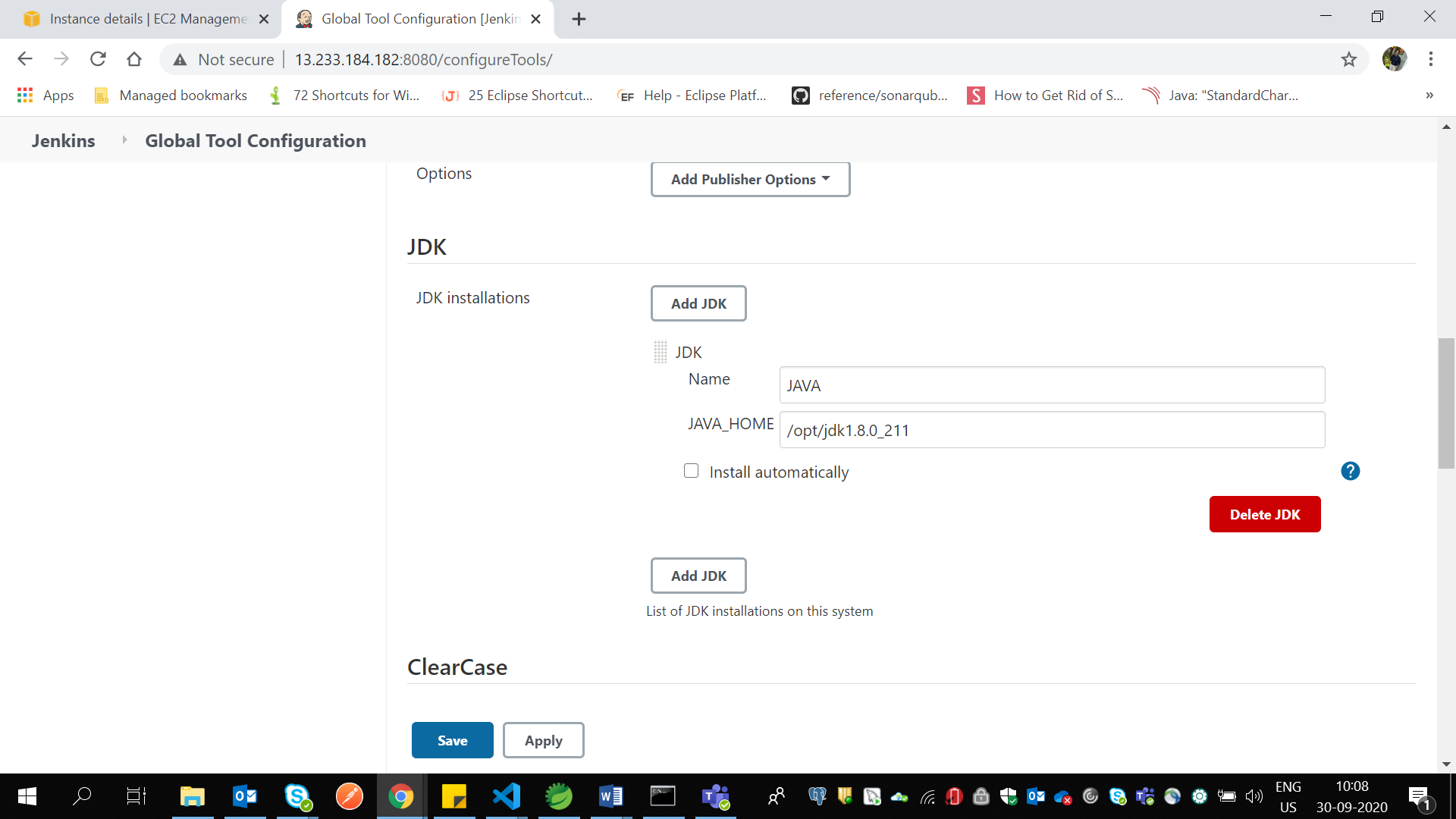


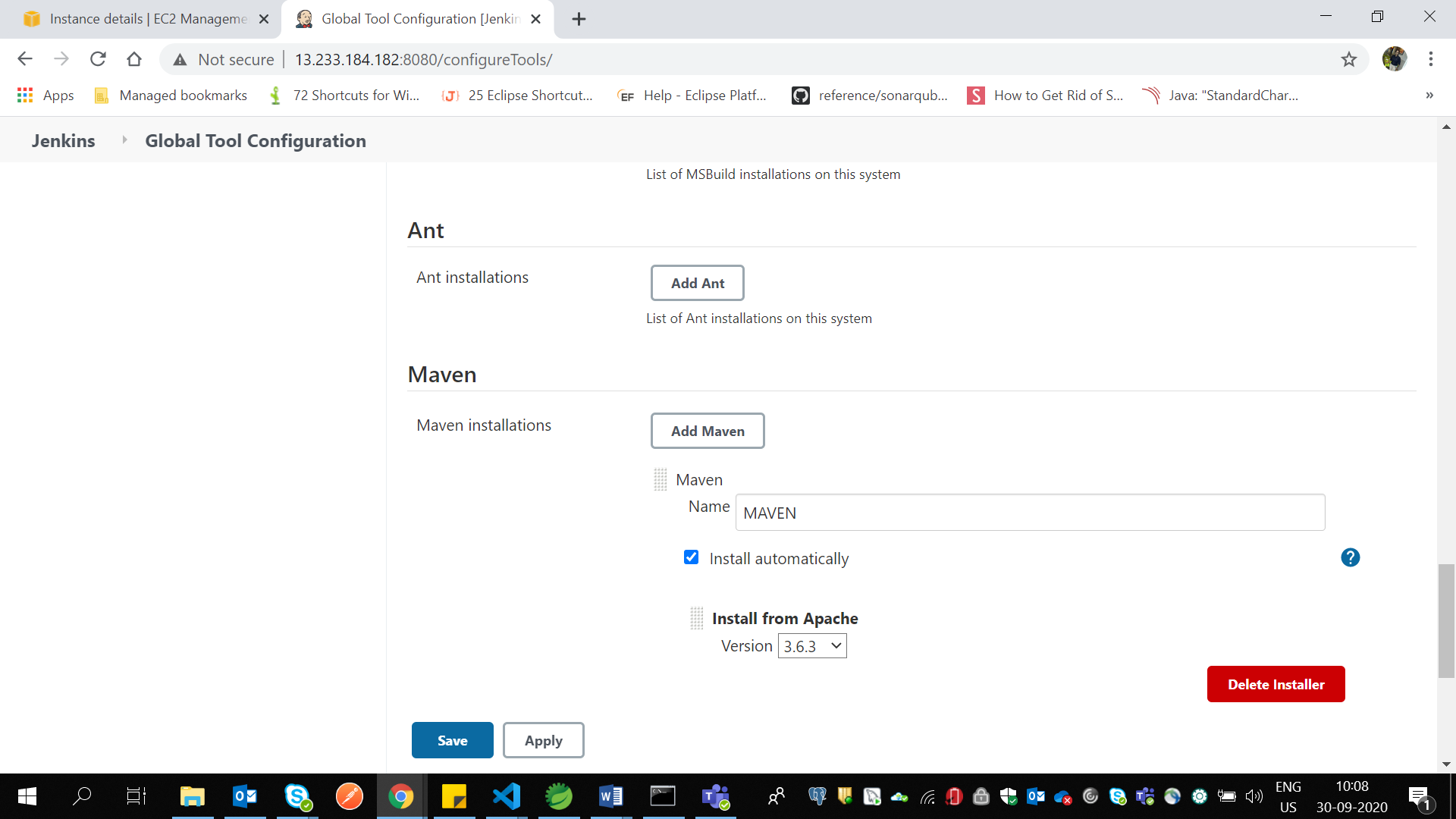






Jenkins Tool Config:





Amazon Linux 2 AMI(for mysql)

1. ssh -i "Shopping.pem" [ec2-user@ec2-13-233-107-194.ap-south-1.compute.amazonaws.com](mailto:ec2-user@ec2-13-233-107-194.ap-south-1.compute.amazonaws.com)
2. sudo su
3. wget <https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm>
4. md5sum mysql57-community-release-el7-9.noarch.rpm
5. sudo rpm -ivh mysql57-community-release-el7-9.noarch.rpm
6. sudo yum install mysql-server
7. sudo systemctl start mysqld
8. sudo systemctl status mysqld
9. sudo grep 'temporary password' /var/log/mysqld.log
10. mysql -u root –p(to get into sql prompt)
11. SET PASSWORD = PASSWORD('Arjun@123');
12. create database proj;
13. show tables;
14. GRANT ALL ON proj.\* TO root@13.233.184.182 IDENTIFIED BY 'Arjun@123';(for jenkins instance IP)
15. GRANT ALL ON proj.\* TO root@3.6.126.23 IDENTIFIED BY 'Arjun@123'; (for docker instance IP)
16. show tables;

Amazon Linux AMI (jenkins and jdk)

1. Install the jdk and Jenkins as mentioned.
2. <https://www.linuxtechi.com/install-configure-jenkins-on-centos-7-rhel-7/>
3. sudo yum install java-1.8.0-openjdk
4. sudo update-alternatives --config java
5. Type “java -version”.
6. Service Jenkins start
7. .application properties need to change the password and the localhost to sql server
8. Cd /opt
9. Chmod –R 777 shopping
10. In local project outside folder cmd->(Scp –r projectname root@jenkinsserver:/opt)
11. Go to java tar.gz where it is downloaded and open cmd from there🡪( Scp –r jdk427linux.tar.gz root@jenkinsserver:/opt)
12. Chmod –R 777 \* (inside /opt)
13. Service Jenkins restart
14. sudo update-alternatives --config java(For sleecting the java version)

Amazon Linux AMI (docker and jdk)

1. sudo vi /etc/ssh/sshd\_config  
   PasswordAuthentication yes  
   PermitRootLogin yes  
   sudo passwd root  
   sudo service sshd restart
2. cd /opt
3. Type “sudo yum update”. Then type y.
4. Type “sudo yum install docker”. Then type y.
5. Type “sudo usermod -aG docker $USER”.
6. Type “sudo service docker start”.
7. Dockerfile creation: Inside /opt ->vi Dockerfile->Insert mode->
8. FROM tomcat:8.0-alpine
9. COPY \*.war /usr/local/tomcat/webapps/
10. EXPOSE 8080
11. CMD ["catalina.sh", "run"]
12. Esc->Shift+:->wq
13. Chmod –R 777 Dockerfile
14. Go to /opt and type docker ps
15. Docker ps after deploying
16. To stop the container: docker stop <container id>
17. To remove the application container: docker rm <container id>