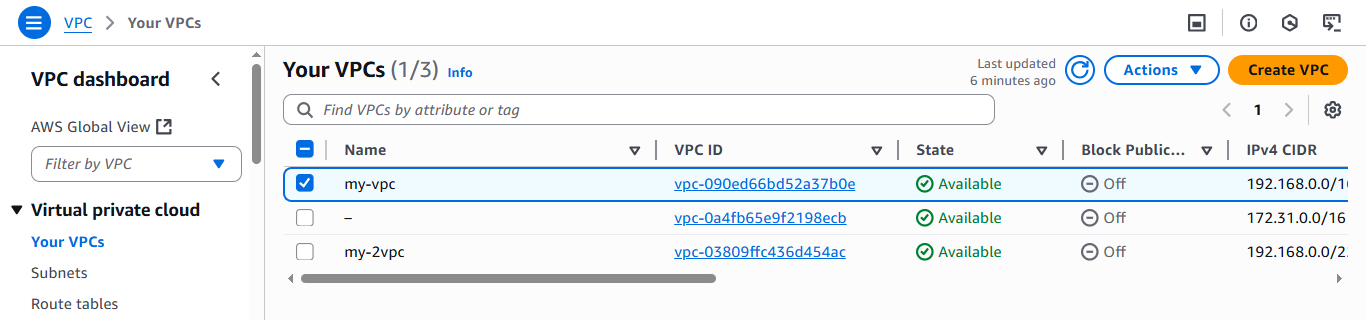
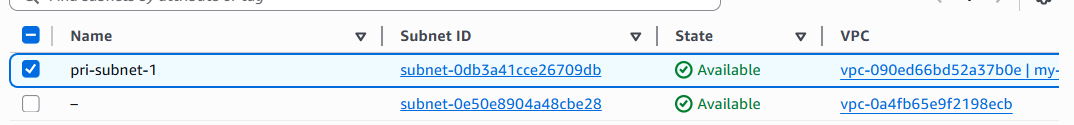
# VPC first vdo tasks

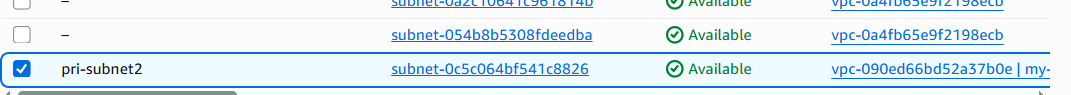
1. **Create VPC with 2 private and 2 public subnets.**

* Create vpcnew for open aws es2 instance go the search bar and type vpc.
* Then selecte the vpc nd create a new vpc==name nd gave ip cidr number:192.168.0.0/20 for vpc nd create.
* Then after that there is a bar in left side and create subnets





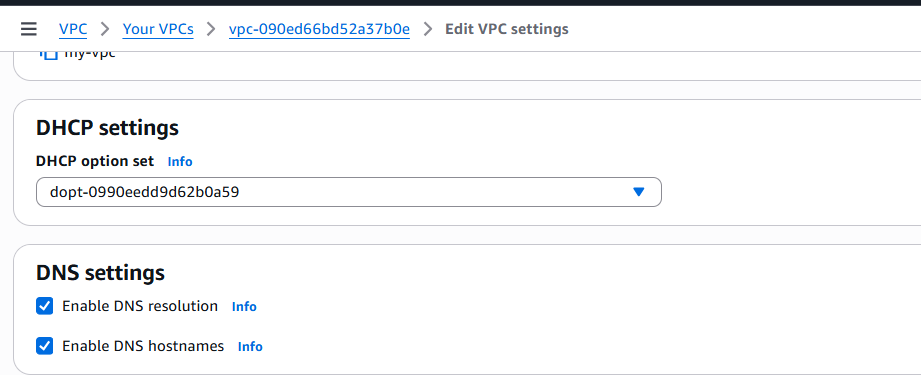




* Then after that there is a bar in left side and create subnets.
* In that subnets cidr number is in a range.
* And create 2 public&private subnets.

1. **Enable DNS Hostname in VPC.**

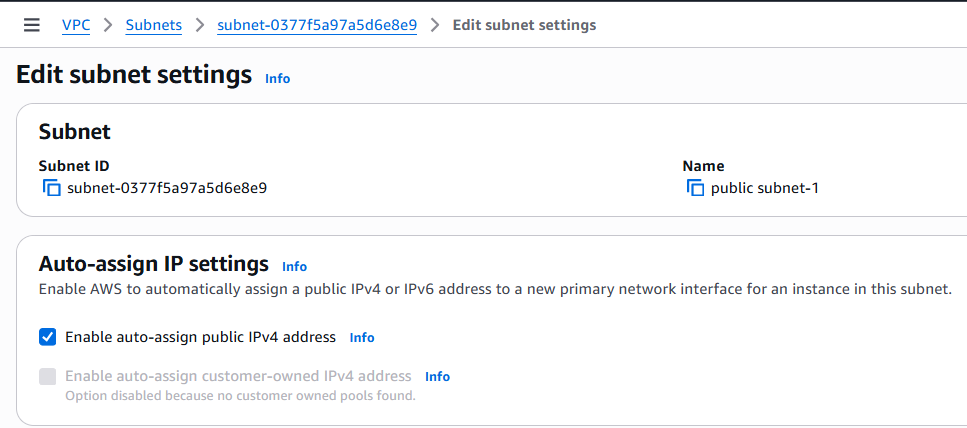
* Open ec2 console
* And go to the search bar and type it vpc
* Enter in your vpc which want uh modified.
* Then edit vpc and enable the “**DNS host name”.**



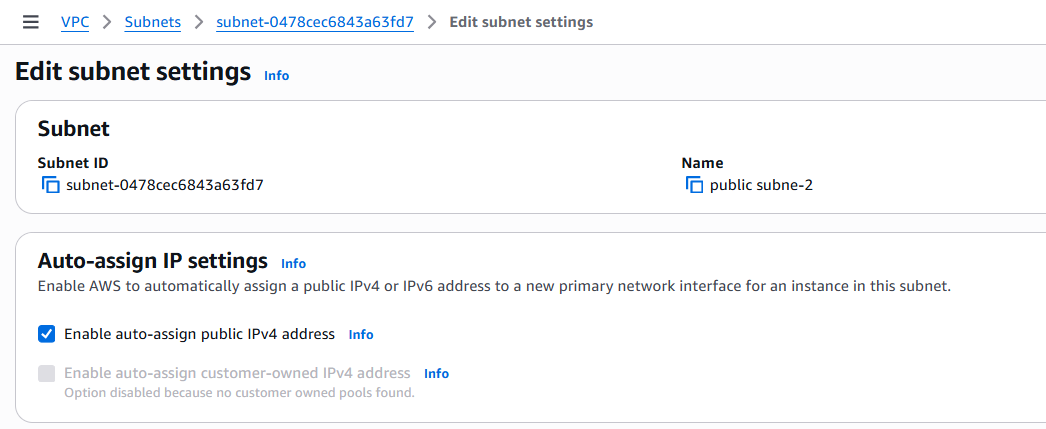
1. **Enable Auto Assign Public IP in 2 public subnets.**

* Open ec2 console.
* Then go the search bar in vpc.
* Then the left coloumn there is subnet option.
* Entering in the subnets which we are created in previously in task 1.
* Select the public subnet1 and edit .
* And **enable Auto assign in public ips.**
* Here the results are.

**Public subnet 1:**

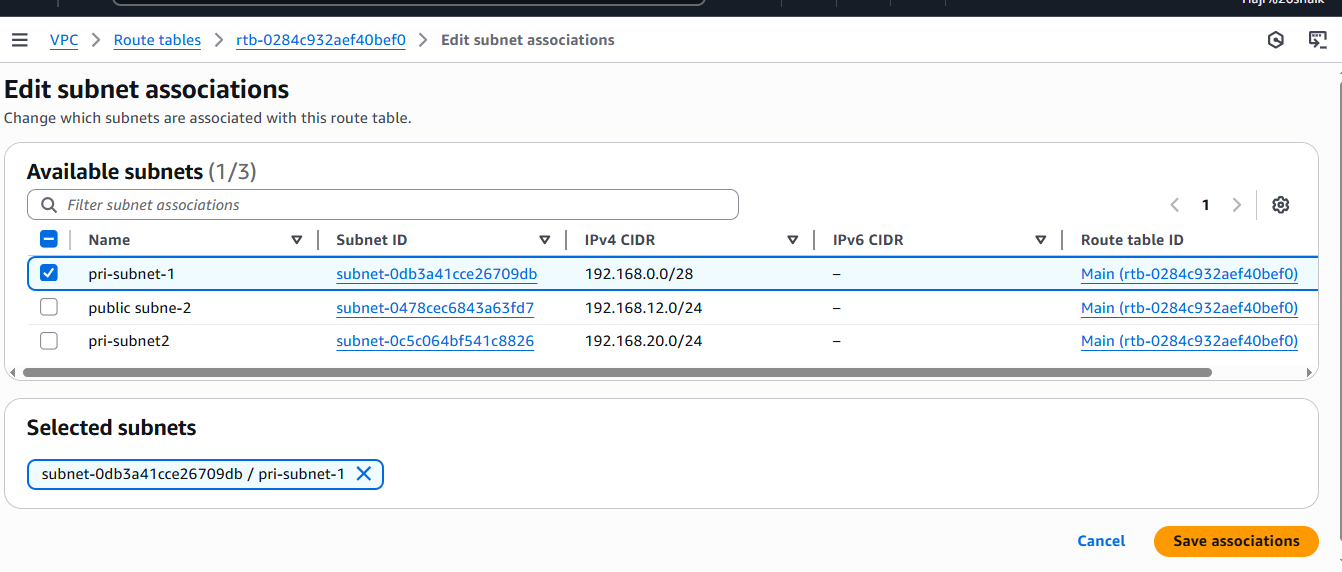


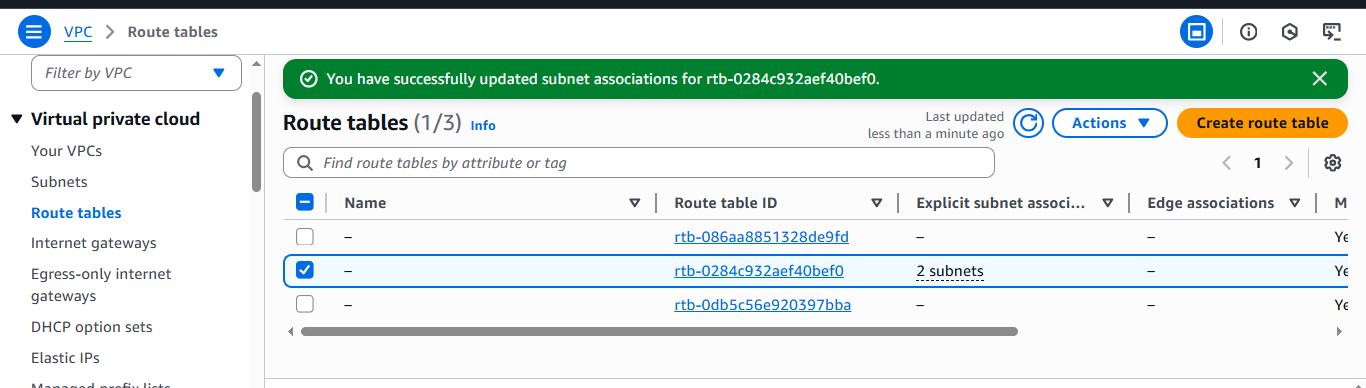
**Public subnet 2:**



1. **Add 2 private subnets in private route table.**

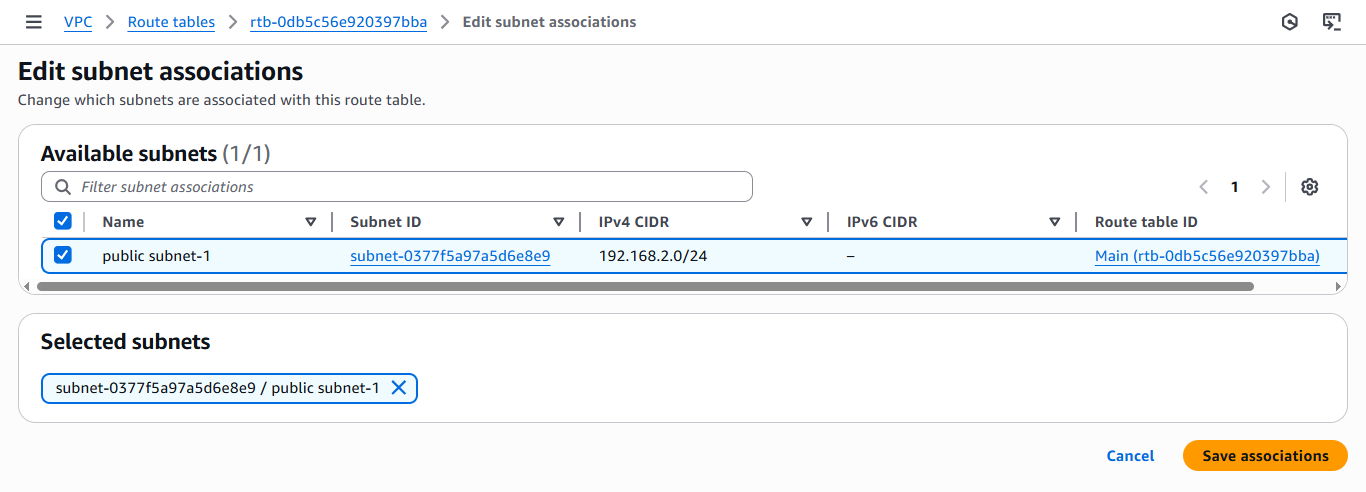
* Open ec2 console.
* Then go the search bar in vpc.
* Then the left coloumn there is route table option.
* Then you will see the the private route table.
* Then select the subnet and choose option on bottom of the subnet there is **“Edit sub-net association”** .
* Then select 2 private subnets and save it.
* Here the results are.

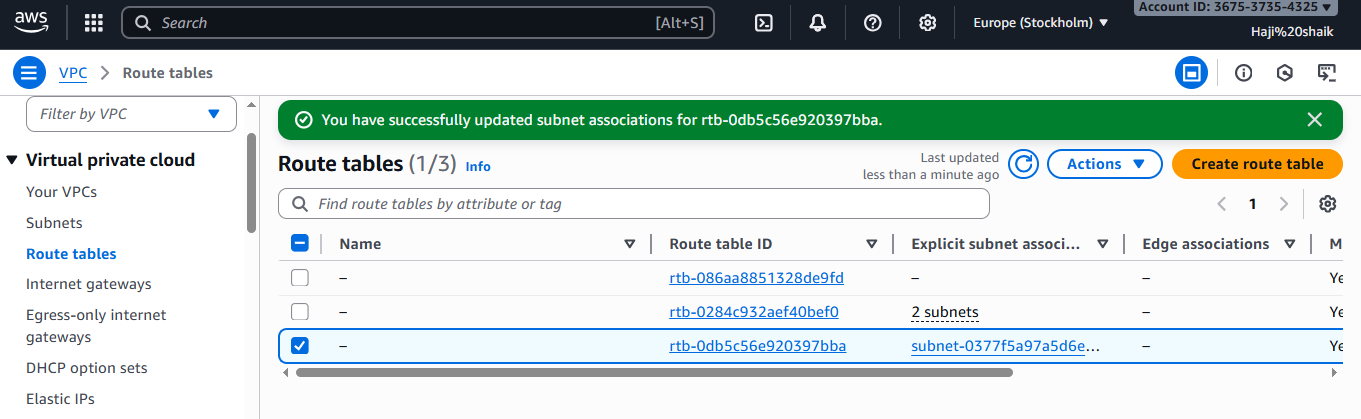




1. **Add 2 public subnets in public route table.**

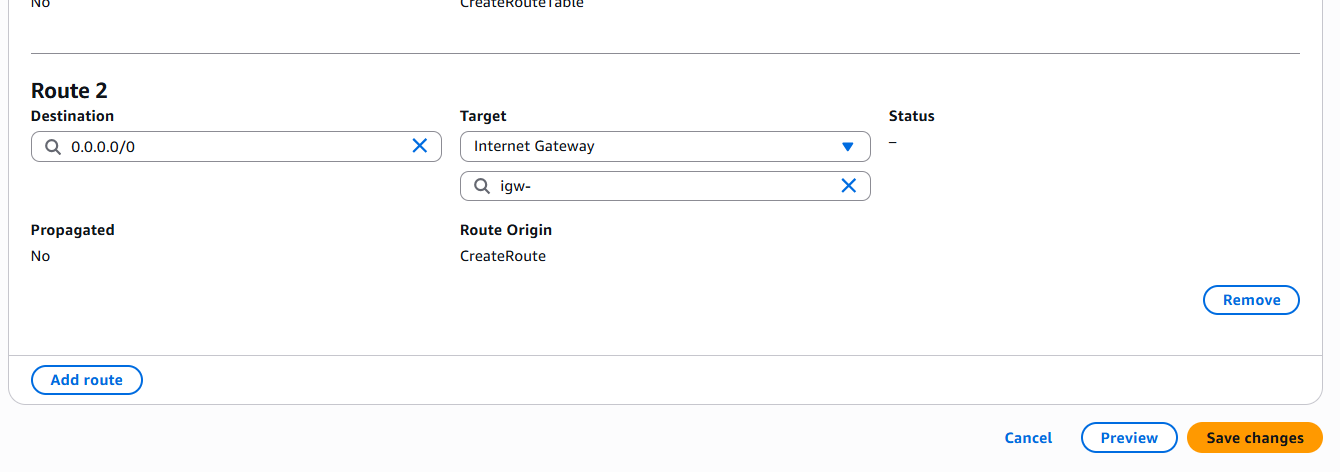
* Open ec2 console.
* Then go the search bar in vpc.
* Then the left coloumn there is route table option.
* Then you will see the the private route table.
* Then select the subnet and choose option on bottom of the subnet there is **“Edit sub-net association”** .
* Then select 2 private subnets and save it.
* Here the results are.

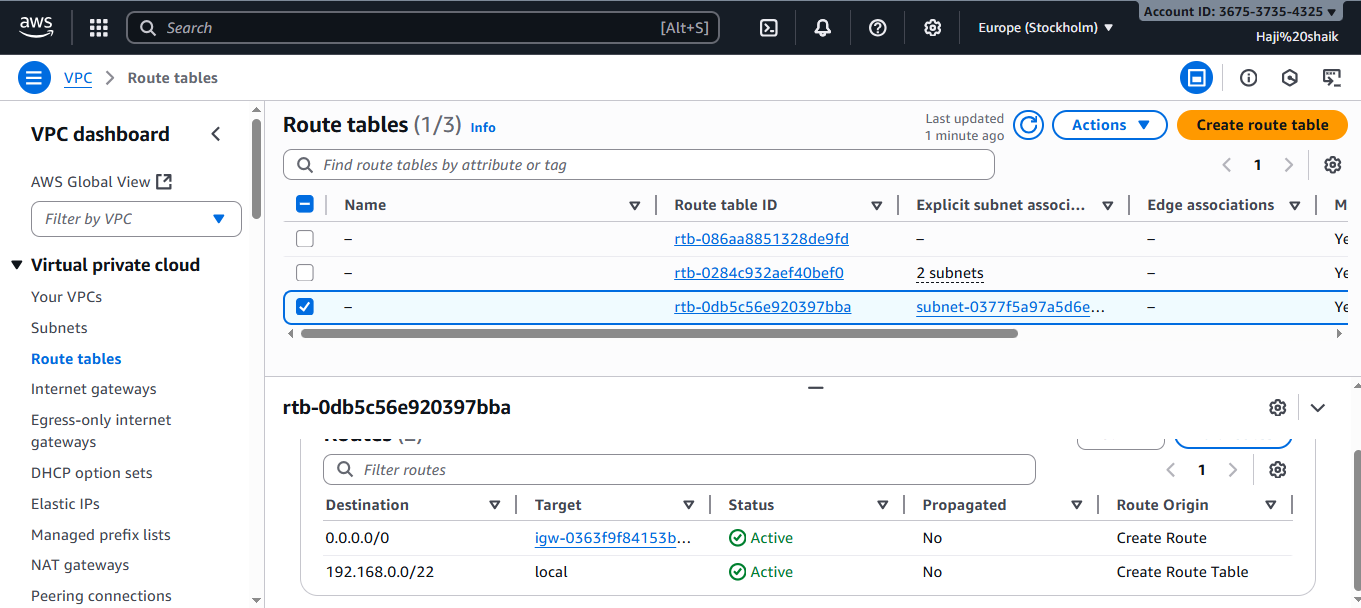




1. **Public route table will have the routes to internet and local.**

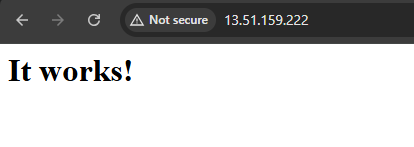
* Open ec2 console.
* Then go the search bar in vpc.
* Then the left column there is route table option.
* Create one **internet gate way** first.
* gave a name and create the internet gate-way.
* Then go to the route table >> select a route >> and edit the route >> and add the route.
* In that route values are 0.0.0.0 and name should b ur **IGW(internet gate way)** name.
* And save the changes.
* Here the results are…>>





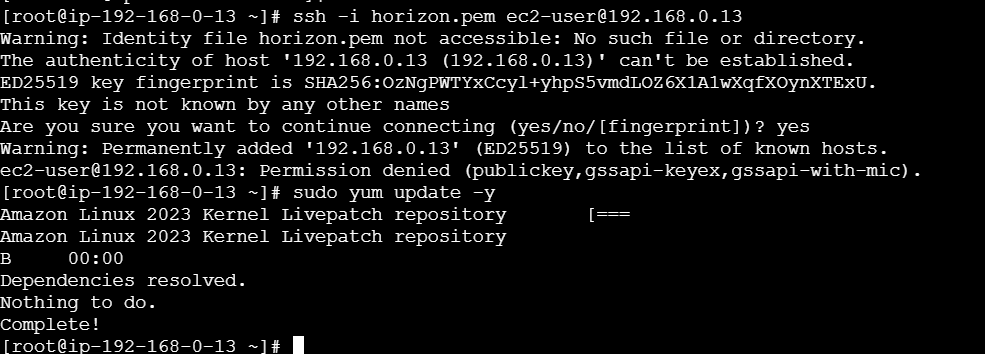
1. **Create EC2 in public subnet with t2.micro and install PHP.**

* open a browser and luanch the istance with the name of php-server.
* Then take amazon linux image and get ur pem key.
* And change the network settings in select ur vpc and select the public-subnet.
* Then go to security group instances and a rule of ssh-22 & http-80.
* Then lunch a instance and connected to the server.
* Sudo yum update.
* Sudo yum install httpd -y.
* Systemctl check status
* Systemctl start httpd
* Systemctl enable httpd.
* Then yum install **PHP (HYPER TEXT PREPROCCER)**
* sudo amazon-linux-extras enable php7.4
* sudo yum install -y php php-cli php-mysqlnd
* Then systemctl restart httpd.
* And visit the browser and use public ip and **<http://13.51.159.222:80>**
* Here the results are:



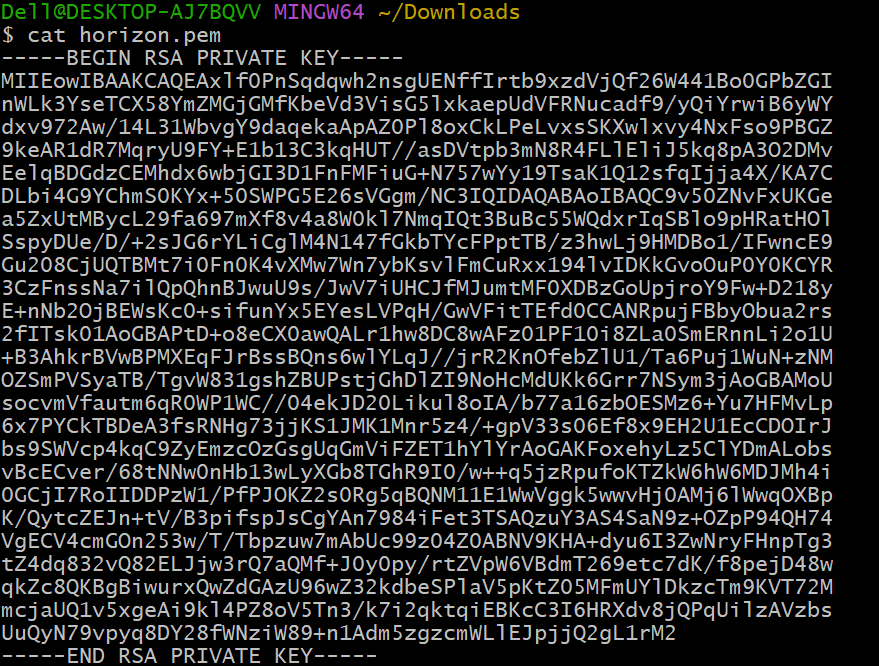
1. **Configure NAT gateway in public subnet and connect to private instance.**

* First go to the ec2 instance.
* Type in to the the vpc.
* Enter the vpc range and choose the vpc and internet gate way for private and natgateway for public subnet.
* Then check the routing table which are co-ordinate with public and private sub-nets.
* Then allocate the elastic ip for that.
* Then launch a instance with amazon linux image >> and take a pem then >> network settings.
* And selected a exciting vpc and private subnet and status will be disabled.
* Then check the security inbound rules which for ssh and http.
* Then luanch a instance and selected the private ip.
* Connect the server with ssh and pem and private ip.
* The go to the root user.
* Enter sudo yum uapdate..
* The out put will nothing to dependence.
* Here the reusults are:

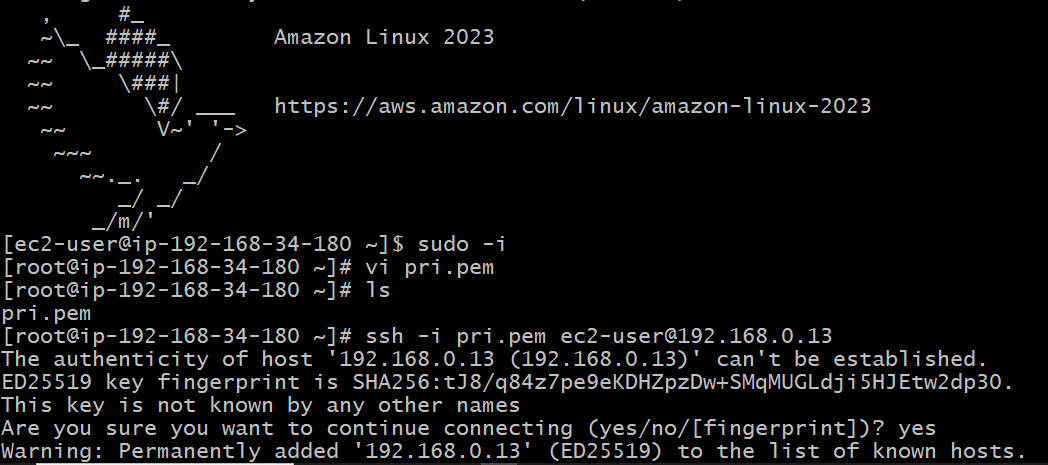


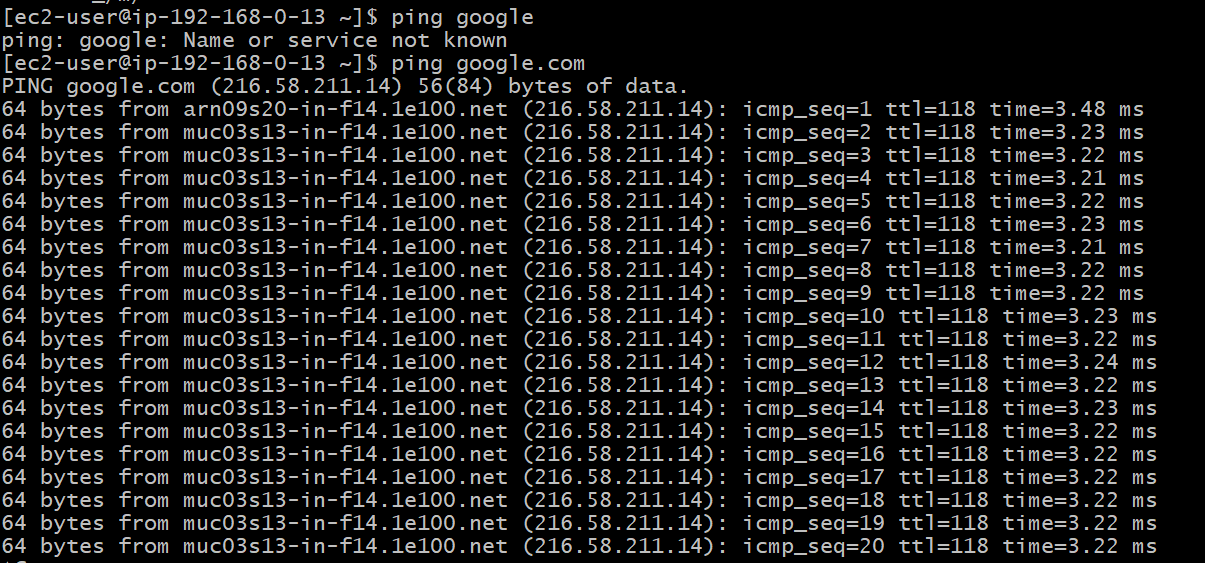
1. **Install Apache Tomcat in private EC2 and deplone in sample app.**

* Lunch 2 instances in aws. One is private instance and one is public instance.
* Then lunch private instance with public ip
* Cat the pem key.
* Then duplicate the pem key.
* Then ssh key private instance private key.

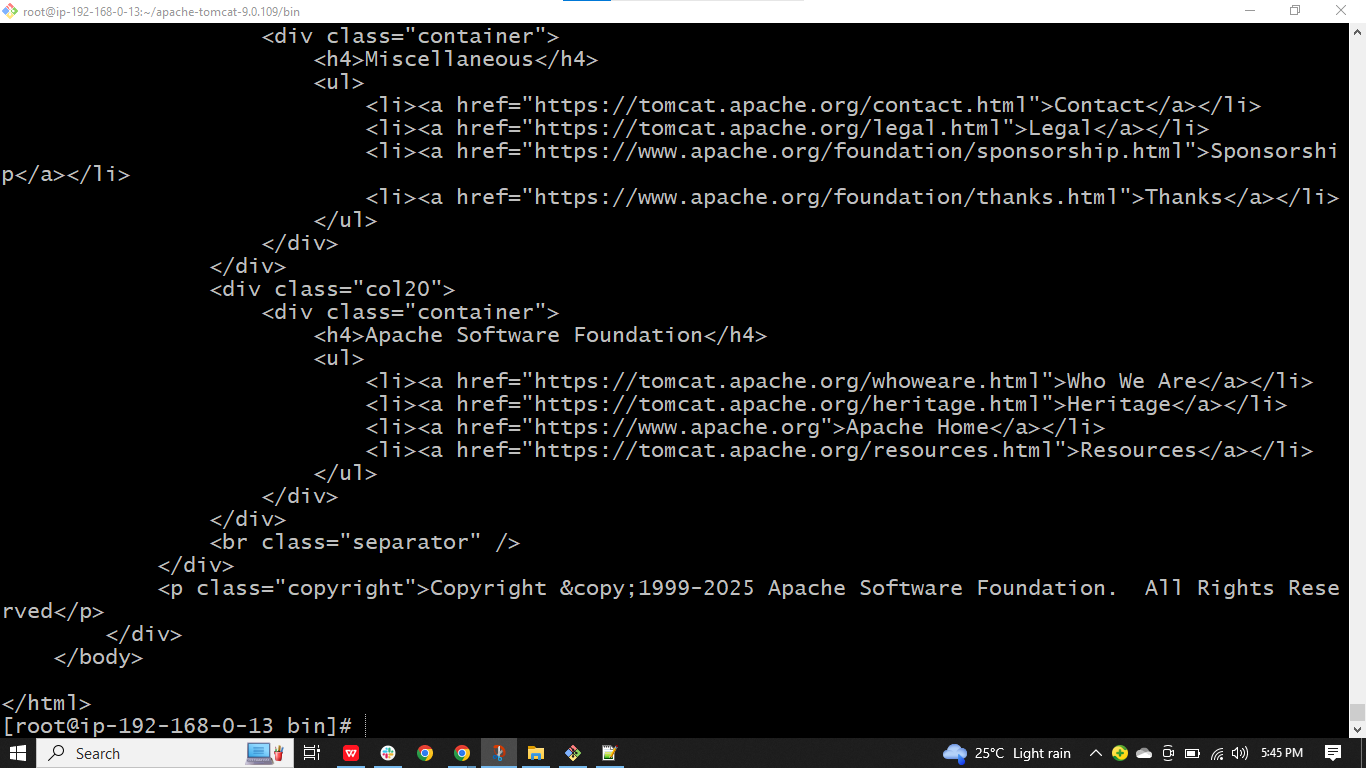


* Then ssh key private instance private key.
* it will connected to the server nd pin google.com





* Then java -version
* Why java for apache tomcat.
* Its very mandatory because of they gave uh out put in html language.
* Then java installation process.
* Then after download tar xvf it will extract the file
* Then ls
* Go to bin
* After ./startup.sh
* Check the results here:



1. **Configure VPC flow logs and store the logs in S3 and CloudWatch.**

* **Go** to your vpc.
* Select the vpc which uh want to create the flowlog.
* Then open action.
* And gave fill the details of first flowlog create.
* For a creating flow log.
* Create s3 bucket.
* And cloud watch log.
* Then IAM role created then only flow log will create.
* Here the results are:

