

# AWS Overview

## **Agenda:**

Introduction to Cloud?

Cloud features: pay as you use and Horizontal/Vertical scalability

Cloud Models: Public, Private, Hybrid, Community

Cloud Services: IAAS PAAS SAAS

Cloud providers

AWS introduction

## **AWS Services:**

EC2 – Elastic Cloud Compute

EBS – Elastic Block Store

Auto scaling

ELB – Elastic Load Balancer

S3 – Simple Storage service

IAM – Identity and Access Management

VPC – Virtual Private Cloud

DNS Route 53 - DNS

RDS – Relational Database Service

## **Cloud Overview:**

### **Definition:**

Cloud computing refers to the **on-demand** delivery of IT resources and applications via the **internet** with pay as you go.

If we need we can get it right away

If we don't we can give it away.

These services are available over internet, we just need web browser and internet connection, we can request computing or storage infrastructure

Pay as you go: postpaid mobile bill or electricity bill

SAAS – Software as service

**Cloud Computing:** The practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer

### **Features:**

On demand provisioning

Scalability in minutes: we can scale out, or scale in

**Horizontal scalability:** can scale from 1 instance to several instances

Or can scale down from several instances to 1

Vertical Scalability: can go from 1GB RAM to 100GB RAM – for a single machine

### **Cloud Models:**

Public

Private

Hybrid

Community

**Cloud services:**

IAAS – Infrastructure as service

PAAS – Platform as service

**IAAS:**

Creating infrastructure as the service

Create VMs, attach storage, assign network to it

Creating infrastructure ourselves in IAAS

Pay per hour basis

**PAAS:**

Developers may need a java/.net platform.

They create all the infrastructure and put your code on it and run it

**SAAS:**

Software as a service:

In SAAS everything is managed by the cloud providers, h/w, servers, network, storage, virtualization, OS, middleware, data, run time, applications etc

**Examples:**

IAAS: Amazon ec2 microsoft azzure google cloud rackspace, ibm softlayer

PAAS: elastic beanstalk google app engine

SAAS: HRMS Human Resource Management system

When you are a simple client who want to make use of a software but you have nothing in hand then you use SaaS.

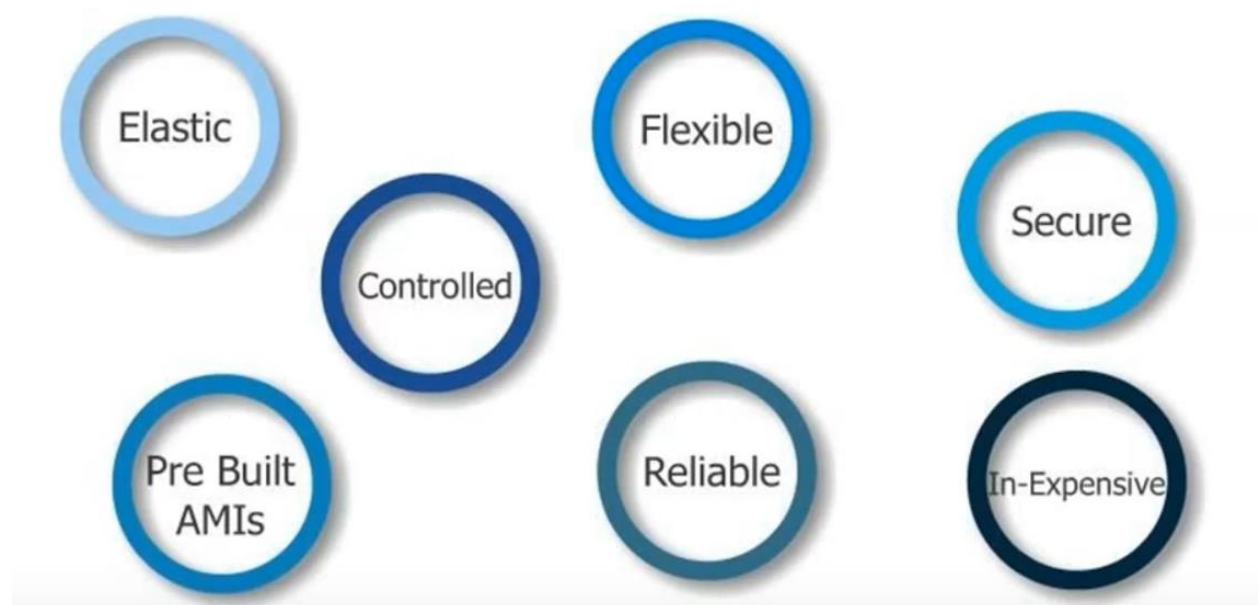
When you have a software developed by you, but you want to deploy and run on a publicly available platform then you use PaaS.

When you have the software and the platform ready but you want the hardware to run then you use IaaS.

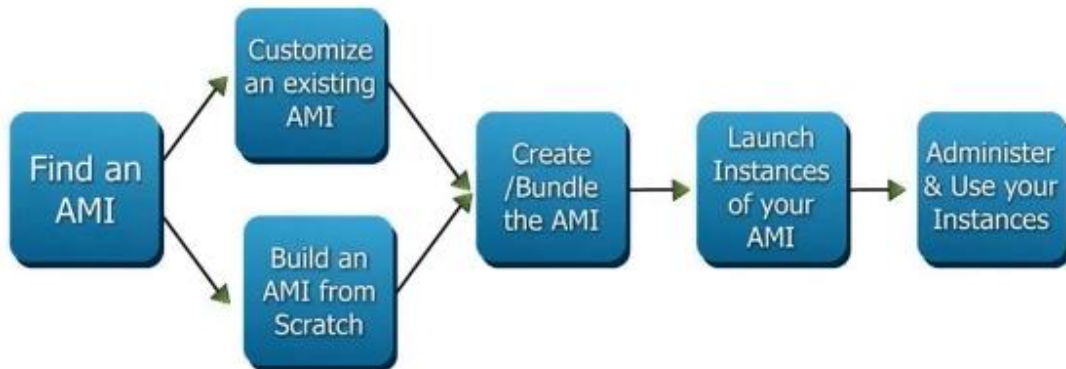
### **EC2 (Elastic Cloud Computing):**

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to allow you to obtain and configure capacity with minimal friction and to make web-scale cloud computing easier for developers.

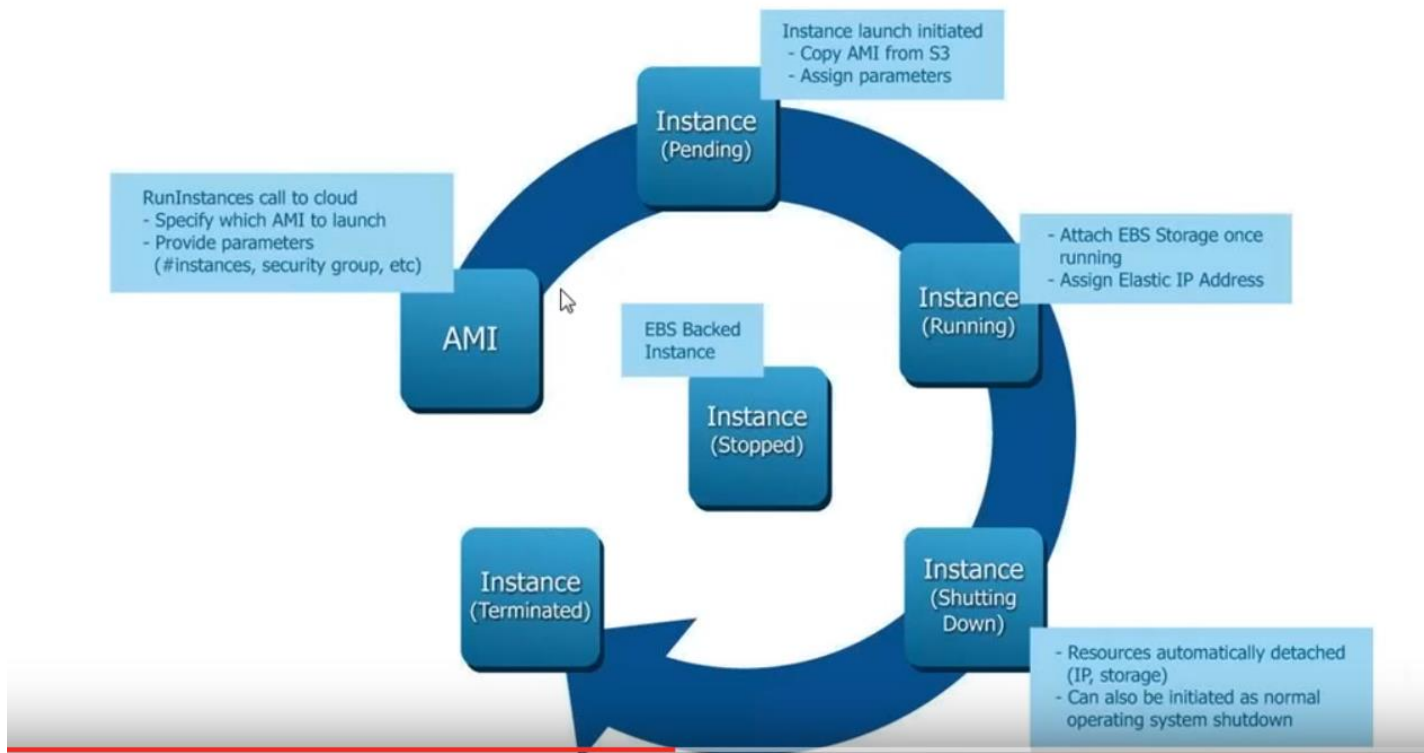
Amazon is elastic/flexible, supports lot of OS flavours



## EC2 Flow:



## EC2 Life Cycle:



**Default users to login to ec2:**

**ec2-user**

ubuntu

centos

root

admin

**To configure password less authentication between 2 ec2 instances:**

1. Login to ec2 instance and change ssh configuration file **/etc/ssh/sshd\_config** as below

PermitRootLogin prohibit-password to **PermitRootLogin yes**

PasswordAuthentication no to **PasswordAuthentication yes**

Then restart ssh

**RHEL → service sshd restart**

**Ubuntu → service ssh restart**

**To have password less authentication between 2 instances, master and slave:**

2. Generate the keys using ssh-keygen on master
3. Copy the keys using from master to slave using

**ssh-copy id -I ~/.ssh/id\_rsa.pub root@slave**

**ec2 instances:**

When you launch an EC2 instance, you receive a Public IP address by which that instance is reachable.

Once you stop that instance and restart the you get a new Public IP for the same instances.

So, Public IP gets changed every time for an instance after stop/start.

To overcome with this problem, we attach an Elastic IP to an Instance which doesn't change after you stop / start the instance as many times.

### **Advantage of Having Elastic IP**

-> It is kind of static IP for your Instance.

-> Doesn't change after stop/start.

Note: If you have Elastic IP in your account and it's not in use, then you will be charged for it.

### **EC2 Load balancer - Elastic Load Balancing**

It Is simply a software to manage the load of application.

For ex: if there are 10 web servers, we have public ip for one webserver, the traffic flow to one webserver will be balanced across the 10 webserver using load balancer.

To automatically distribute incoming web traffic across multiple ec2 instances.

Add and remove ec2 instances without changing the inflow

If one of the instance fails, AWS ELB will automatically redirect the traffic to another running instance.

If the failed instance is restored, ELB restores the traffic to that instance.

ELB is like a traffic manager, it minimizes the risk of overloading one single instance and provides continuous monitoring of the health of AWS instances.

ELB can be setup to load balance incoming application traffic across Amazon ec2 instances in a single Availability zone or multiple AZs – AZ is a data center.

The AWS Classic Load Balancer (CLB) operates at Layer 4 of the OSI model. What this means is that the load balancer routes traffic between clients and backend servers based on IP address and TCP port.

AWS Application Load Balancer (ALB) operates at Layer 7 of the OSI model. At Layer 7, the ELB has the ability to inspect application-level content, not just IP and port. This lets it route based on more complex rules than with the Classic Load Balancer.

Internal load balancer -> internal, not external, not internet facing

Response timeout – 5 sec – time out value

Health check interval – how often it pings

Unhealth threshold – 2 consecutive time outs, it tries to access page twice

Healthy threshold - 3 ping the LB and report it as healthy as 3 times. – 3 successful pings

#### **Exercise:**

- 1) Create 2 ec2 instances
- 2) Install apache server on both and configure
- 3) On one of the machine in /var/www/html/index.html add server1
- 4) On the other machine in /var/www/html/index.html add server2
- 5) Create load balancer with 2 ec2 instances
- 6) Open the load balancer using the url provided, by default it points to server1, so in the web browser output we can see server1
- 7) Bring down the server1, and after some time try to access the same load balancer, it should display server2
- 8) If ec2 instance is brought up again, it should show server1 in the web browser output



**EC2 Auto Scaling:**

Auto scaling helps to have correct number of EC2 instances available to handle the load of application.

Create collection of EC2 instances called auto scaling groups

While creating the group, we need to specify the desired capacity.

**Example:**

There is website to download video games.

And 2 servers are serving the requests from the users.

There is increase in the demand, and the servers are scaled to 4 servers.

Same thing is done by Auto scaling group automatically.

Minimum - 2

Maximum – 10

Desired – 4

If desired no. of servers is changed to 6, 2 instances will be spinned up.

Launch configuration:

Tells the auto configuration service what exactly it must launch.

EC2 instance type and size

AMI

Security groups/ssh keys/IAM instance profile

User data

-

Launch configuration → which instances you want to launch/ OS/Config etc – defined here

Create auto scaling group → minimum max desired instances are defined here.

### **Elastic Block Storage:**

EBS is a virtual hard drive that can be used with ec2 instances.

Can attach it to ec2 instance.

Can place file system on it

If EBS is used to store your data, if any ec2 instance fails, we can still get the data as data is not stored on the instance, but on EBS volume.

1GB to 1TB in size, created in a particular AZ – availability zone/data center

It appears as a mounted device.

1 volume can be attached to 1 instance only

Great performance for read/write in case of heavy load – iops

To create to go EC2 dash board → ebs volume → create volume

Attach volume to instance,

And run lsblk/fdisk -l to check the volume on the ec2 instance.

We can remove the volume by force detach

We can create snapshots to replicate the data

Go to snapshots to check snapshot created.

To create high performance drive, we can create volume from snapshot.

### **S3 vs EBS vs EFS**

Simple storage service

Elastic Block Storage

Elastic File System.

S3 can be used when the requirement is for WORM – write once and read many times

EBS works best as server disks – high performance in terms of read and write.

EFS – to share the file system across different servers.

**Cloud watch** – monitoring service

Monitoring can be enabled while creating instance.

Detailed monitoring – happens every 1 min

Standard monitoring happens every 5 min

We can create alarms using cloud watch

Like: if cpu utilization goes > 90% for 2 min, send me a email/text message

**RDS – Relational Data base Service:**

Allows to create mysql, mariadb, oracle sql, postgresql, Aurora db (Amazon's database)

Easy administration

RDS db instances are preconfigured with appropriate settings

- Fully Managed Relational database – does not allow access to underlying OS
- Can connect to data base using mysql
- RDS runs on instance types ex: t2.micro, m4.large etc

Enable multi AZ deployments for HA, backups

Main benefits:

- 1) AWS takes care of underlying platform, so automatic updates

2) Automatic backups

3) Automatic recovery in case of failures

For production env – multi AZ failover has to be enabled

Read replicas – copies of the primary data base

Can scale up read replicas to increase the data base

We can enable HA/elasticity/fail over using multi AZ / read replicas

We can take backup/snapshots manually

Exercise:

Go to RDS Service

Launch a new db instance

Select mysql

Select Dev/Test environment – as it is free tier eligible

Select same VPC and Subnet for the DB so that ec2 instances can access it

Create a subnet group

Select vpc

Select the availability zone – us-east-1c

DB Instance type: t2.micro

Multi AZ deployment – na for free tier

## **DNS/ Route 53:**

Domain Management service

Key features:

1. Domain Registration
2. DNS – Domain Name System Service
3. Health Checking

Which resource should have dns

ELB?

EC2?

## **Hosted Zones:**

1. Can register a domain with AWS

Public hosted zone/ Private hosted zone

## **Resource Record sets:**

Instructions that match domain names to IP

They are comprised of

Record type

Standard/alias

Routing policy

Evaluate target health

## **Common record types:**

A – to point to a domain on IPV4 address

AAAA – to point to a domain in IPV6 address

CNAME - to point to hostname to another hostname

MX - used to route email

Alias Record sets:

Alias record set contains a pointer to an AWS specific resource for ex:

Elb

Ec2 instance

**Routing policy:**

Simple: Route all traffic to one end point

Weighted: Route all traffic to multiple end points

Latency: Route traffic to end point based on the users latency to various end points

Failover: Route traffic to a secondary end point if the primary is not available

GeoLocation: Route traffic to and end point based on the geo graphical location of the user

Amazon Aurora – data base engine

Amazon glacier – to store the archives , huge data bases

Design your own website: <http://www.opendesigns.org/design/blue-skies/>

Development; dropbox – application to upload docs /pdfs/videos

### **Amazon VPC:**

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.

### **Using VPC:**

Launch instances into a subnet of our choice

Assign custom ip address ranges in each subnet

Configure route tables between subnets

Create internet gateways and attach them to subnets

Better security control.

### **Create custom VPC:**

1. create custom vpc

give cidr block: 10.0.0.0/16

2. create subnet -

cidr 10.0.1.0/24

3. create internet gateway

4. after creating vpc, it creates route table by default

5. attach internet gateway to vpc

6. go to route table add a route to internet

7. make sure that subnet has internet gateway. which has route to internet

launch ec2 instance in the custom vpc

try to access using private ip --> cannot be accessed



create elastic ip

associate with ec2

### **How to access instances in private n/w:**

1. Create nat server - n/w address translation - public instances to communicate out to internet

outside world should be able to access using 80 port

create 3 subnets

10.0.1.0/24

10.0.2.0/24 private subnet

10.0.3.0/24 private subnet

create a new security group:

nat security group

select custom vpc

edit the security group

add inbound

http / and https from 10.0.2.0/24

https/ and https from 10.0.3.0/24

add outbound

http and https from 0.0.0.0/0

3. provision network address translation server

create ec2 instance

amzn-ami-vpc-nat

4. while creating instance select custom vpc , select internet facing subnet

security group - select as nat security group

5. allocate new address and associate to nat server instance

6. go to ec2 dashboard

actions--> network --> change source destination check --> disable it

7. enable a route out to using nat

copy instance id

select second route table

edit

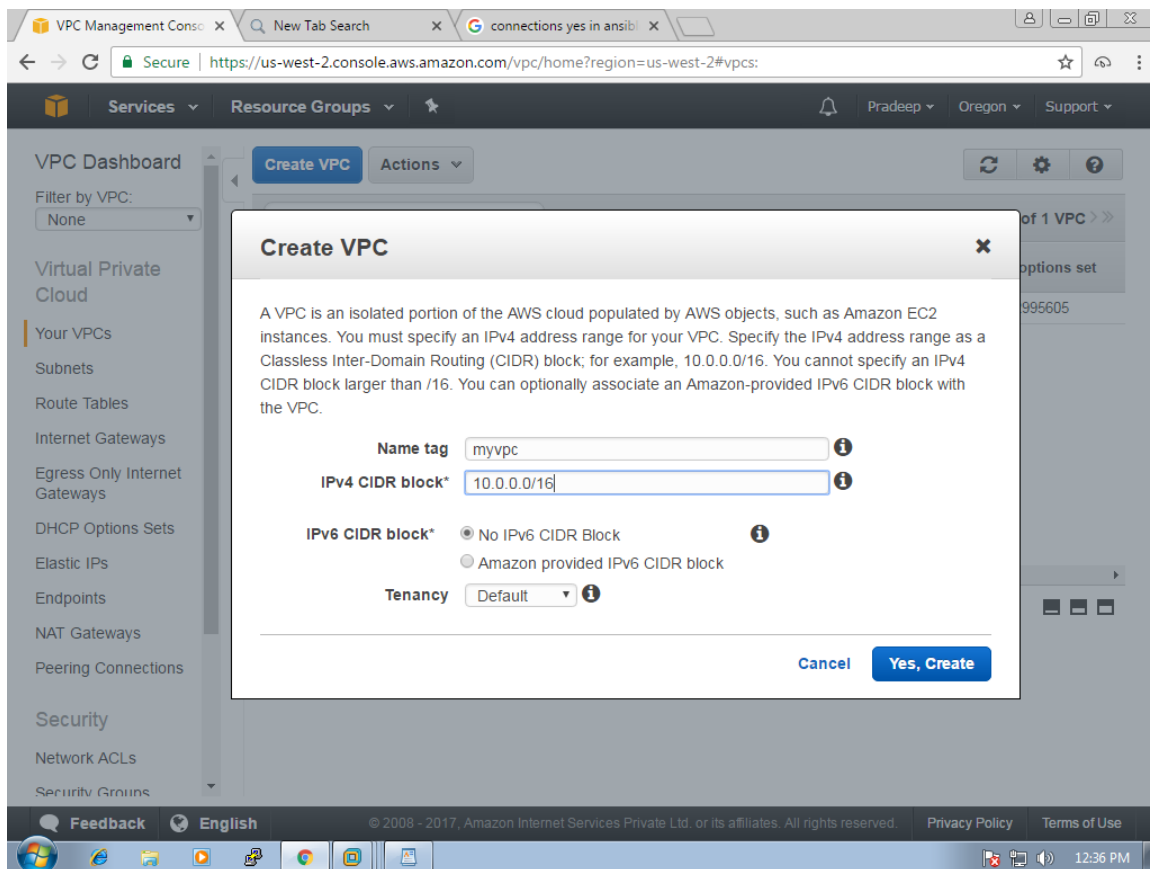
add another subnet 10.0.2.0/24

this rt is associated with 2 private subnets

add route to access internet

0.0.0.0/0 target is instance . paste the instance

## Creating VPC:



VPC Management Console | New Tab Search | connections yes in ansibl |

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#vpcs: |

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

Create VPC | Actions

Search VPCs and their properties

<< 1 to 2 of 2 VPCs >>

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set
myvpc	vpc-4ca4d02b	available	10.0.0.0/16		dopt-62995605
	vpc-b4bbcf3	available	172.31.0.0/16		dopt-62995605

vpc-4ca4d02b | myvpc

Summary | Flow Logs | Tags

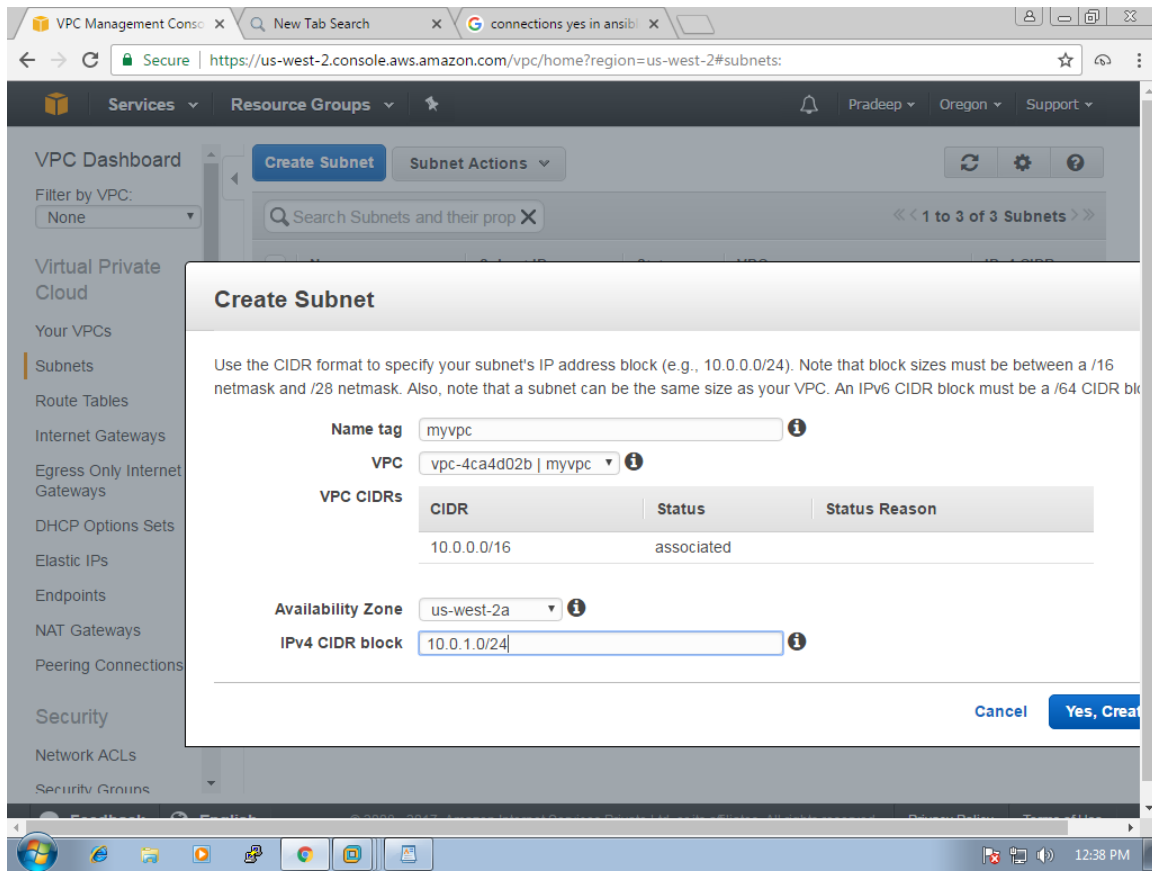
VPC ID: vpc-4ca4d02b | myvpc  
State: available  
IPv4 CIDR: 10.0.0.0/16  
IPv6 CIDR:  
DHCP options set: dopt-62995605

Network ACL: acl-f5755592  
Tenancy: Default  
DNS resolution: yes  
DNS hostnames: no  
ClassicLink DNS Support: no

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

12:36 PM

create subnets



another subnet

VPC Management Console

New Tab Search

connections yes in ansibl

Securehttps://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#subnets:

Services

Resource Groups

Pradeep

Oregon

Support

VPC Dashboard

Create Subnet

Subnet Actions

Filter by VPC:None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

Create Subnet

Subnet Actions

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag

myvpc

VPC

vpc-4ca4d02b | myvpc

VPC CIDRs

CIDR	Status	Status Reason
10.0.0.0/16	associated	

Availability Zone

us-west-2b

IPv4 CIDR block

10.0.2.0/24

Cancel

Yes, Create

myvpc

IPv4 CIDR: 10.0.1.0/24

IPv6 CIDR:

State: available

Route table: rtb-d80b1dbf

Network ACL: acl-45755592

Default subnet: no

12:38 PM

VPC Management Console | New Tab Search | connections yes in ansibl |

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#subnets: |

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

**Create Subnet** | Subnet Actions

Search Subnets and their prop X

<< 1 to 5 of 5 Subnets >>

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR	Availability Zone
<input type="checkbox"/>		subnet-0d606655	available	vpc-b4bbcf3	172.31.0.0/20	us-west-2b
<input type="checkbox"/>	myvpc	subnet-7e058919	available	vpc-4ca4d02b   myvpc	10.0.1.0/24	us-west-2b
<input type="checkbox"/>		subnet-e904888e	available	vpc-b4bbcf3	172.31.16.0/20	us-west-2b
<input checked="" type="checkbox"/>	myvpc	subnet-144cea5d	available	vpc-4ca4d02b   myvpc	10.0.2.0/24	us-west-2b
<input type="checkbox"/>		subnet-e24fe9ab	available	vpc-b4bbcf3	172.31.32.0/20	us-west-2b

subnet-144cea5d | myvpc

Summary | Route Table | Network ACL | Flow Logs | Tags

Subnet ID: subnet-144cea5d | myvpc | Availability Zone: us-west-2b

IPv4 CIDR: 10.0.2.0/24 | Route table: rtb-d80b1dbf

IPv6 CIDR: | Network ACL: acl-f5755592

State: available | Default subnet: no

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

12:39 PM

route tables:



VPC Management Console

New Tab Search

connections yes in ansibl

Securehttps://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routetables:

Services

Resource Groups

Pradeep

Oregon

Support

VPC Dashboard

Filter by VPC:  
None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

Create Route Table

Delete Route Table

Set As Main Table

Search Route Tables and their

<< 1 to 2 of 2 Route Tables >>

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>		rtb-d80b1dbf	0 Subnets	Yes	vpc-4ca4d02b   myvpc
<input type="checkbox"/>		rtb-370b1d50	0 Subnets	Yes	vpc-b4bbcf3

Create Route Table

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag

myvpcroute

VPC

vpc-4ca4d02b | myvpc

Cancel

Yes, Create

Feedback

English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

12:41 PM

VPC Management Console | New Tab Search | connections yes in ansibl |

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routetables: |

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables**
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

Create Route Table | Delete Route Table | Set As Main Table

Search Route Tables and their X

<< 1 to 3 of 3 Route Tables >>

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>		rtb-d80b1dbf	0 Subnets	Yes	vpc-4ca4d02b   myvpc
<input type="checkbox"/>		rtb-370b1d50	0 Subnets	Yes	vpc-b4bbcf3
<input checked="" type="checkbox"/>	myvpcroute	rtb-4b091f2c	0 Subnets	No	vpc-4ca4d02b   myvpc

rtb-4b091f2c | myvpcroute

Summary | Routes | Subnet Associations | Route Propagation | Tags

Route Table ID: rtb-4b091f2c | myvpcroute

Explicitly Associated With: 0 Subnets

Main: no

VPC: vpc-4ca4d02b | myvpc

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

12:42 PM

edit in subnet association

VPC Management Console | New Tab Search | connections yes in ansibl |

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routetables: |

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

Your VPCs

Subnets

**Route Tables**

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

Create Route Table | Delete Route Table | Set As Main Table

Search Route Tables and their

<< 1 to 3 of 3 Route Tables >>

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>		rtb-d80b1dbf	0 Subnets	Yes	vpc-4ca4d02b   myvpc
<input type="checkbox"/>		rtb-370b1d50	0 Subnets	Yes	vpc-b4bbcf3
<input checked="" type="checkbox"/>	myvpcroute	rtb-4b091f2c	0 Subnets	No	vpc-4ca4d02b   myvpc

Summary | Routes | **Subnet Associations** | Route Propagation | Tags

Edit

Subnet	IPv4 CIDR	IPv6 CIDR
You do not have any subnet associations.		
The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:		
Subnet	IPv4 CIDR	IPv6 CIDR
subnet-7e058919   myvpc	10.0.1.0/24	-
subnet-144cea5d   myvpc	10.0.2.0/24	-

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

12:56 PM

check the check box of 1st subnet

VPC Management Console | New Tab Search | connections yes in ansibl |

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#routetables: |

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: None

Virtual Private Cloud

- Your VPCs
- Subnets
- Route Tables**
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections

Security

- Network ACLs
- Security Groups

Create Route Table | Delete Route Table | Set As Main Table

Search Route Tables and their X

<< 1 to 3 of 3 Route Tables >>

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>		rtb-d80b1dbf	0 Subnets	Yes	vpc-4ca4d02b   myvpc
<input type="checkbox"/>		rtb-370b1d50	0 Subnets	Yes	vpc-b4bbcf3
<input checked="" type="checkbox"/>	myvpcroute	rtb-4b091f2c	0 Subnets	No	vpc-4ca4d02b   myvpc

rtb-4b091f2c | myvpcroute

Summary | Routes | **Subnet Associations** | Route Propagation | Tags

Cancel | Save

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input checked="" type="checkbox"/>	subnet-7e058919   myvpc	10.0.1.0/24	-	Main
<input type="checkbox"/>	subnet-144cea5d   myvpc	10.0.2.0/24	-	Main

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

12:57 PM

and click on "Save", output is

The screenshot shows the AWS VPC Management Console interface. The left sidebar contains navigation links for VPC Dashboard, Virtual Private Cloud, Your VPCs, Subnets, Route Tables (selected), Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups.

The main content area displays the 'Route Tables' section. At the top, there are buttons for 'Create Route Table', 'Delete Route Table', and 'Set As Main Table'. Below these is a search bar and a table listing route tables. The table has columns: Name, Route Table ID, Explicitly Associated, Main, and VPC.

Name	Route Table ID	Explicitly Associated	Main	VPC
	rtb-d80b1dbf	0 Subnets	Yes	vpc-4ca4d02b   myvpc
	rtb-370b1d50	0 Subnets	Yes	vpc-b4bbcf3
myvpcroute	rtb-4b091f2c	1 Subnet	No	vpc-4ca4d02b   myvpc

Below the table, the 'Subnet Associations' tab is selected. It shows an 'Edit' button and a table of subnet associations. The table has columns: Subnet, IPv4 CIDR, and IPv6 CIDR.

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-7e058919   myvpc	10.0.1.0/24	-

A message states: 'The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:'. Below this is another table of subnet associations.

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-144cea5d   myvpc	10.0.2.0/24	-

The bottom of the console shows a footer with 'Feedback', 'English', copyright information, 'Privacy Policy', and 'Terms of Use'. The system clock shows 12:58 PM.

## Network ACLs:

where we can give access to inbound and out bound. as of now we are left with default

you can edit by clicking "edit button"

VPC Management Console | New Tab Search | connections yes in ansibl

Secure | https://us-west-2.console.aws.amazon.com/vpc/home?region=us-west-2#acls:

Services | Resource Groups | Pradeep | Oregon | Support

### VPC Dashboard

Filter by VPC: **None**

- Virtual Private Cloud
- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- DHCP Options Sets
- Elastic IPs
- Endpoints
- NAT Gateways
- Peering Connections
- Security
  - Network ACLs**
  - Security Groups

**Create Network ACL** | Delete

Search Network ACLs and the X

<< 1 to 2 of 2 Network ACLs >>

<input type="checkbox"/>	Name	Network ACL ID	Associated With	Default	VPC
<input type="checkbox"/>		acl-4975552e	3 Subnets	Yes	vpc-b4bbcf3
<input checked="" type="checkbox"/>		acl-f5755592	2 Subnets	Yes	vpc-4ca4d02b   myvpc

**acl-f5755592**

Summary | **Inbound Rules** | Outbound Rules | Subnet Associations | Tags

Allows inbound traffic. Because network ACLs are stateless, you must create inbound and outbound rules.

Cancel | **Save**

View: All rules

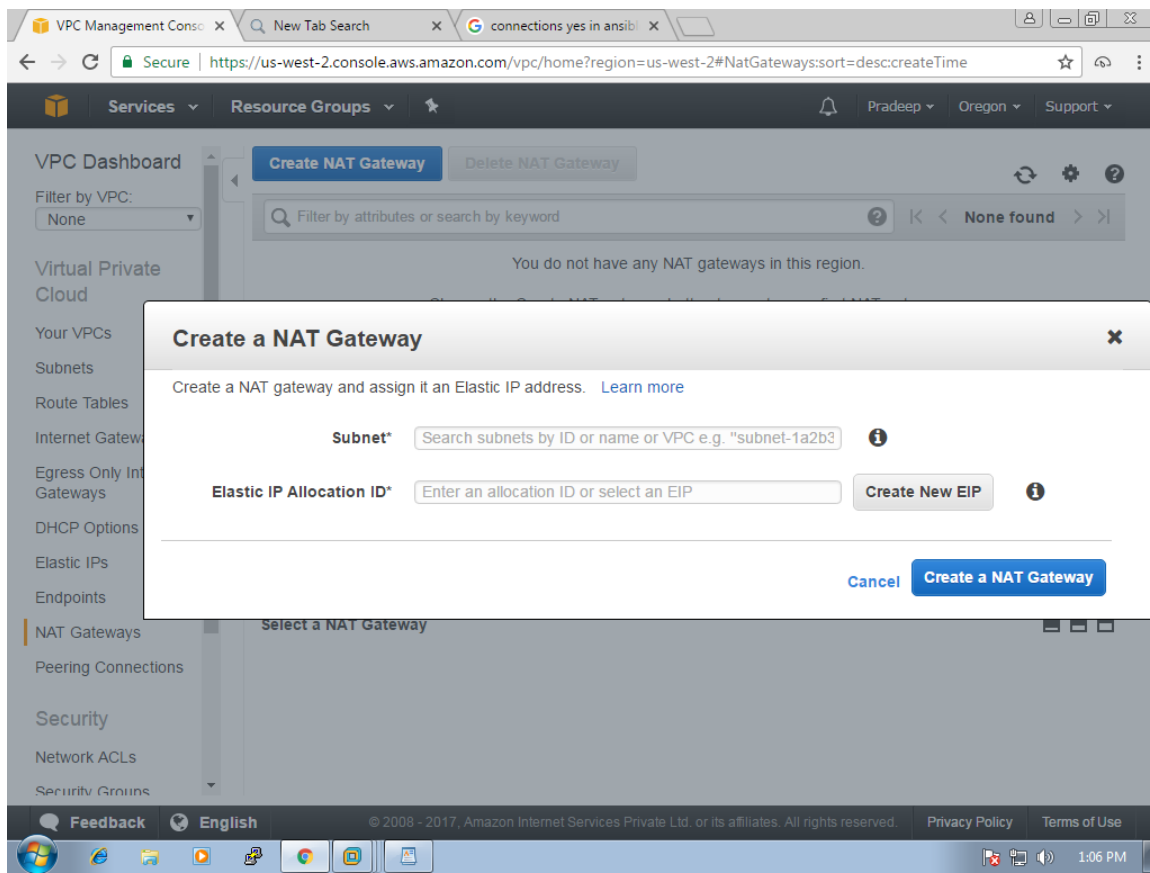
Rule #	Type	Protocol	Port Range	Source
100	ALL Traffic	ALL	ALL	0.0.0.0/0

Add another

Feedback | English | © 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

1:02 PM

Now navigate to NAT Gateways in the left pane and click on "create a NAT Gateway"

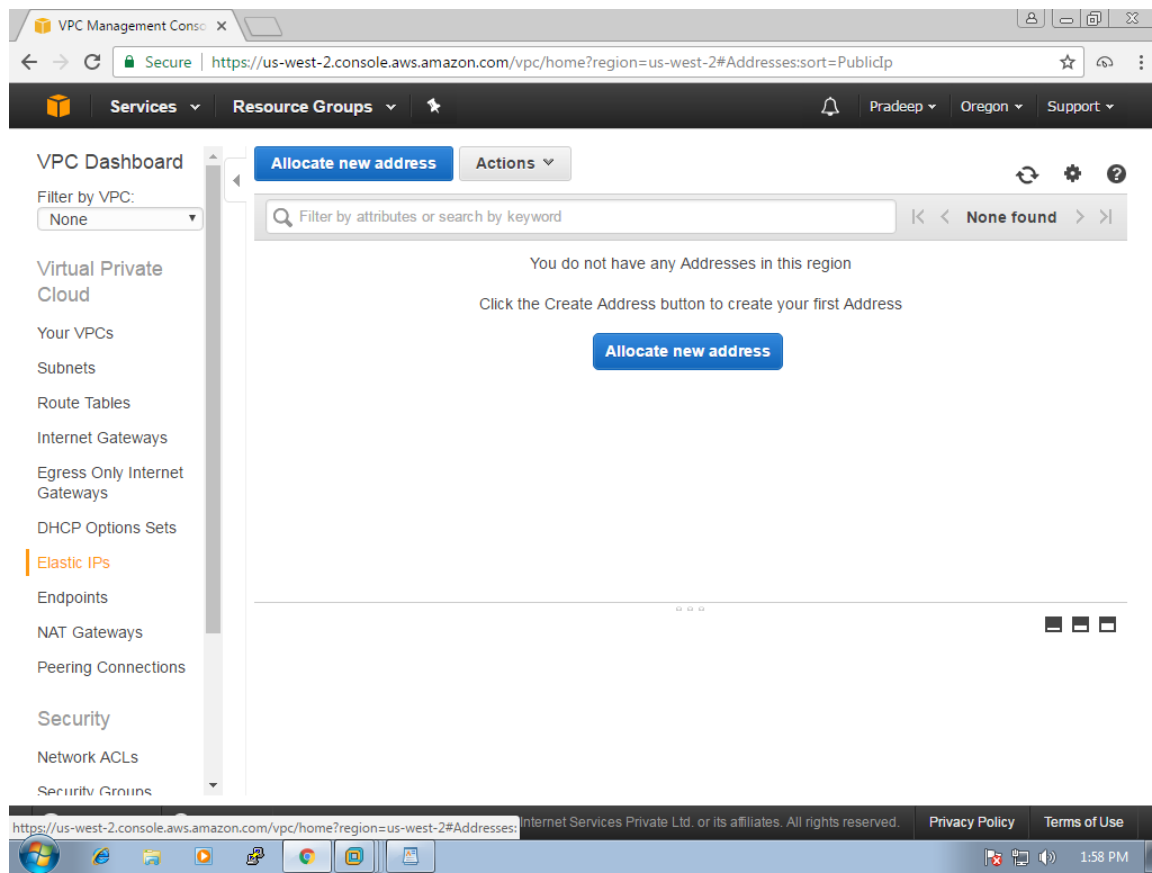


click on create New EIP button, where elastic IP address is created.

if we need fixed public IP , we need to create elastic IP address which chargeable.

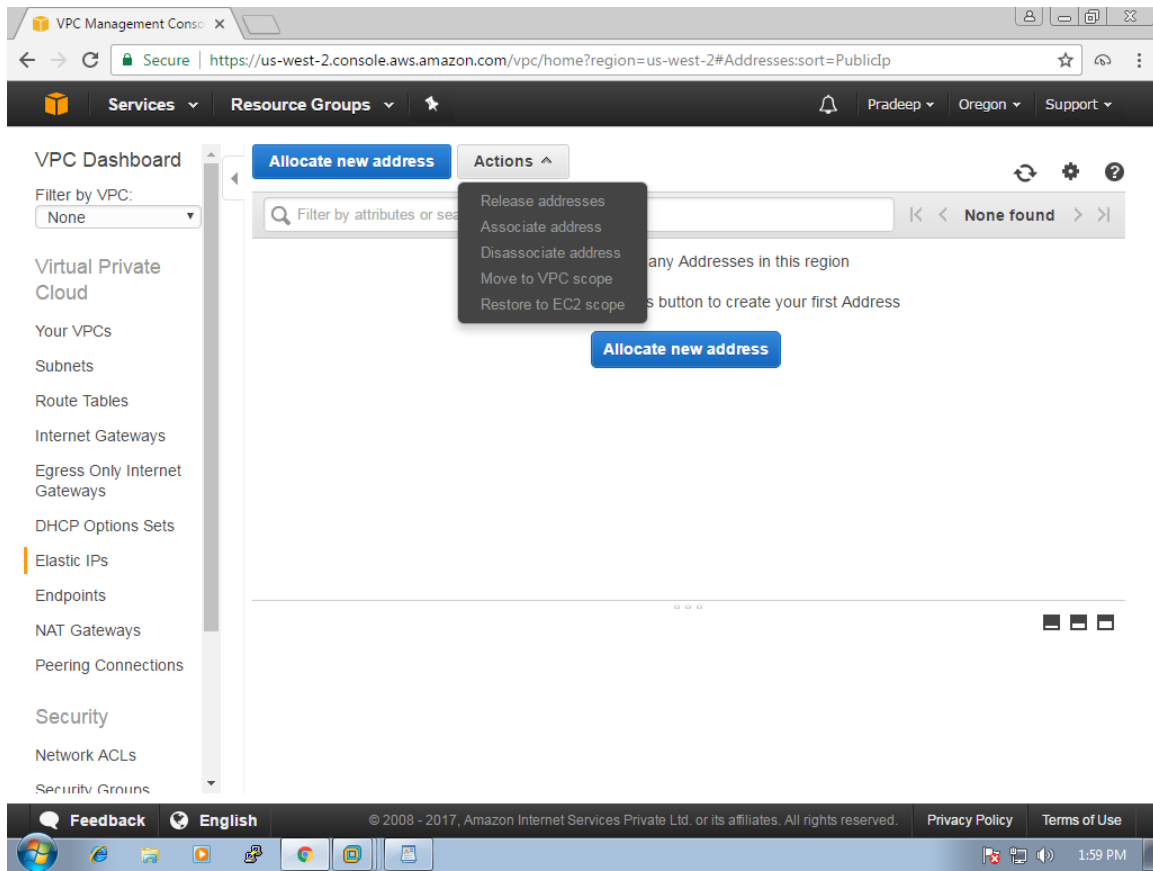
if you have created the elastic Ip address, delete the ip address in the NATgateways.

and also remove in the Elastic IP's as follow in the screen shot



select the ip address and click on release address's in the actions drop down





Creating and restoring from backup/snapshot:

lsblk

partprobe

fdisk -l

fdisk /dev/xvdf

create a partition number 2

mkdir /data

mount /dev/sdfx2 /data

cd /data

mkdir test1 test2

create snapshot

remove files

unmount old volume

detach old volume

attach new volume

mount it

files will be restored

resize:

umount

delete partition

recreate with new size

mount

resizefs /dev/xvdf2

e2fsck

mount

ls

### **AWS CLI:**

apt-get install awscli

aws configure

aws ec2 create-key-pair --key-name test-key --query 'KeyMaterial' --output text > key.pem

aws ec2 create-security-group --group-name group1 --description "group1"

aws ec2 authorize-security-group-ingress --group-name group1 --protocol tcp --port 22 --cidr 0.0.0.0/0

```
aws ec2 run-instances --image-id ami-6e1a0117 --count 1 --instance-type t2.micro --key-name test-key --security-groups group
```

```
aws ec2 terminate-instances --instance-ids i-0c47113d770259890
```

### **Cloud Formation:**

- Cloud formation is the pure definition of infrastructure as code
- We can convert the application's architecture into JSON formatted template, so our architecture is literally code
- We can use that JSON template to deploy updated copies of that architecture to multiple regions

### **Benefits:**

Saves time – we don't have to manually create duplicate architecture in additional regions.

We can version control our infrastructure

Allows backup of infrastructure

Great solution for disaster recovery

Sample cloudformation template to create s3 bucket:

Cat cf1.template

```
{  
  "AWSTemplateFormatVersion" : "2017-07-02",  
  "Description" : " this is a s3 bucket created from CF",  
  "Resources" : {  
    "S3Bucket" : {  
      "Type" : "AWS::S3::Bucket",  
      "Properties" : {  
        "AccessControl" : "PublicRead",  
        "BucketName" : "bucket-cf-1234"
```

}

}

}

}

## **AWS Levels of Certifications:**

### **Associate level:**

Solution Architect Associate

Developer Associate

Sysops Associate

### **Professional level:**

Solution Architect Professional

Devops Professional

### **Special Certifications:**

bigdata certification

network certification