Nginx as a load balancer

Steps

Go to the VPC

**Your vpcs**

* **Create Vpc**

Name :- **VPC-PUBLIC**

IPV4 CIDR :-**192.168.0.0/20**

**Create vpc**

* **Create Subnet First**

Select :-**VPC-PUBLIC**

**Name:- public-sub**

Availablity zone:- **select 1a**

**192.168.1.0/24**

**Create subnet**

* **Create Subnet Second**

Select :-**VPC-PUBLIC**

**Name:- private-sub-1**

Availablity zone:- **select 1a**

**192.168.2.0/24**

**Create subnet**

* **Create subnet**  **Third**

Select **VPC-PUBLIC**

**Name:- private-sub2**

Availability zone :- **select 1c**

**192.168.3.0/24**

**Create subnet**

Select **internet gateway**

* **Create internet getway**

Name;-**public-IG**

**Create internet getway**

**Select VPC-PUBLICig : select action :- select public-IG**

**Attach internet gateway**

**Select NAT**

**Create nat getway**

Name:- new-IG

Subnet:- VPC-PUBLIC bcoz vo internet public se lenga

Allocate Elastic IP

Create nat getway

Select **route tables**

* **Create route table** **first**

Name:- **public-RT**

Select :- **VPC-PUBLIC**

**Create route table**

* **Create route table second**

Name :- **Private-RT**

Select :- **VPC-PUBLIC**

**Create route table**

Select **public-RT**

Now you see the routes

**Edits routes**

**Add route :-0.0.0.0/0 select Internet getway Save Changes**

**Select subnet association**

Edit Select **subnet association**

Select **public-sub** save association

Select **private-RT**

Now you see the routes

**Edits routes**

**Add route :-0.0.0.0/0 select Nat getway Save Changes**

**Select subnet association**

Edit Select **subnet association**

Select **private sub -1**

**Private-sub-2** save assocaiation

Go to the EC2

* **Launch instance** **first**

Name:- **public-instance**

**Ami**

**T2 micro**

Key name:- **newkey**

**Network setting:- select vpc :- VPC-PUBLIC Subnet:-public sub Auto-assign public ip:- Enable**

Create security group

Name:-**newone-SG**

**Add security group rule :- port :-80 source:- Anywhere**

**Launch**

* **Launch instance** **second**

Name:- **private-instance-1**

**Ami**

**T2 micro**

Key name:- **newkey**

**Network setting:- select vpc :- VPC-PUBLIC Subnet:-private-sub-1 Auto-assign public ip:- Disable**

Select **existing security group**

Select :-**newone-SG**

**Launch**

* **Launch instance** **third**

Name:- **private-instance-1**

**Ami**

**T2 micro**

Key name:- **newkey**

**Network setting:- select vpc :- VPC-PUBLIC Subnet:-private-sub-2 Auto-assign public ip:- Disable**

Select **existing security group**

Select :-**newone-SG**

**Launch**

**Now go to gitbash and send the key to the public instance using ssh command**

**Scp -I newkey.pem newkey.pem ec2-user@public-ip:/home/ec2-user**

**Now go to the console**

**Select public instance – connect this instance**

**Ls – now you see the key when we send**

**Sudo chmod 700 newkey.pem - bcoz we want to connect ssh in private instance**

**Ssh -I newkey.pem ec2-user@private-instance-1-ip (private ip)**

**Now you see the private instance**

**Sudo yum install nginx -y**

**Cd /usr/share/nginx/html - vim index.html h1 hello palash save**

**Sudo systemctl start nginx**

**Sudo systemctl enable nginx**

**Exit**

**Now go to the privet instance 2**

**Ssh -I newkey.pem ec2-user@private-instance-2-ip (private ip)**

**Now you see the second private instance**

**Sudo yum install nginx**

**Cd /usr/share/nginx/html - vim index.html h1 hello good morng save**

**Sudo systemctl start nginx**

**Sudo systemctl enable nginx**

**Exit**

**Now you are a public instance**

**Cd /etc/nginx**

**Sudo vim nginx.conf**

**events {**

**worker\_connections 768;**

**}**

**http {**

**upstream myapp**

**{**

**server private1-instance-ip-add:80 weight=1;**

**server private2-instance-ip-add:80 weight =2;**

**}**

**#These are my settings SHROBON**

**server {**

**listen 80;**

**location / {**

**proxy\_pass** [**http://myapp**](http://myapp/)**;**

**}**

**}**

**}**

**sudo systemctl start nginx**

**sudo systemctl enable nginx**

**now go to the public ip copy**

**and paste**

**now you see the hello palash refresh hello good morng you see two output in one time**

**that means single public ip hit and see the tow output when we refresh the load is balanced**

**note ham 3 no private instance pr alg server install kr skte hai like 1st private instance nginx, 2nd private instance httpd , 3rd private instance tomcat all the process are same**

**bus configuration me thoda changes krna ip:8080; or security group me tomcat ka port add krna**

**then public ip hit krne k bad 3 output show krenge nginx page refresh httpd refresh tomcat page**