A Practical Guide to Configuring

AWS

(Amazon Web Services)

Cloud Platform

Lab Manual

Introduction

We are pleased to release the practical guide to configuring AWS (Amazon Web Services). This lab manual can be used as a standalone guide or in conjunction with the AWS course.

The list of exercises ranges from the basic to the advanced, with each exercise building over the one before it. All the steps are clearly outlined with screenshots so that students can practically work through the manual by themselves.

Each of the exercises is divided into four sections:

1. Objective
2. Prerequisite
3. Topology
4. Tasks

We hope this practical guide will be a useful addition to an IT professional's collection, providing reliable step by step how-tos for general AWS configuration. Any feedback or suggestions to improve this would be gratefully accepted.

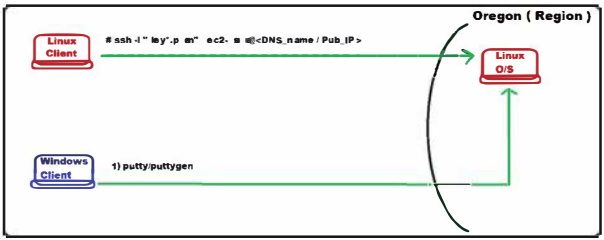
Table of Contents

Lab 1: To Launch Amazon Linux EC2 instance

OBJECTIVE

To Launch Amazon Linux instance and to connect from linux and windows client PC.

TOPOLOGY



Note : This lab helps to launch your first instance quickly, so it doesn't cover all possible options.

PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK :

Launch Amazon Linux instance

Select Region

Select Amazon Machine Image (AMI)

Create key pair

Connect to Amazon Linux instance from linux client PC using ssh.

Connect to Amazon linux instance from from Windows client PC using putty/puttygen

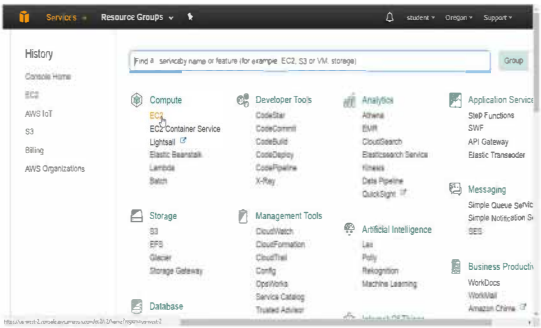
Start/stop/terminate instance

1. To Launch Amazon Linux instance in default VPC

Open the Amazon EC2 console

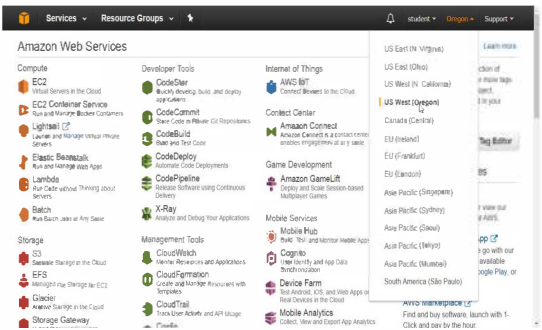
Select **Compute**

Click on **EC2** service



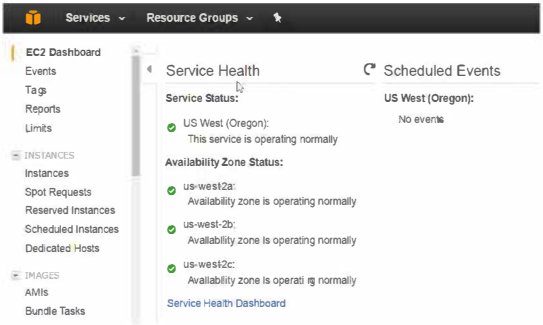
Select the **Region**, **" US West (Oregon) "**

Note: Select the region which is nearest to your Geographical Location.



To check **Service Health**

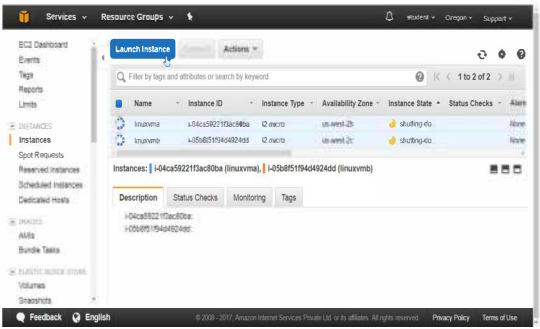
Drag down and check **Service Status**&**Availability Zone Status**:



From the "**EC2 Dashboard"** panel

Select Instance

Click on "**Launch Instance"** button

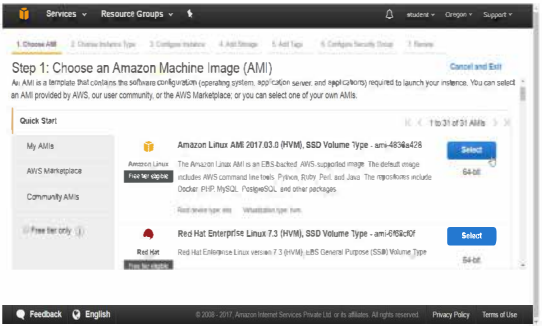


On **"Choose an Amazon Machine Image (AMI)"** page

Select "Quick start"

Select **"Amazon Linux AMI"** and click **select** button

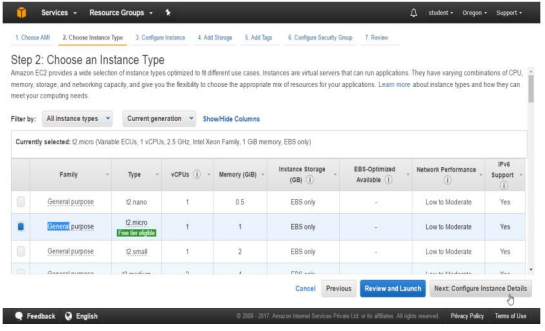
[Notice that this AMI is marked **"Free tier eligible."**]



On **“Choose an Instance Type”** page

Select type **“t2.micro”,** eligible for the free tier.

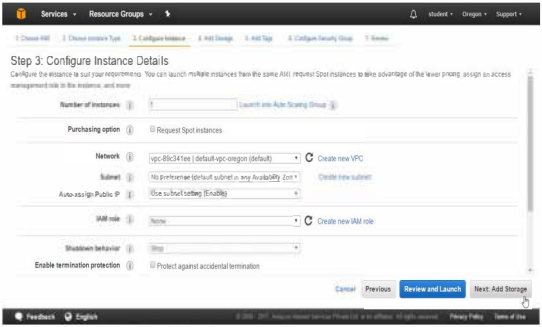
Click on **“Next: Configure Instance Details”** button



On **"Configure Instance Details",** page

Leave all values as default

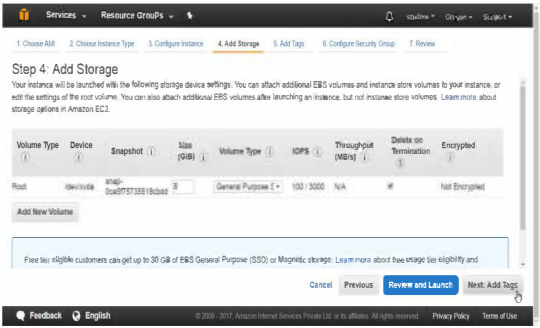
Click on "**Next: Add storage"** button



On "Add Storage", page

Leave all values as default

Click on "**Next: Tag Instance"** button



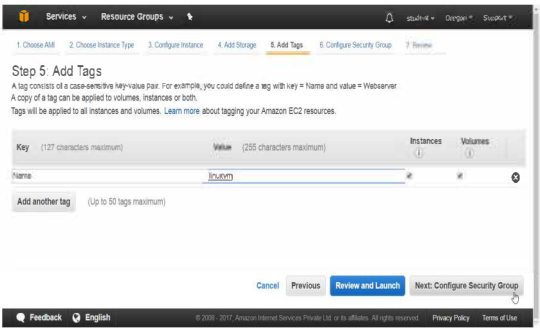
On **" Add Tags"**, page

Provide following values

Key →Name

Value →linuxvm

Click on "**Next: Configure Security Group"** button



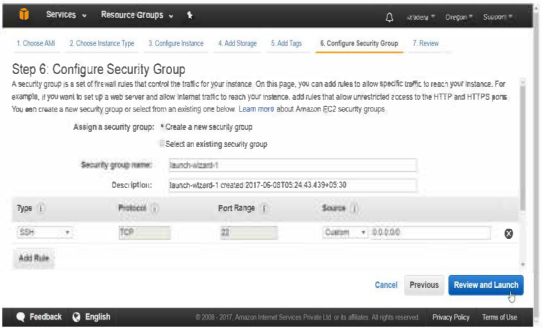
On **"Configure Security Group"** page

Select -→ Create a new security group

Leave all values as default.

Note: By default for linux instance **port 22** i.e ssh is used.

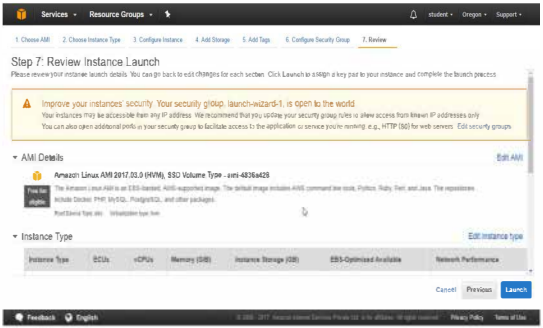
Click **"Review and Launch"** button



On **" Review Instance Launch",** page

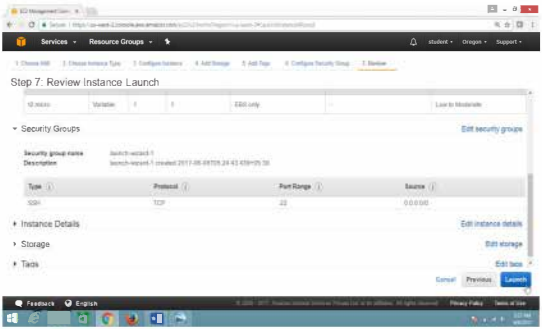
Leave all values as default.

Verify the summary, then drag down



Verify the summary

Click on **Launch** button



On **"Select an existing key pair or create a new key pair"**, box

Select **"Create a new key pair"**

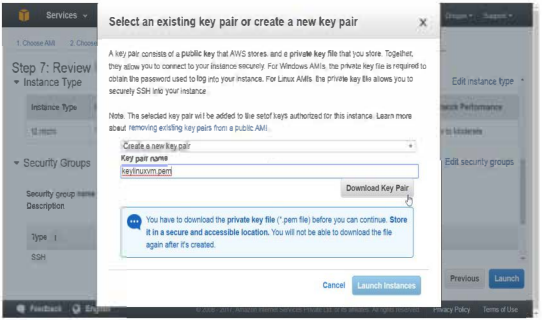
Enter **Key pair name→** keylinuxvm.pem

Click on **"Download Key Pair"**

Note: Store it in a secure and accessible location.

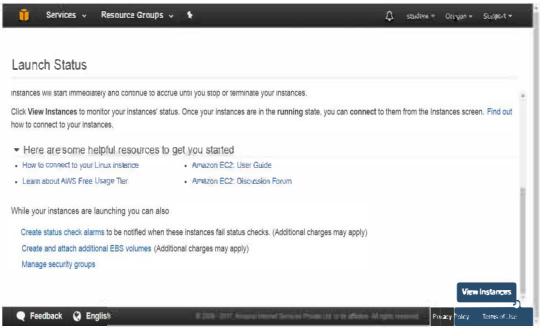
You will not be able to download the file again after it's created.

Click on "**Launch an instance"**



On **Launch Status** page, go to right bottom corner

Click on "**View instances**" button

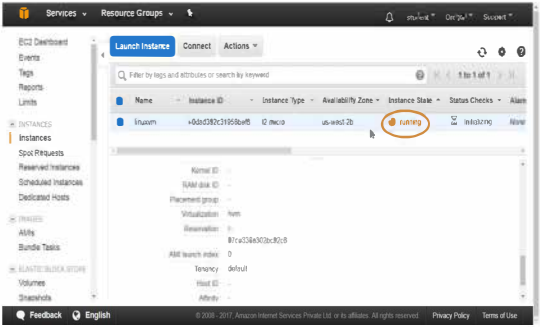


On **EC2 Dashboard** panel

Click on Instances,

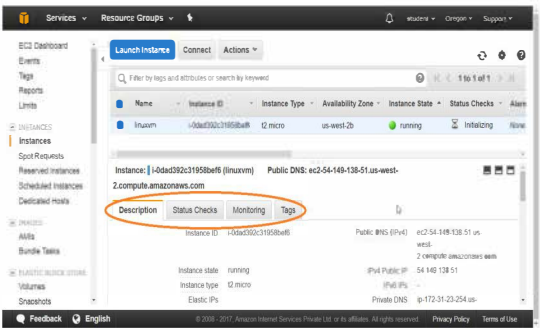
Select instances

Check instance status →running



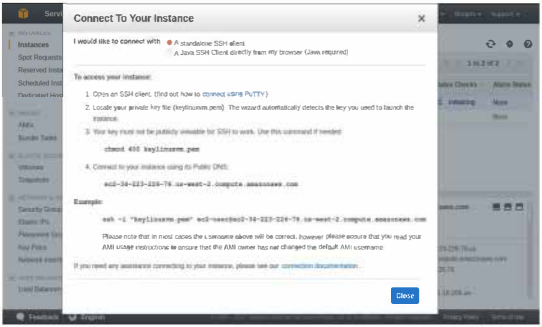
To check instance details like

Description, Status check, Monitoring, Tags



1a) To connect to "Amazon linux instance" from linux client operating system.

On "**Connect To Your Instance**" page see the guide lines to connect to linux instance.



Login to linux client PC, Open the terminal and run following commands.

First go to the folder where your private key file ***.pem** is stored.

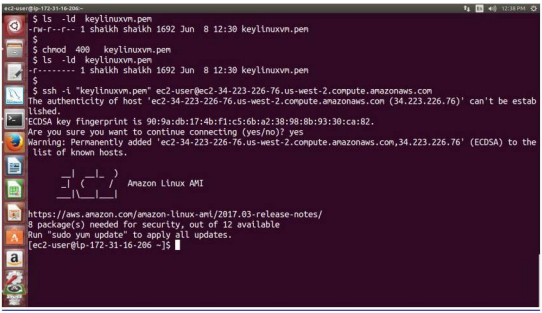
eg : keylinuxvm.pem

ls

ll

chmod **400** keylinuxvm.pem

ssh -i "keylinuxvm.pem" ec2-user@ec2-54-191-200-74.us-west-2.compute.amazonaws.com



Note : ec2-user is the default user for this instance

To know current user in linux

$ whoami

To switch to root user in linux

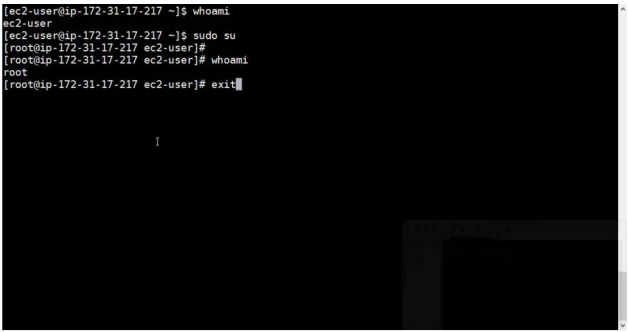
$ sudo su

Verify (root user)

whoami

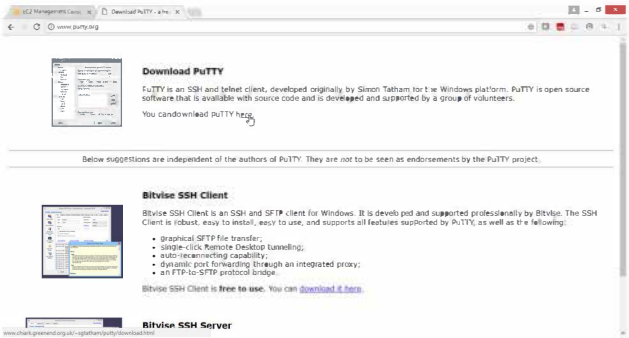
To logout

exit



1b) To connect to "Amazon linux instance" from Windows Client Operating System.

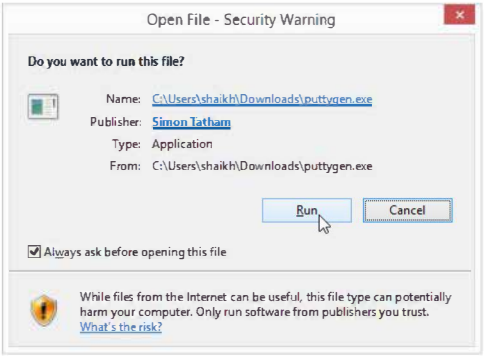
Download**putty.exe** and **puttygen.exe** from **putty.org** website



Note: Because putty cannot understand .pem file format, so use puttygen.exe to converting *.pem file into *.ppk format

Click on puttygen.exe file in windows operating system

Click on Run

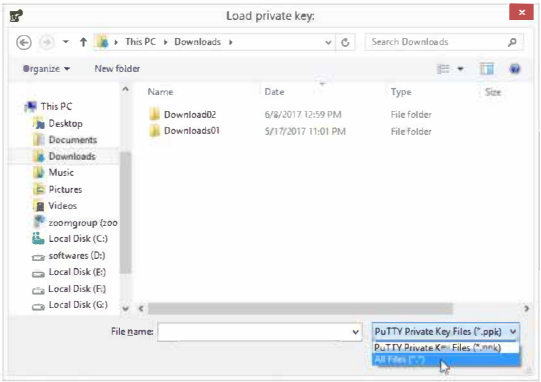


Click on **Load** button

Note: By default, PuTTYgen displays only files with the extension .ppk

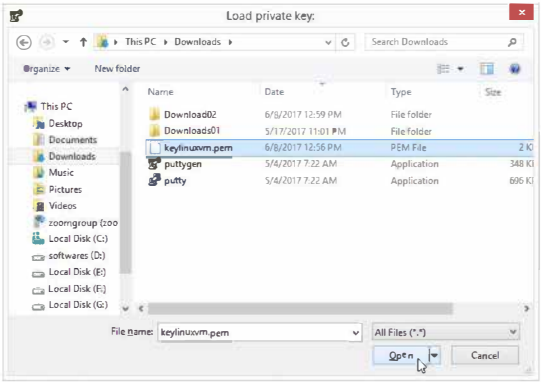
So to locate your .pem file

On **file names** Select →**All files (*.*)**

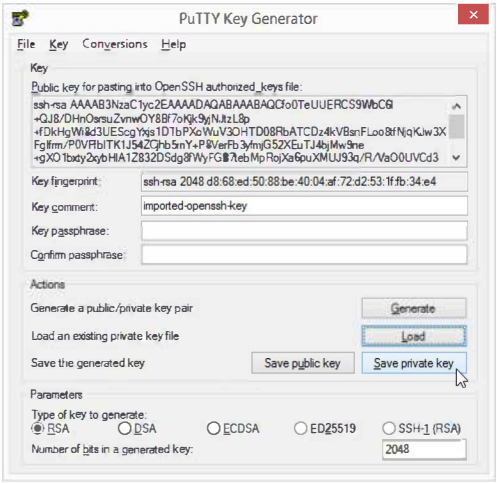


Locate keylinuxvm.pem in your folder

Click on open

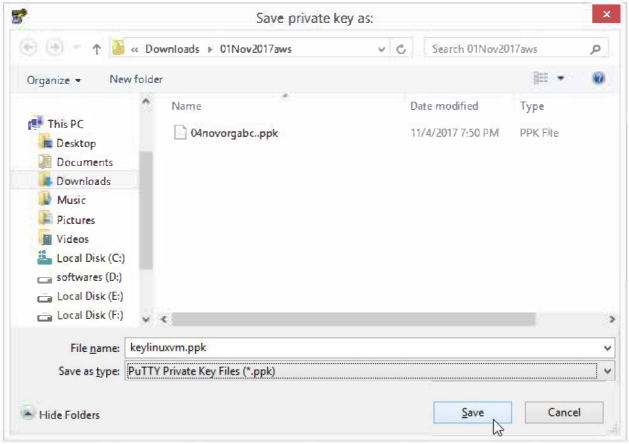


Click on **"Save private key"** button



Save the file → keylinuxvm.ppk

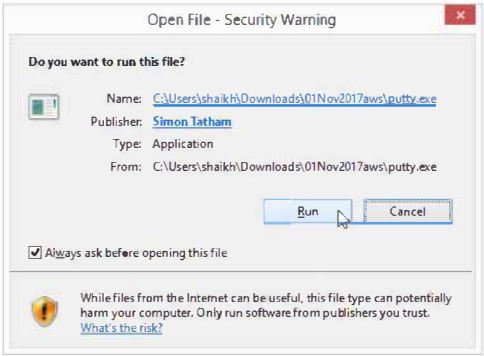
Click on Save button



To connect to linux instance Run putty.exe from windows operating system.

Run putty.exe

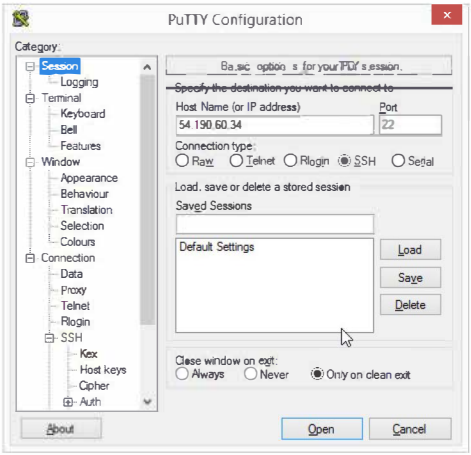
Click on Run



On **Category** page provide following values

Host Name (or IP address) → Provide public IP or DNS name of the instance

Port → 22

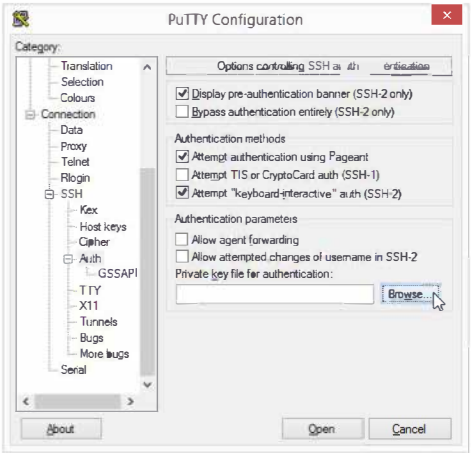


Under Connection expand

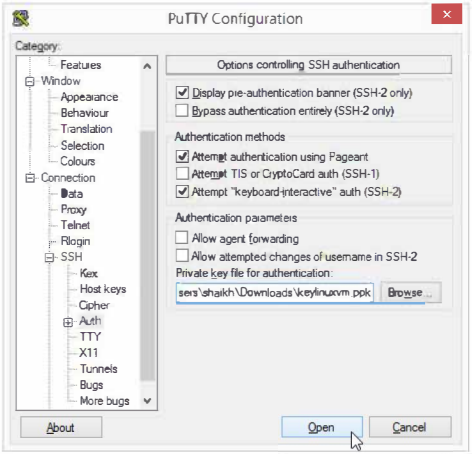
Click on SSH → Auth

Select Browse button

Provide the path of *.ppk file



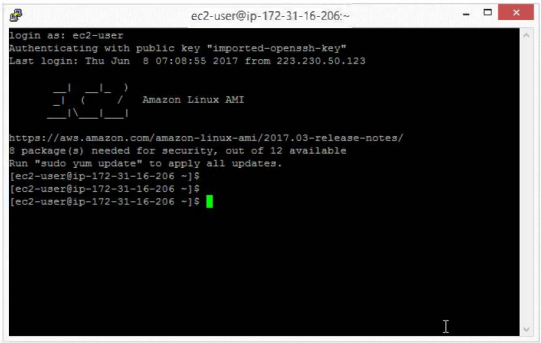
Click on Open button



Verify

Putty login screen is for linux

Provide user name **ec2-user**



Now you had logged in as ec2-user in Amazon Data Center Linux Machine.

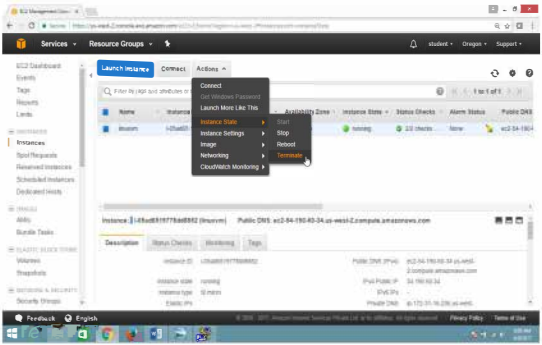
To start/stop/terminate instance

On Ec2 Dashboard

Select the Instance

Drop down on **Action** button

Select **Instance state** to **Start/Stop/Reboot//Terminate** the instances.



Note:

If you are not going to use the instance, terminate the instance,

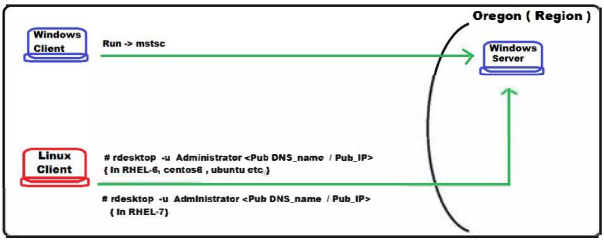
Otherwise it will be charged if the limit is over after free tier usage.

Lab 2: To Launch Windows EC2 instance in AWS

OBJECTIVE

To Launch Windows instance and to connect from windows and linux client PC.

TOPOLOGY



Note : This lab helps to launch your first Windows instance quickly, so it doesn't cover all possible options.

PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK :

To Launch Windows instance

Select Region

Select Amazon Machine Image (AMI)

Create key pair

Connect from Windows operating system

Connect from Linux Operating system

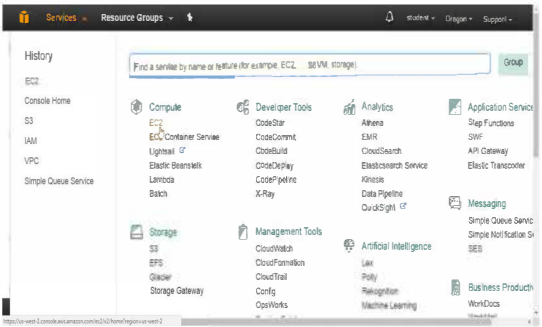
Start/stop/terminate instance

1. To Launch Windows instance in default VPC

Open the Amazon EC2 console

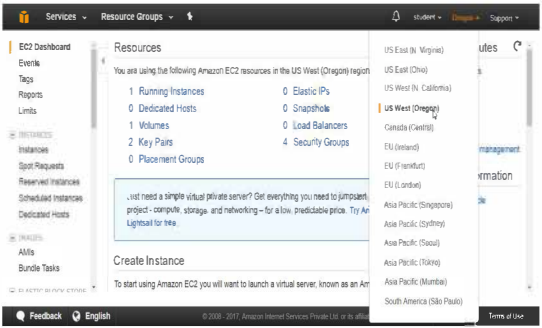
Select **Compute**

Click on **EC2** service



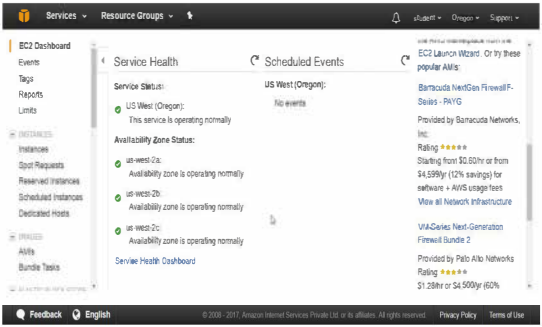
Select the **Region**, **"US West (Oregon) "**

Note: Select the region which is nearest to your Geographical Location.



To check **Service Health**

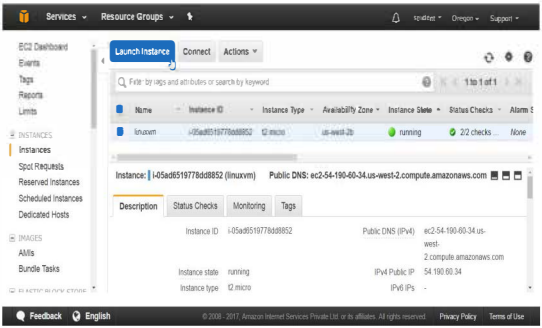
Drag down and check **Service Status**&**Availability Zone Status**:



From the **"EC2 Dashboard"** panel

Select Instance

Click on **"Launch Instance"** button



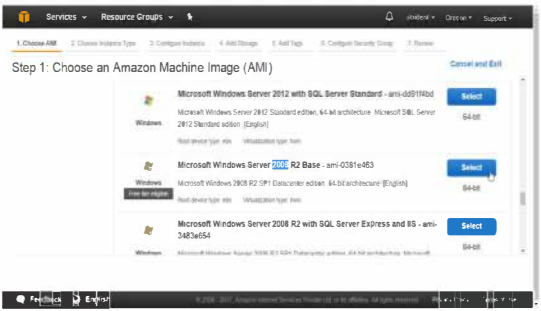
On "**Choose an Amazon Machine Image (AMI)**" page

Select "Quick start"

Select **"Microsoft Windows Server AMI"** and click **select** button

[Notice that this AMI is marked **"Free tier eligible."**]

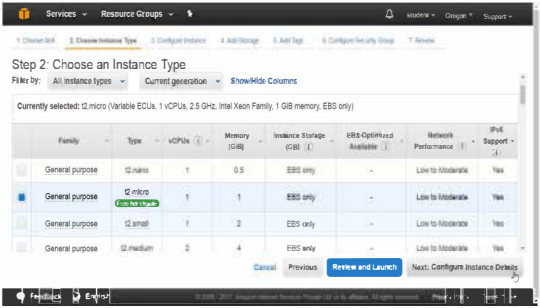
Click on **Select** button



On **"Choose an Instance Type"** page

Select type **"t2.micro",** eligible for the free tier.

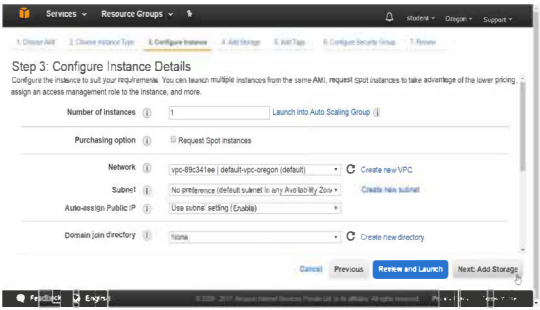
Click on **"Next: Configure Instance Details"** button



On **"Configure Instance Details",** page

Leave all values as default

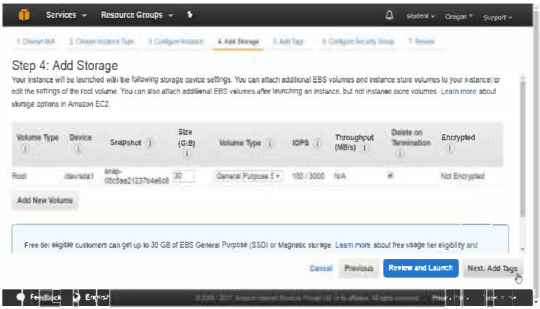
Click on "**Next : Add storage"** button



On "Add Storage", page

Leave all values as default

Click on "**Next: Tag Instance"** button



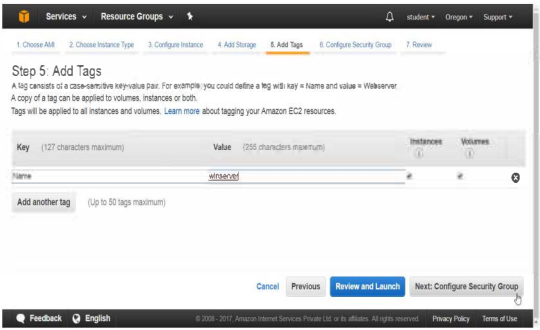
On **" Add Tags"**, page

Provide following values

Key →Name

Value →winserver

Click on "**Next: Configure Security Group"** button



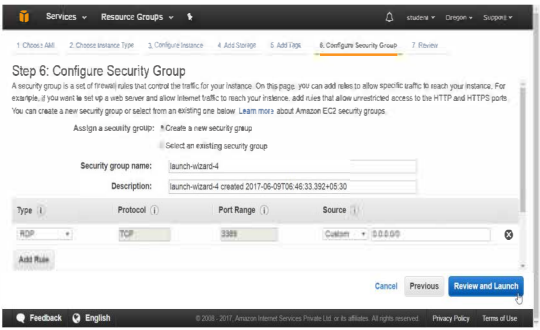
On **"Configure Security Group"** page

Select → Create a new security group

Leave all values as default.

Note: By default for linux instance **port 3389** i.e RDP is used.

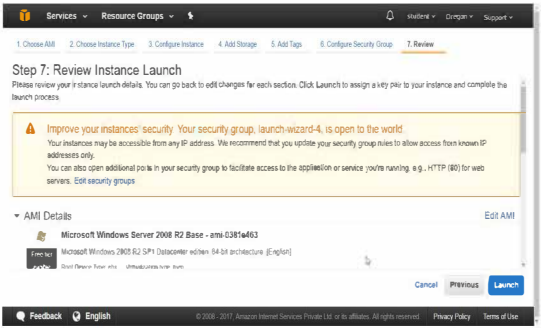
Click **"Review and Launch"** button



On **" Review Instance Launch",** page

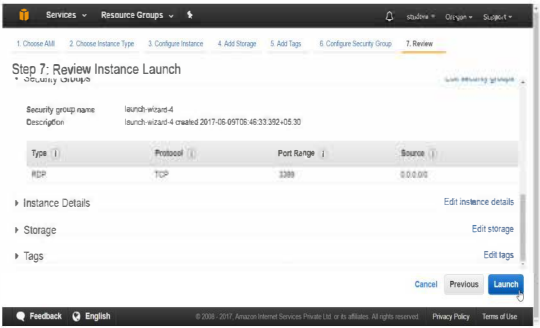
Leave all values as default.

Verify the summary, then drag down



Verify the summary

Click on **Launch** button



On **"Select an existing key pair or create a new key pair"**, page

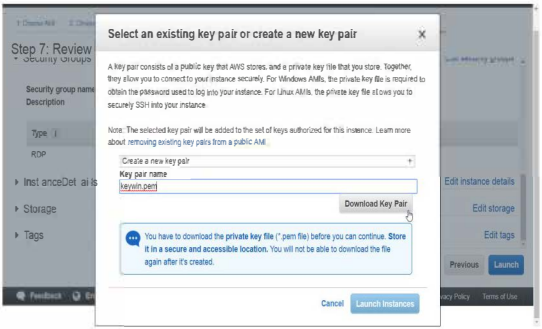
Select **"Create a new key pair"**

Enter **Key pair name→** keywin.pem

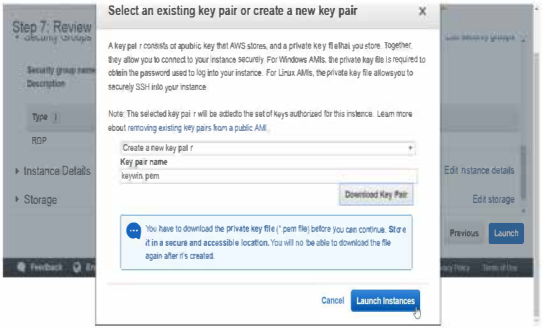
Click on **"Download Key Pair"**

Note: Store it in a secure and accessible location.

You will not be able to download the file again after it's created.

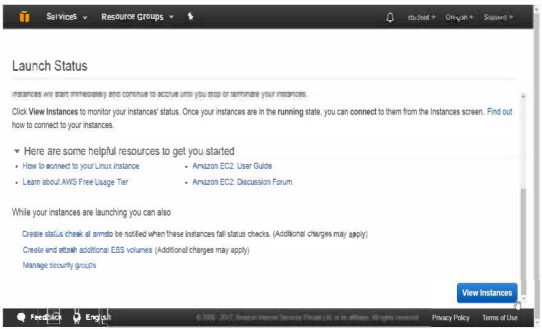


Click on "**Launch an instance"**



On **Launch Status** page, go to right bottom corner

Click **"View instances"** button

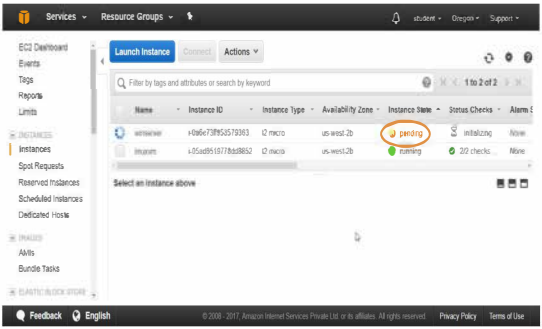


On **EC2 Dashboard** panel

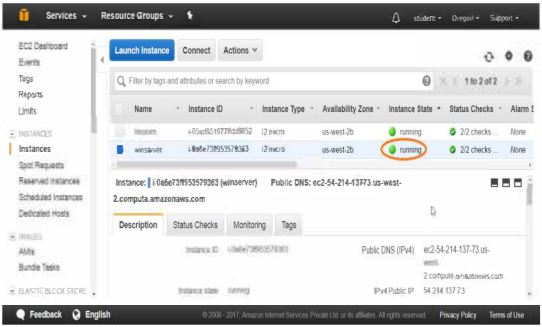
Click on Instances

Select instances

Check instance state →pending

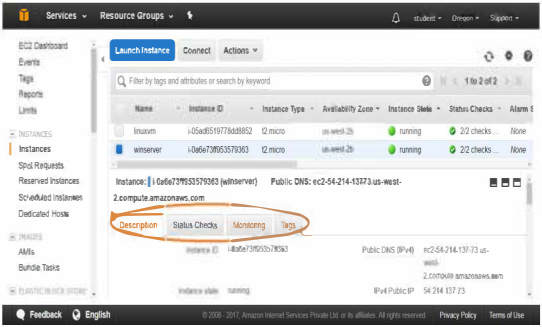


Once instance starts state is →**running**



To check instance details like

Description, Status check, Monitoring, Tags



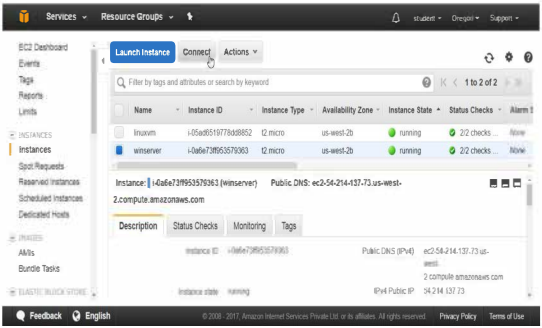
2 a) To connect to "Windows instance" from Windows client operating system.

Open Ec2 Dashboard Console

Go to instance

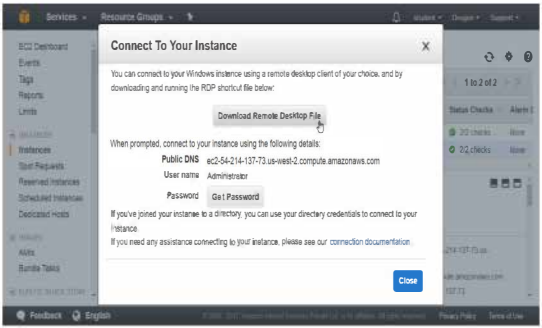
Select the instance you want to connect

Click **Connect** button

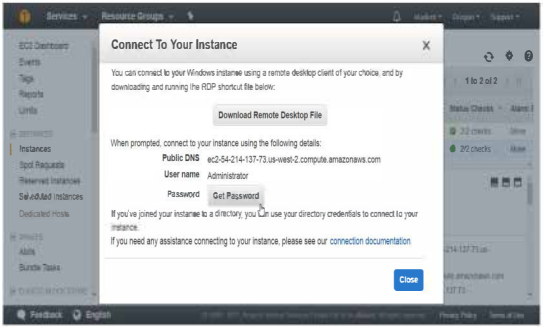


On "**Connect To Your Instance**" page, see the guide lines to connect to **W**indows instance.

Click on **"Download Remote Desktop file"** button



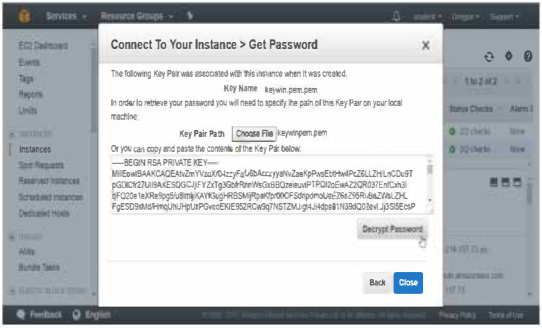
Click on "**Get Password**" button



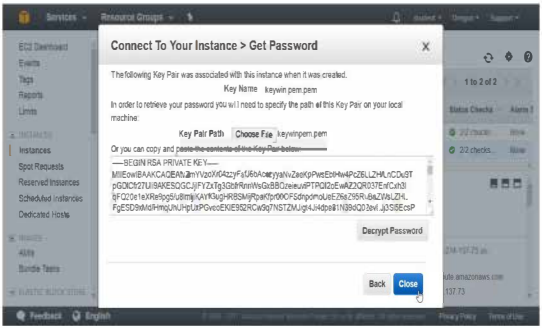
Click on **"Choose file"** button

Provide the path of key file

Click on "**Decrypt Password"** button



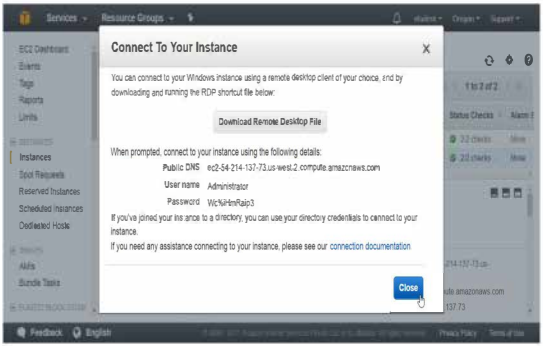
Click on **Close** button



Copy your instance Detail in Notepad

Public DNS	ec2-54-213-234-57.us-west-2.compute.amazonaws.com
User name	Administrator
Password	**********

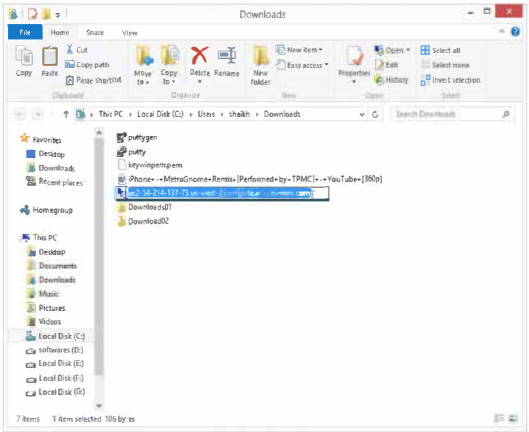
Click on **Close** button.



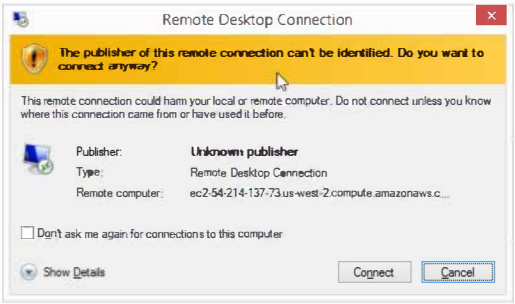
3) Now you can login to Amazon Windows instance

Double click on downloaded RDP file

Provide username as Administrator and give Password.

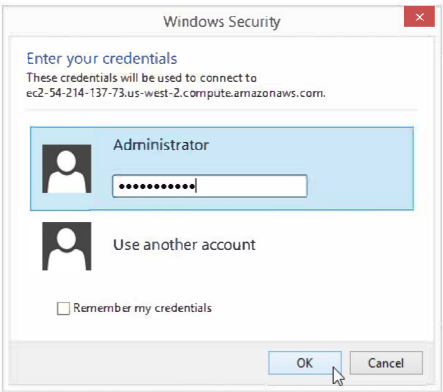


Click on connect

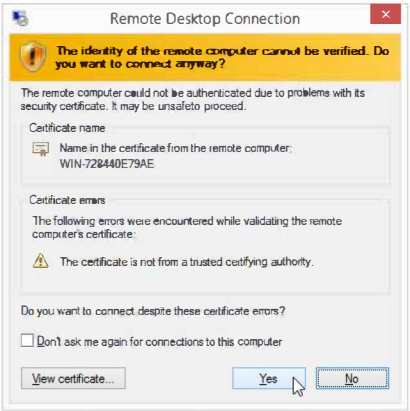


Provide username Administrator and Password

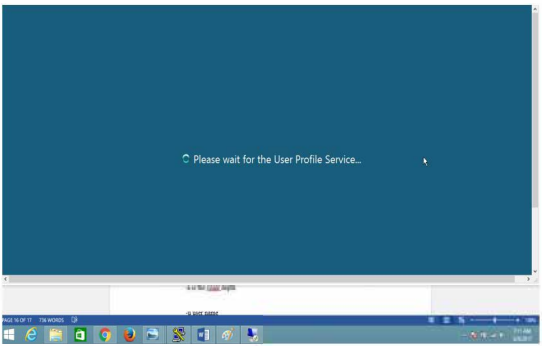
Click on OK



Click on Yes button



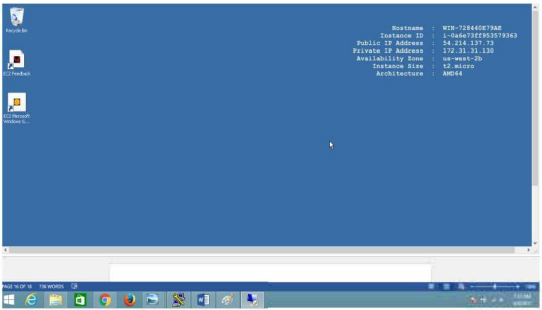
Wait for a movement



Verify

Successfully Logged in to windows instance

Check Public and Private IP of Windows instance



2b) To connect to your Windows instance using Linux client operating system.

Login to Linux client operating system

Open linux terminal

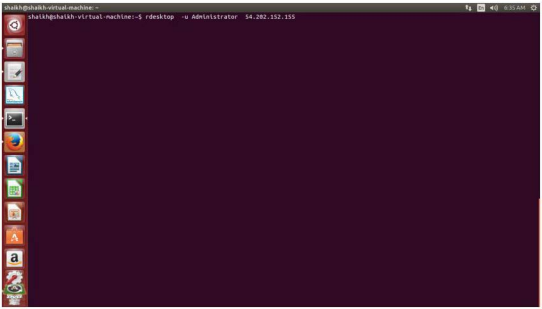
Note: rdesktop or xfreerdp { RHEL-6,7 } package should be installed

$ rdesktop -u Administrator <Pub_DNS_name / Public_IP>

or

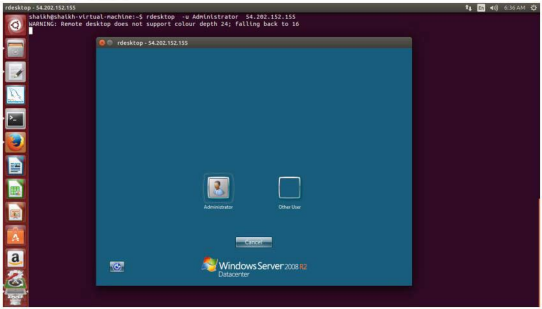
$ xfreerdp -u Administrator <Pub_DNS_name / Public_IP> { in RHEL 6,7 }

-u → user name



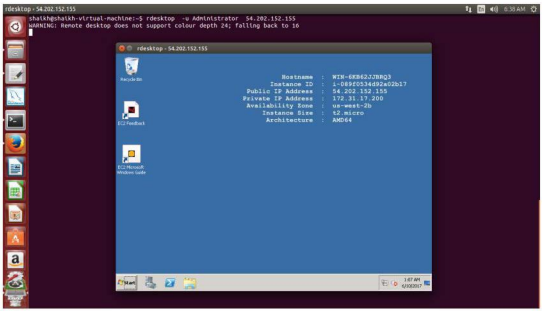
Click on Administrator

Provide the password



Verify:

Once Logged in **W**indows Desktop is available



Note:

If you are not going to use the instance, terminate the instance

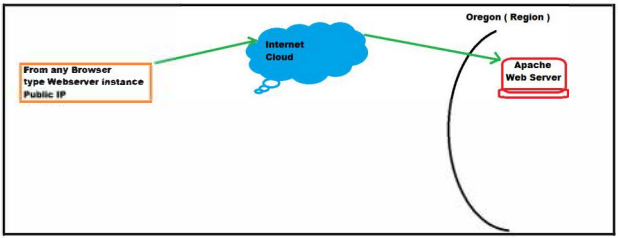
Otherwise it will be charged if the limit is over after free tier usage.

Lab 3: To Configure Webserver on Amazon Linux instance with Elastic IP

OBJECTIVE

To configure Webserver and to verify using Elastic public IP

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK :

Launch linux instance in AWS

Switch to root user

Configure Apache Webserver

Enable HTTP port in security Group

Open the browser and provide public IP or DNS_name of Webserver

Assign an Elastic IP

Releasing an Elastic IP

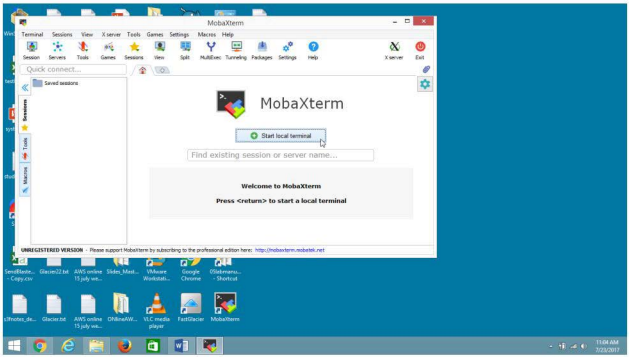
1) Launch Amazon linux instance and login to your instance

Refer to Lab [How to configure amazon linux instance]

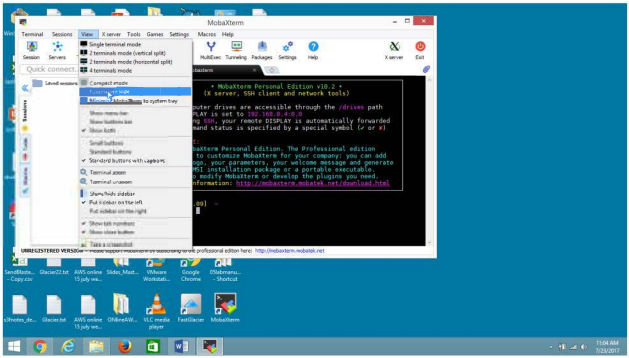
2) Connect to linux instance from windows using MobaXterm

Open **MobaXterm**

Click on **Start local terminal**

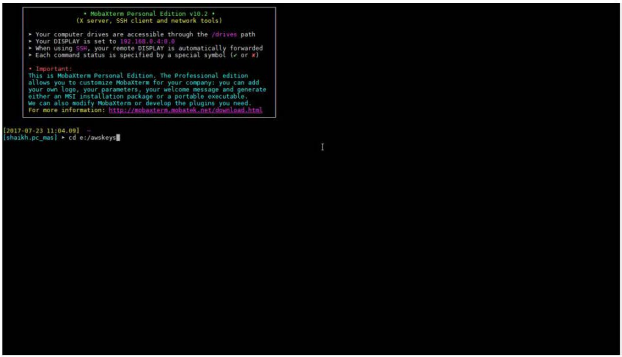


Go to Full Screen mode



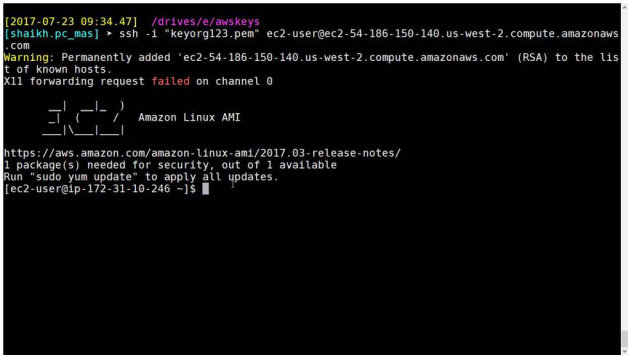
Navigate to the folder where key*.pem file is stored

Eg : cd e:/awskeys



Login to linux instance by typing the following command

ssh -i "keyorg123.pem" ec2-user@ec2-54-186-150-140.us-west-2.compute.amazonaws.com



Switch to root user

Type "sudo su"

Configure Apache Webserver run following commands as shown in the screen

```
[root@ip-172-31-10-246 ec2-user]# yum install httpd -y
[root@ip-172-31-10-246 ec2-user]# chkconfig httpd on
[root@ip-172-31-10-246 ec2-user]# service httpd restart
[root@ip-172-31-10-246 ec2-user]# vi /var/www/html/index.html
```

To use vi editor

Go to insert mode by typing `**i**' and add following code in index.html file

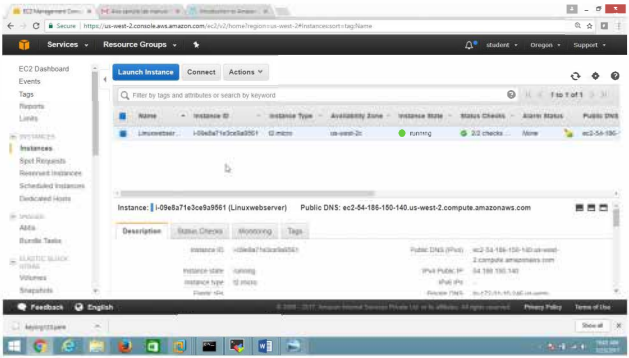
Note: [esc+shift+colon →**:wq!** (to save and quit in Vi editor)]

```
<html>
<body bgcolor=black>
<marquee>
        <font color=gold>
                <h1> Welcome to Apache Webserver in AWS instance </h1>
        </font>
</marquee>
</body>
</html>
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq!
```

3) Create an inbound Rule to Allow http traffic on port 80.

Open the AWS console
On the **EC2 Dashboard** panel

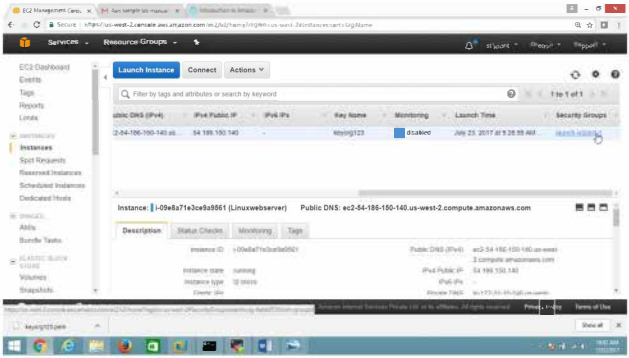
Select the linux instance



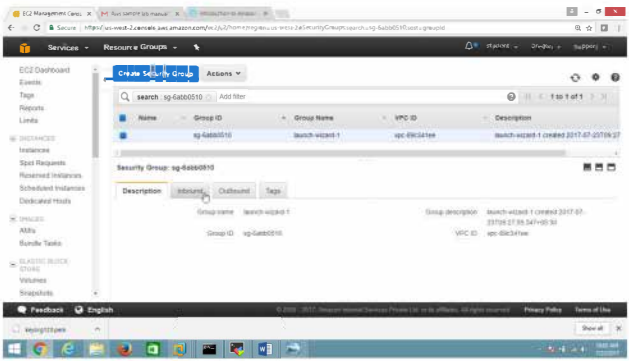
Go to the right end

Select **Security Groups**

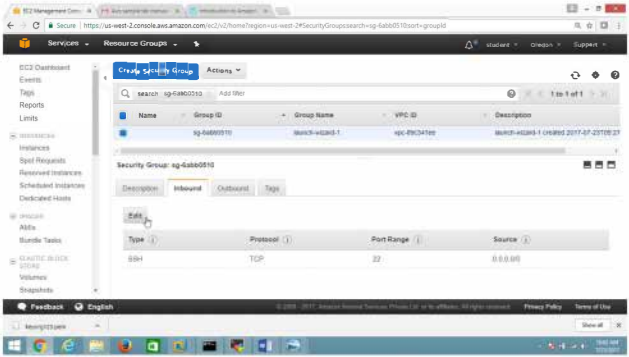
Click on **launch-wizard-1**



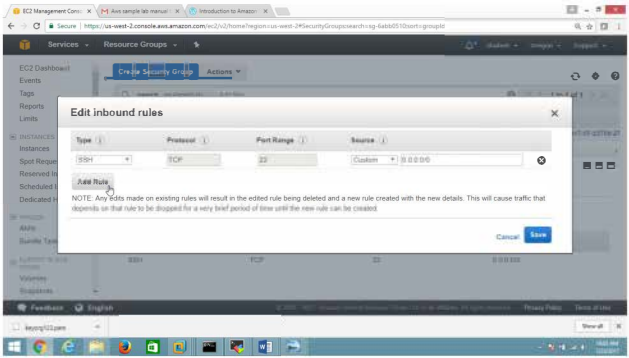
Click on **Inbound** button



Click on **Edit** button



Click on **Add Rule** button

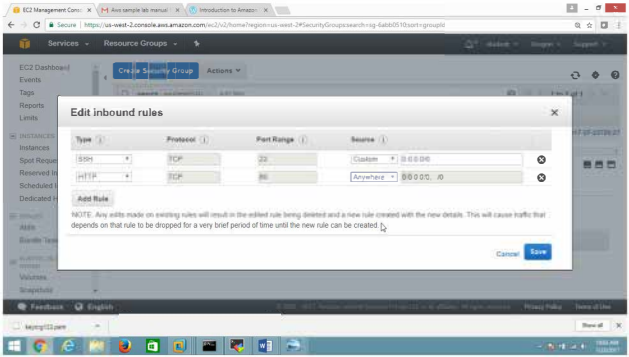


Add **HTTP** Rule

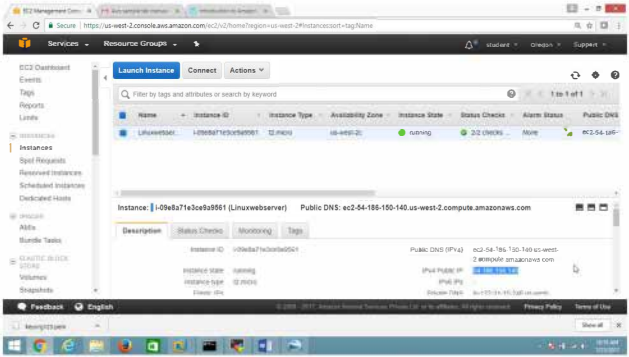
Under **Type** column select **HTTP**

Under **Source** column select **Anywhere**

Click Save button

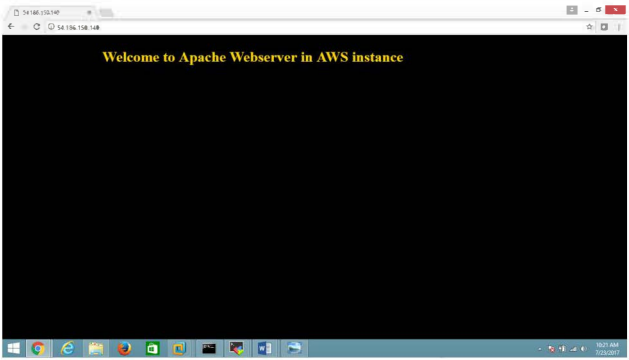


4) Open Browser and provide Webserver instance DNS_name or Public Ip



Verify

Website is running



Lab 4: To Assign Elastic IP address

Elastic IP

Note: Since public IP given by AWS is not permanent, if the instance is stopped or started again, existing public IP is released by the instance, in this case users across internet again cannot visit the same website, so to have permanent Public IP, assign Elastic IP,

Note: If your instance is terminated or not in use, and **Elastice IP** is not released then in this case it will be charged, so be careful if you are using and running under free tier usage.

Best practise is launch an instance assign Elastic IP, and before terminating release Elastic IP then terminate the instances.

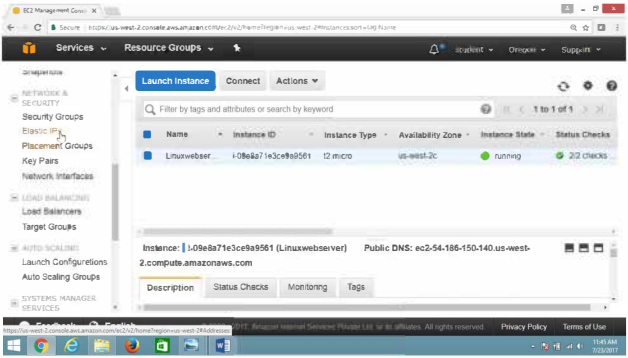
To assigning Elastic IP to an instance

Open AWS console

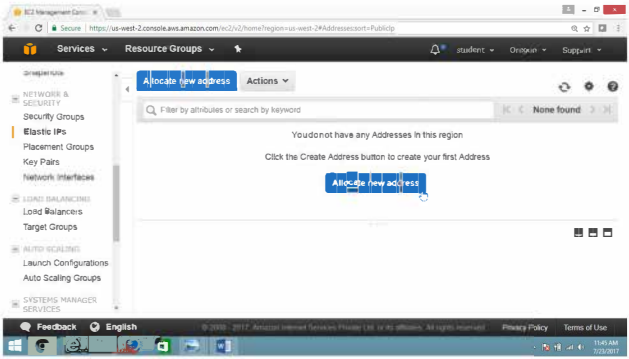
On the **EC2 Dashboard** panel

Select **"Network Security"**

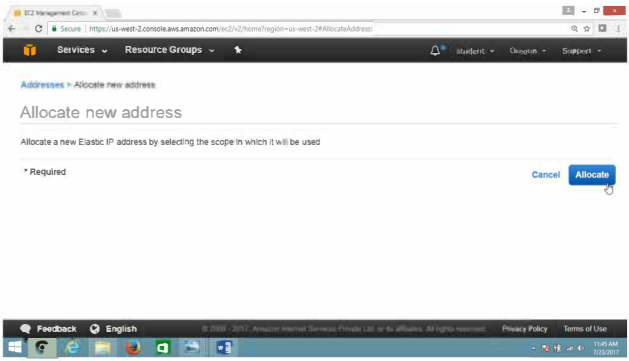
Click on **Elastic IP**



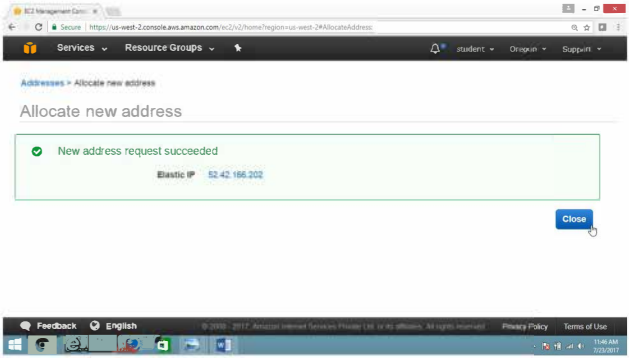
Click on **Allocate new address** button



Click **Allocate** button

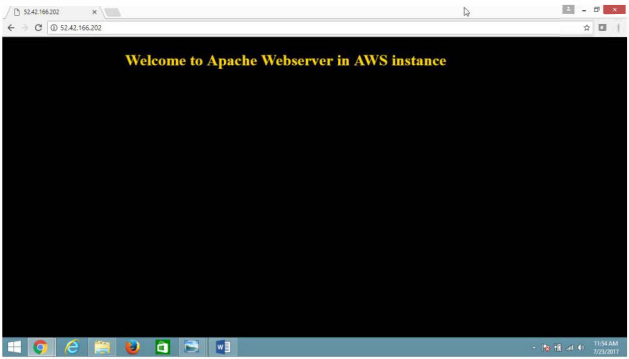


Click on **Close** button



Open your Browser and provide your instance DNS name or Elastic Public Ip

Verify website is running with elastic IP.



To releasing Elastic IP

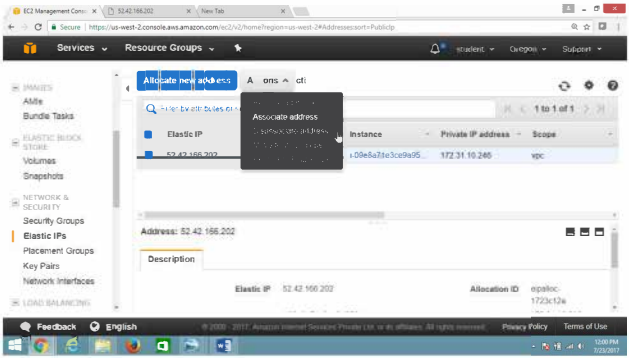
Open the console **EC2 Dashboard**

Expand "Network Security"

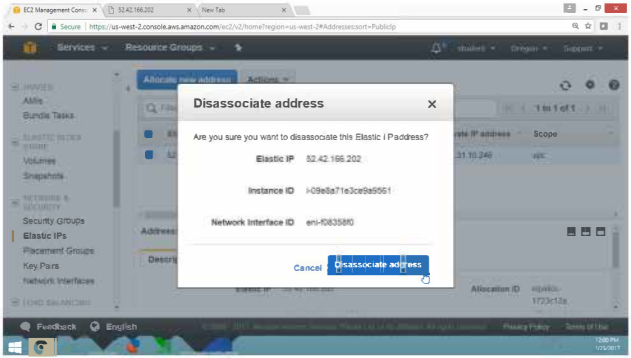
Select **Elastic IP**

Click **Action** button

Select **Disassociate Address**

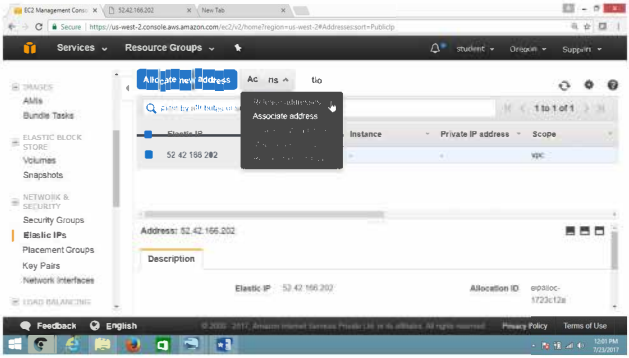


Click **Disassociate address** button

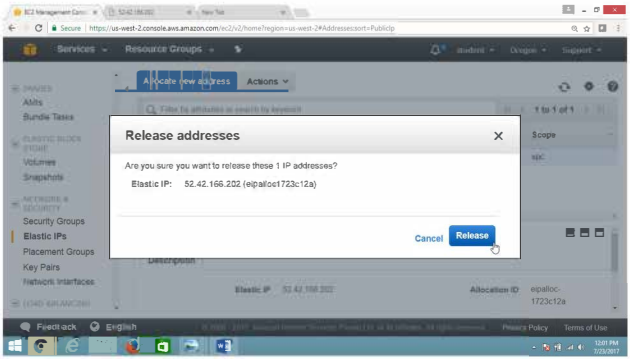


Click **Action** button

Select **Release Addresses**



Click **Release** button

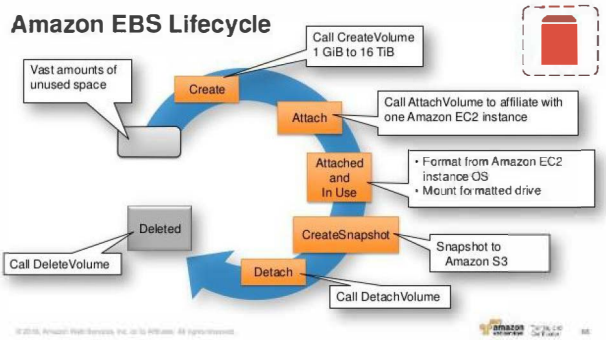


Lab 5: To Manage Elastic Block Store (EBS)

OBJECTIVE

To configure and use AWS EBS service

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

User should have basic knowledge of managing partitions in Windows or Linux

To Configure EBS With following task:

Create EBS Volume

Attaching and Detaching EBS volume.

Expanding the size of EBS volume.

Taking the snapshot of EBS volume.

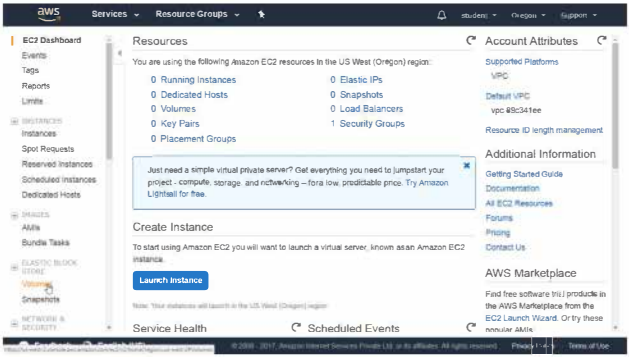
1. ***To create an EBS volume***

Open the Amazon console

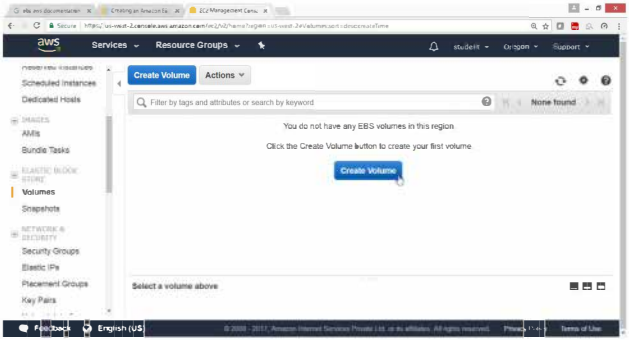
Select **Compute,** choose **EC2** service

On the **EC2 Dashboard** panel

Choose**"ELASTIC BLOCK STORE"** click on **Volumes**



Click on **Create Volumes** button



In the Create Volume dialog box,

Volume Type → General Purpose SSD (GP2)

Size (GiB) → 2 GiB

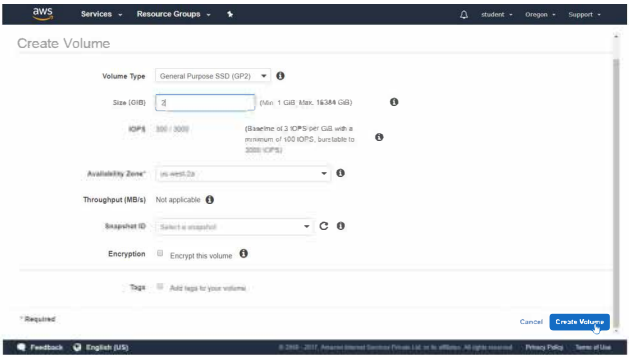
IOPS → 100 / 3000

Throughput (MB/s) → Not Applicable

Availability Zone → us-west-2a (as per your requirement)

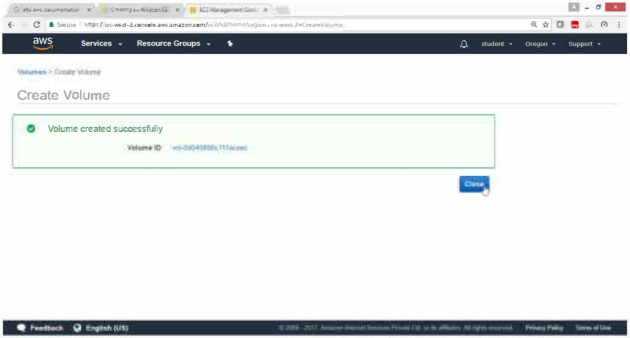
Leave remaining as defaults.

Click on **Create Volume** button



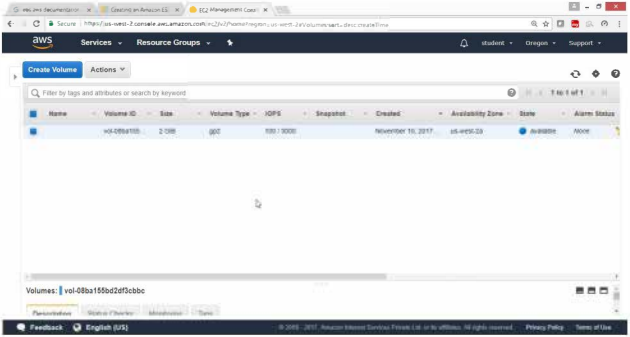
Verify Volume successfully created

Click **Close** button

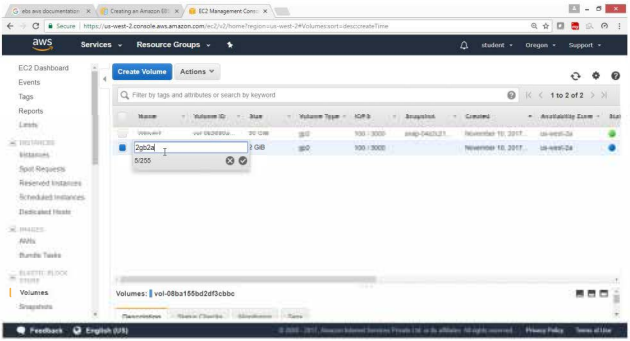


To Monitoring the State of Your Volumes

Select Volume check state → available



In the Name column give name for your volume → 2gb2a



2) To Attaching and Detaching EBS volume in Windows instance

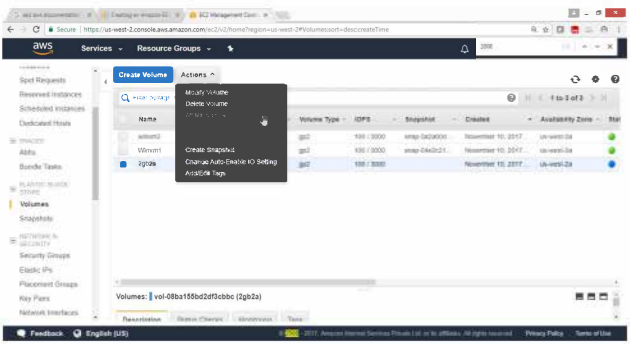
On the **EC2 Dashboard** panel

Choose**"ELASTIC BLOCK STORE"** click on **Volume**

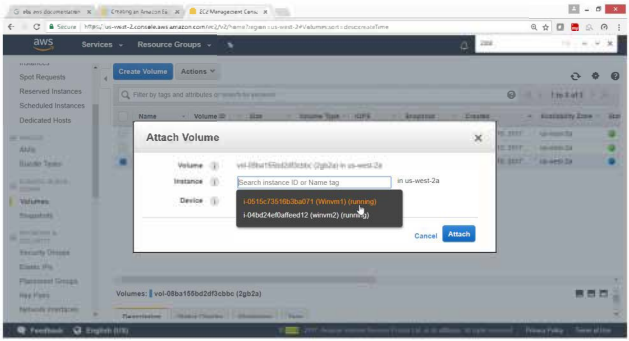
Note : The volume which you want to attach to an instance should be in same Availibility zone.

Drop Down **Action** button,

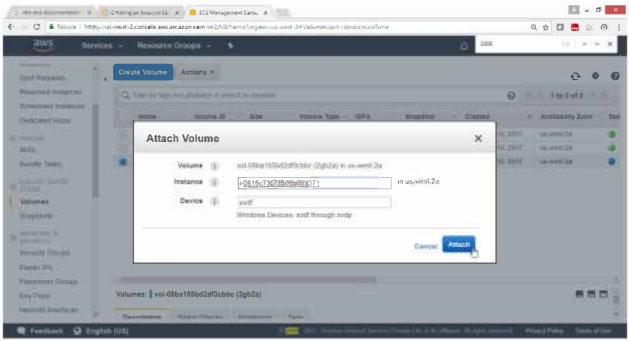
Select **Attach Volume.**



Select instance → **W**invm1



Click on **Attach**

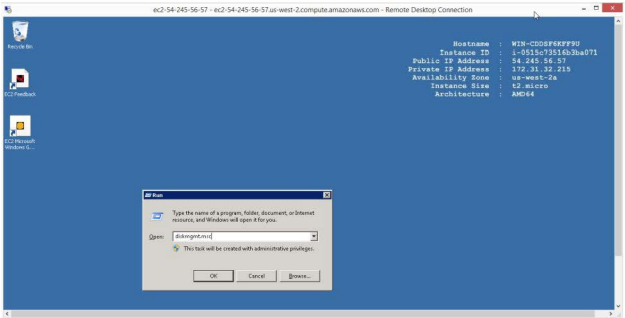


Verify the Availability of new volume

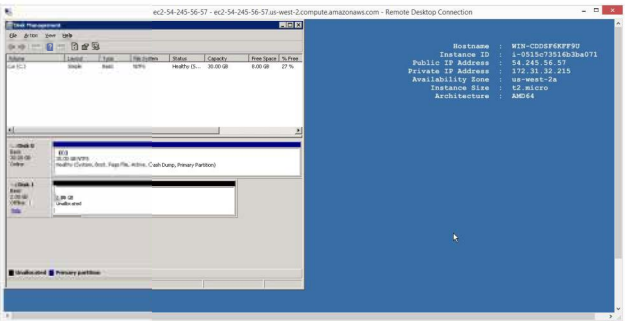
3. To check availibility of new drive login to your Windows instance.

Login to windows instance

Run → diskmgmt.msc

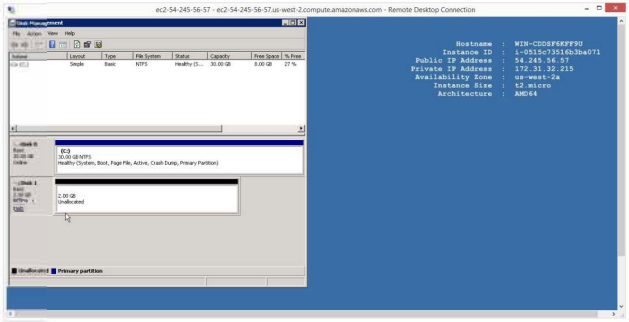


Verifies that 2 GB volume available as unallocated space

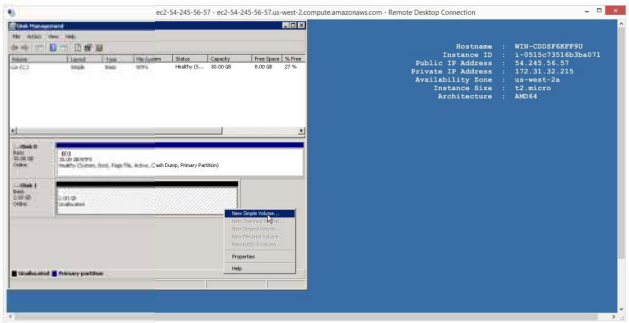


New disk is offline,

So turn it to online by right clicking and select online

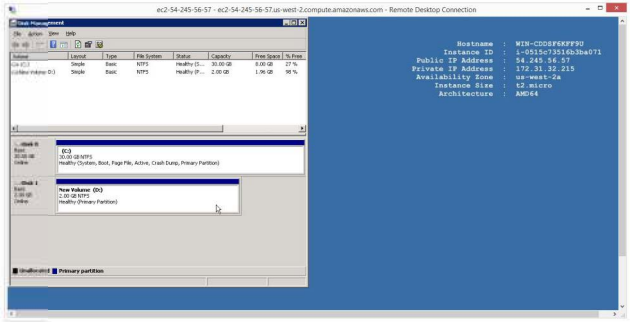


Format the unallocated disk



Verify

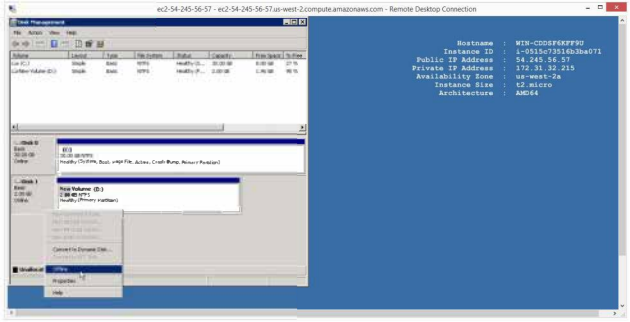
New Volume to 2GB is available to use



4. **To Detach the volume**

In Windows Select Disk 1

Right click select offline



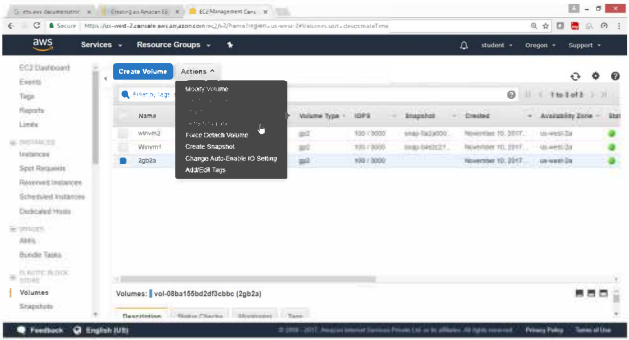
On the **EC2 Dashboard** panel

Choose**"ELASTIC BLOCK STORE"** click on **Volumes**

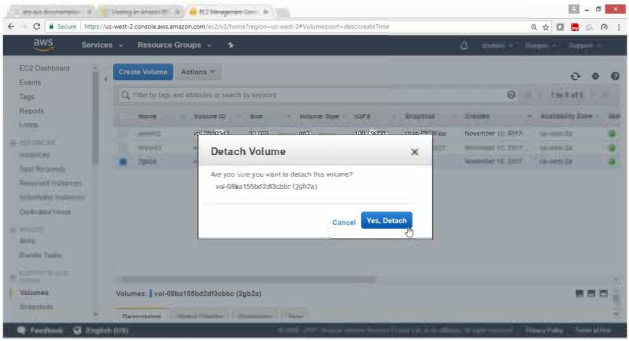
Select volume to be detached under **Name** column.

Drop Down **Action** button

Select **"Detach Volume"**



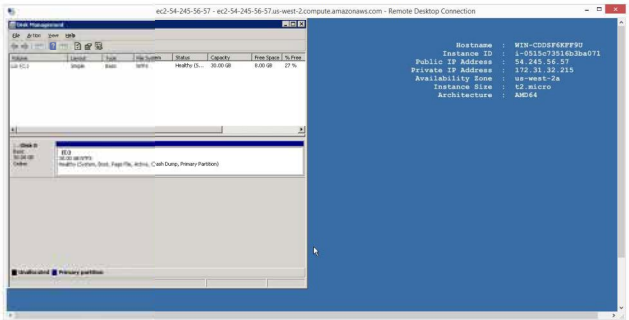
Click on **"Yes, Detach"** button



Verification

Login to windows instance

Check that D: drive is removed

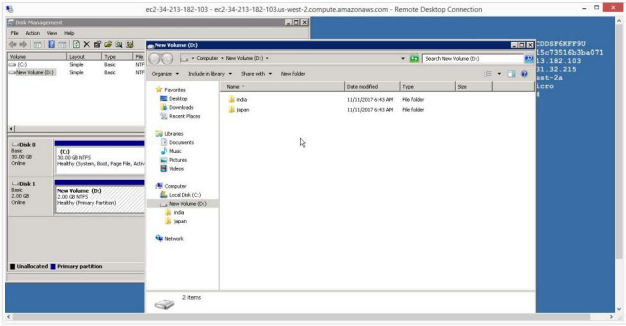


5. To Create Snapshot and Restore EBS volume.

To create a snapshot

In the current D drive two folders are available

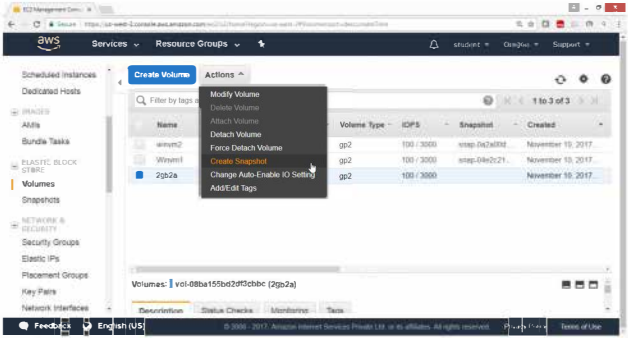
No create a snapshot of this volume



On the **EC2 Dashboard** panel

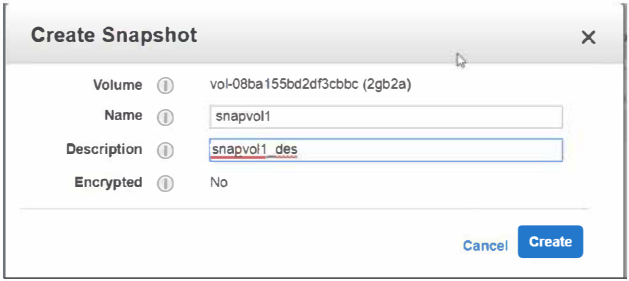
Click on **"ELASTIC BLOCK STORE"**, choose Volumes.

Drop down **Action** button select Create snapshot

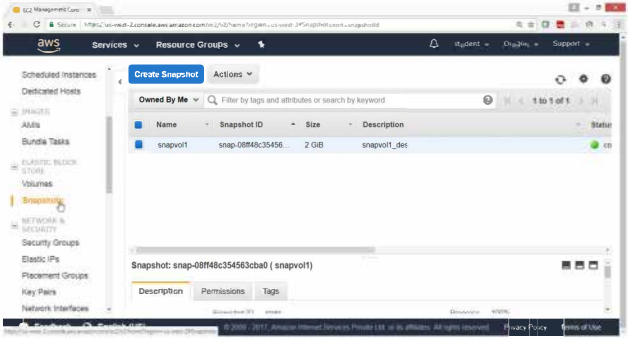


Provide snapshot details

Click **Create** button



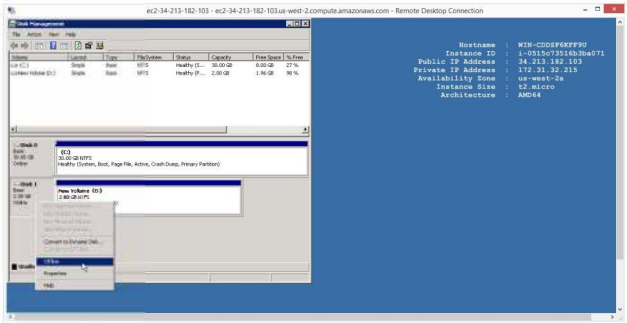
Verify that snapshot is created.



6) To Delete the volume.

First select the disk 1 from Disk Management

Right click select **offline**

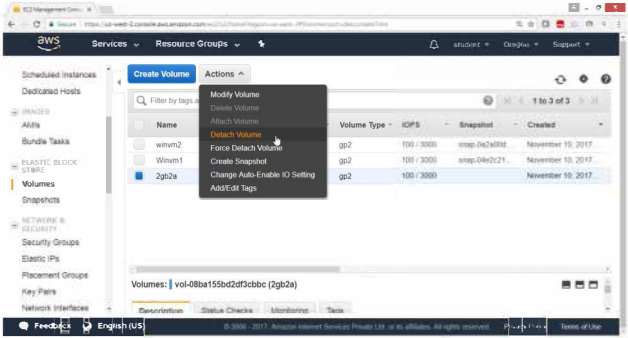


On the **EC2 Dashboard** panel

Expand **"ELASTIC BLOCK STORE"**,choose Volumes.

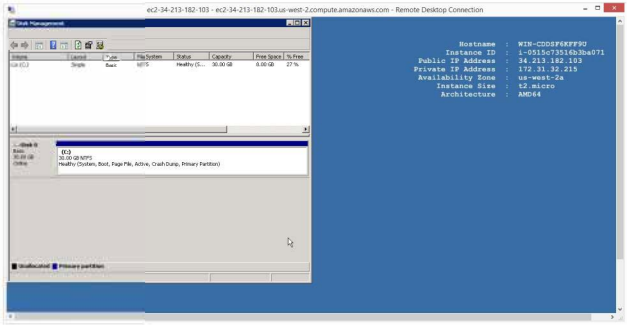
Select volume to be detached under the Name column.

Drop Down **Action** button, Select **"Delete Volume"**

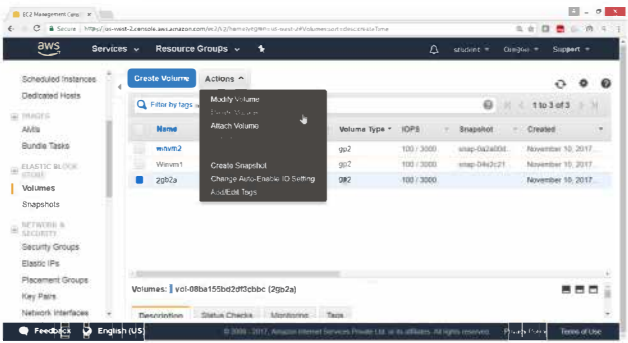


Verify from windows instance open disk Management tool

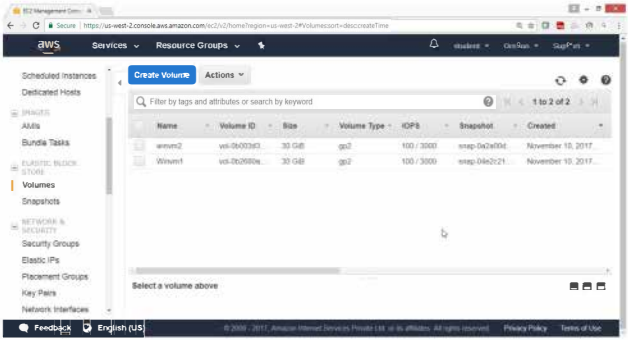
Now D drive is detached



Now delete the volume



Verify volume is deleted.



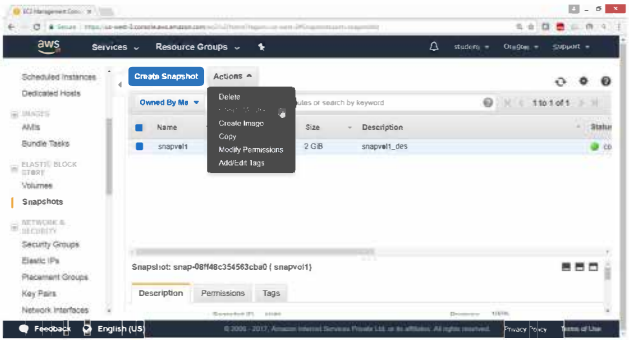
7. To Restore the volume.

From the console **EC2 Dashboard**

Expand **"ELASTIC BLOCK STORE"**, choose Snapshots

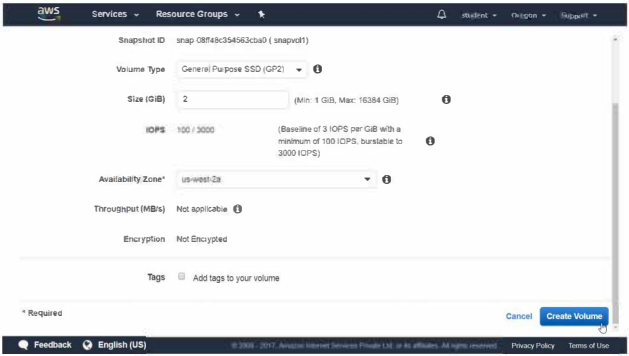
Select the snapshot

Drop Down **Action** button, Select **Create Volume**

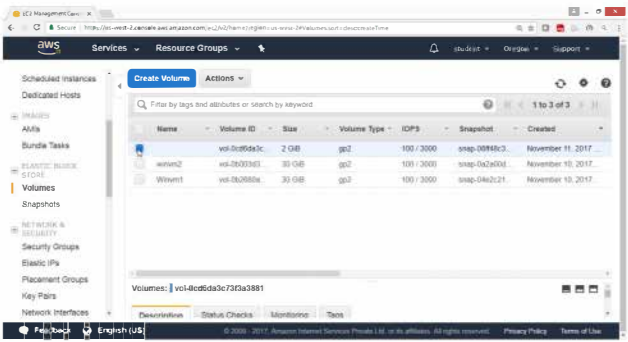


Accept the defaults values in wizard

Note: Check the right availibility zone.



Verify Volume is created



7) To expanding the size of EBS volume.

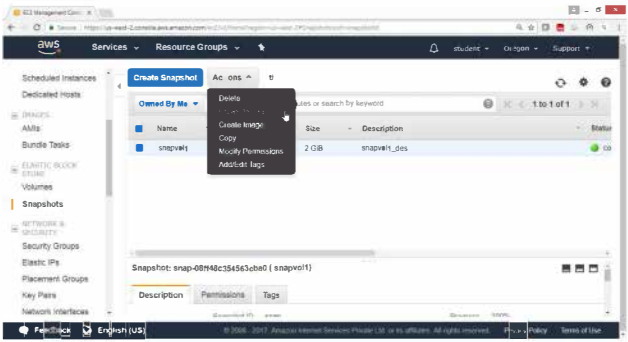
To expand EBS volume first take **snaphot** , now select the snapshot

On the **EC2 Dashboard** panel

Expand **"ELASTIC BLOCK STORE",** choose Snapshots

Drop Down **Action** button

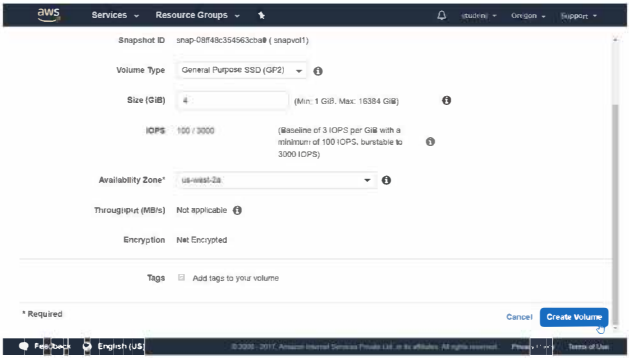
Select **Create Volume**



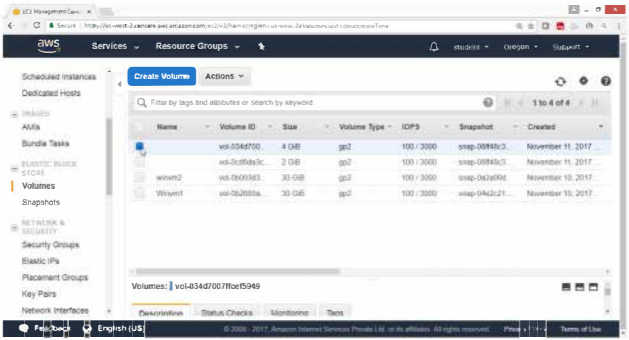
Give the required size → 4 GB

Check the right Avaibility Zone

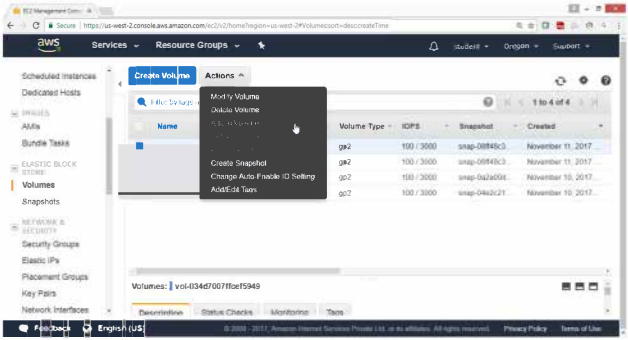
click **Create Volume** button



Verify that 4 GB is created

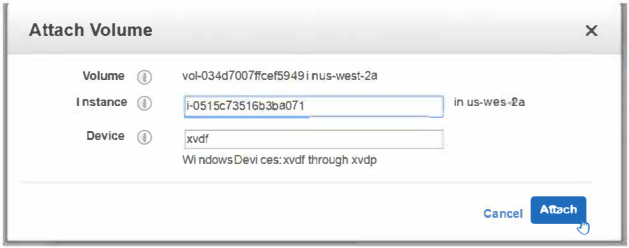


Now attach this expanded volume to your instance.

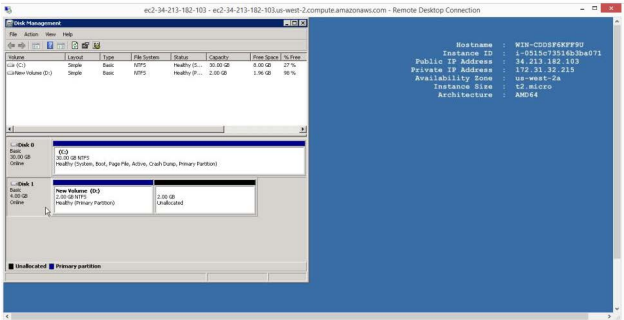


Select instance

Cli ck **Attach** button

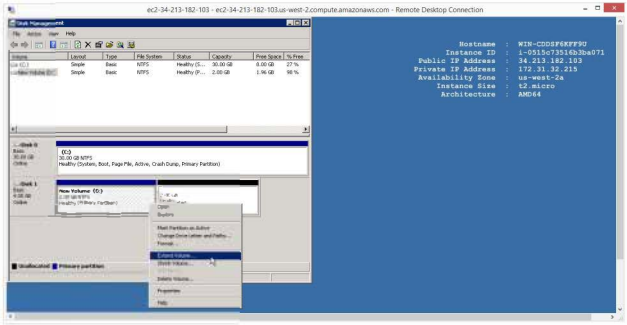


Verify 4 GB drive is available

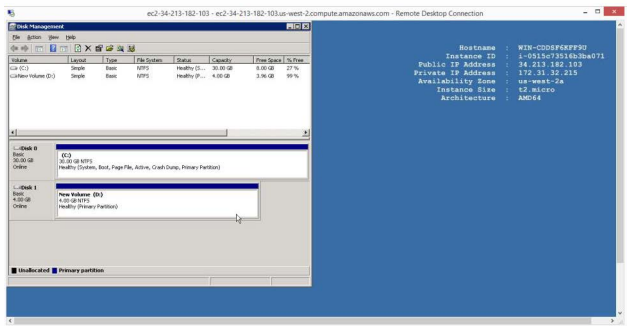


Now with respect to Windows operating system

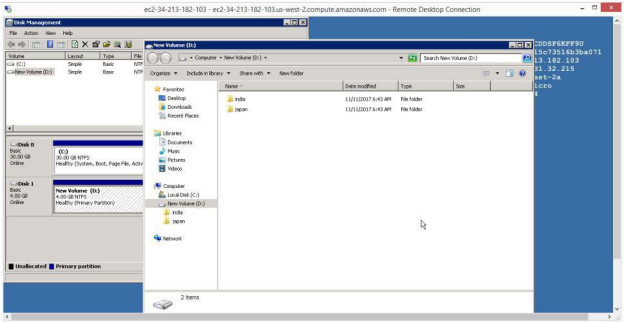
Right click on D drive extend your volume to your desired size



Verified that 4 GB volume available

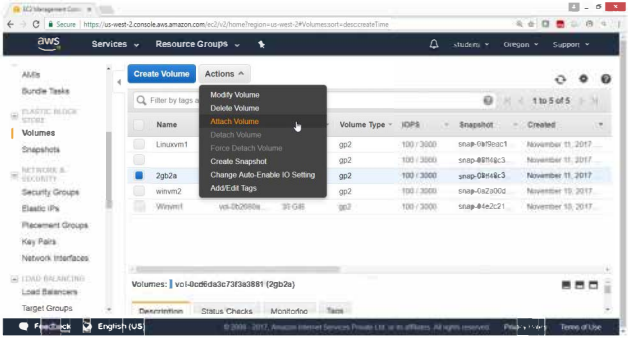


Verified that D drive contains two folders that was there in 2B drive earlier.

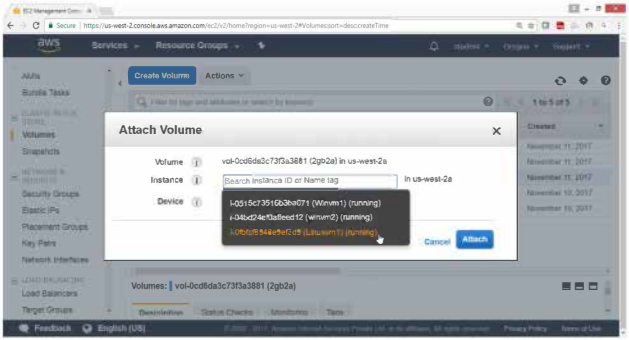


Similarly check volume in linux instance

From Action select **Attach volume**



Select Linux instance



Now connect to Linux instance

```
[2017-11-11 12:58.46]  /drives/e/awskeys
[shaikh.pc_mas] ➤ ssh -i "studentaws.pem" ec2-user@ec2-54-244-106-102.us-west-2.comp
ute.amazonaws.com
X11 forwarding request failed on channel 0
Last login: Sat Nov 11 07:28:43 2017 from 49.206.203.114

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/
[ec2-user@ip-172-31-40-234 ~]$ sudo su
[root@ip-172-31-40-234 ec2-user]#
```

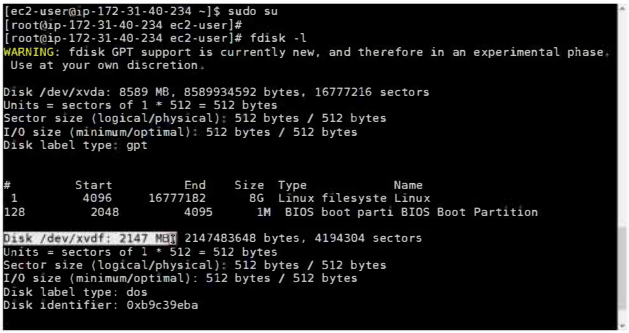
To verify

Switch to root user and run fdisk -l

$ sudo su

To check list of drives and partitions

fdisk -l



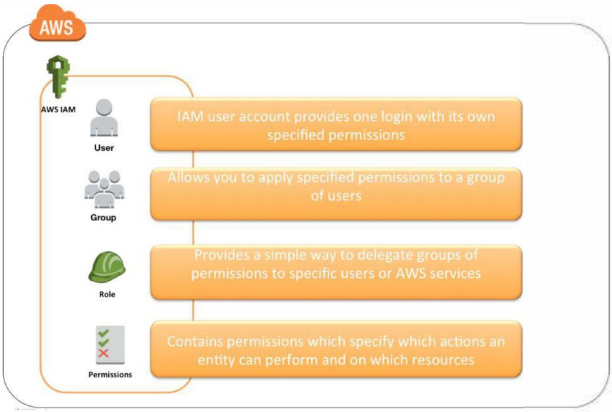
Lab 6: To Manage IAM Users, Groups and Policies

OBJECTIVE

To configure and use AWS IAM Service.

TOPOLOGY

AWS IAM Identities



PRE-REQUISTES

User should have AWS root account

To configure IAM with following task.

Create IAM users, assign password, and change password policy.

Create IAM groups.

Add users to a group.

Add policies to Groups and Users.

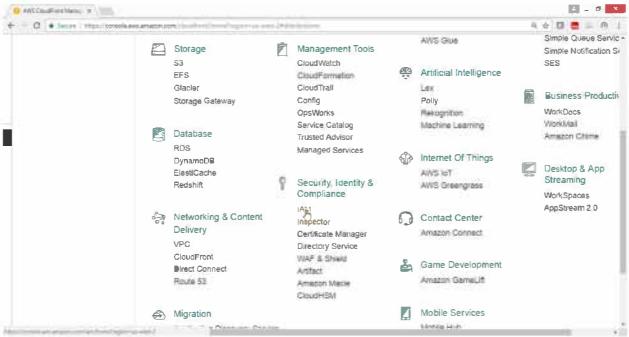
Create your own policies.

Users Login to sign-in page.

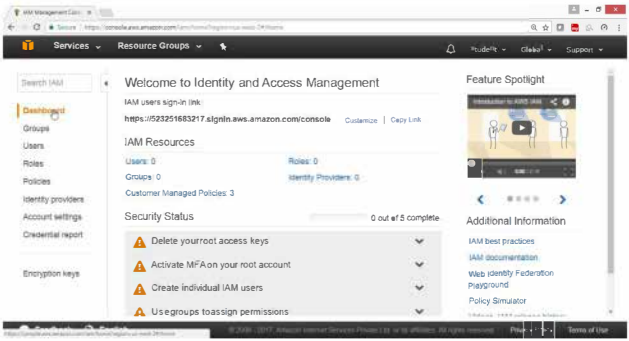
Deleting users and groups.

1) To create user, assign password, change password policy.
Open AWS console select **Security, Identity & Compliance**

Click on **IAM** service



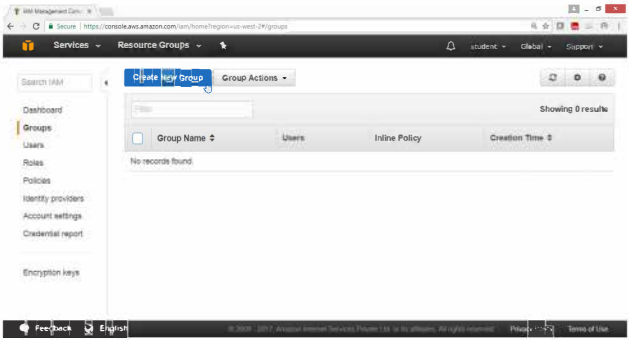
IAM Dashboard panel available



2) To Mange Groups and applying policies

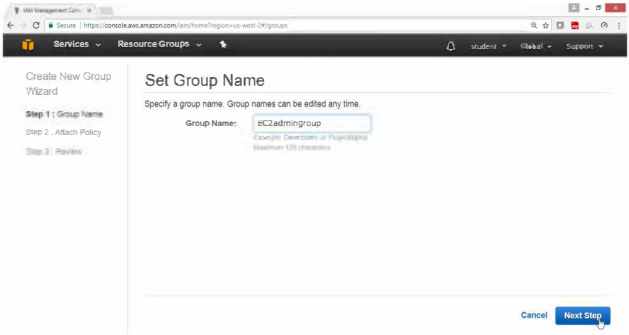
From IAM dashboard, select **Groups**

Click on **Create New Group** button



Give Group Name → EC2admingroup

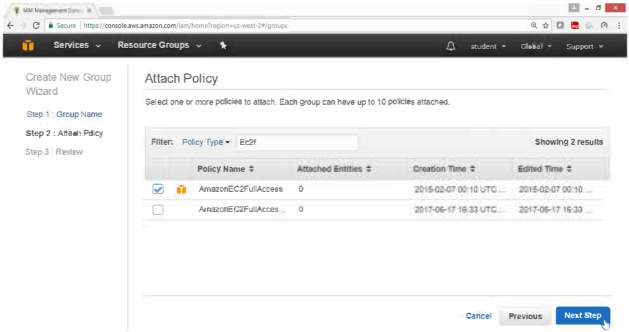
Click on **Next Step** button



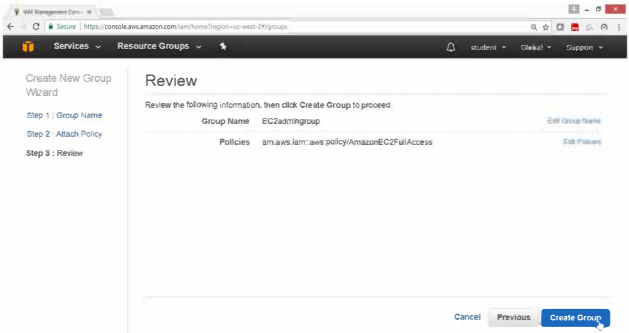
In Filter type → EC2f

Select check box for **AmazonEC2FullAccess**

Click on **Next Step** button

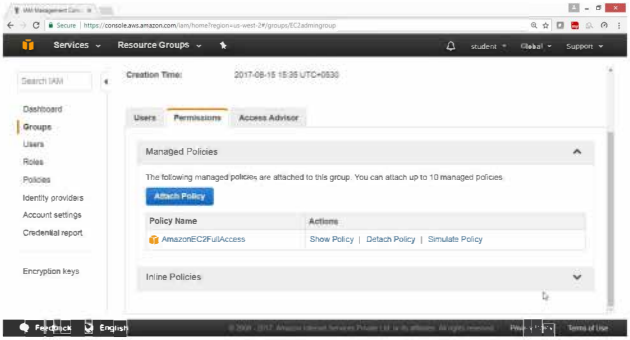


Click on **Create Group**



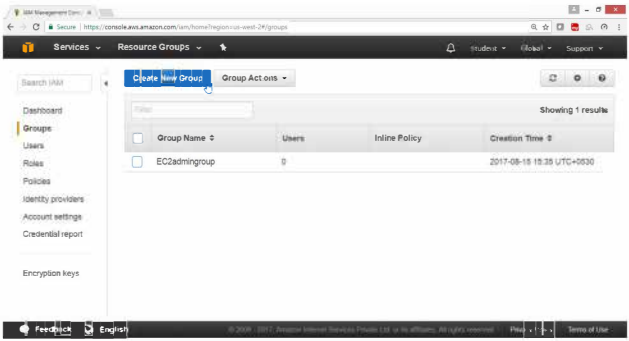
Verify

Group EC2admingrp got created with AmazonEC2FullAccess policy

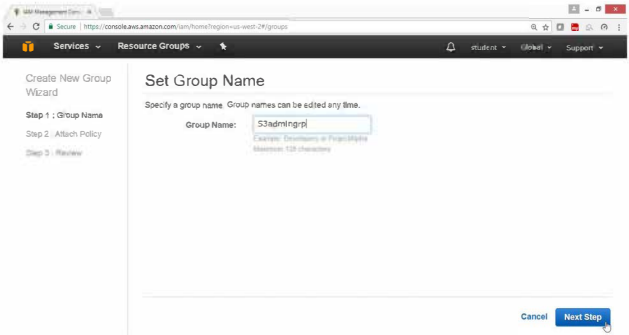


Now again create Another Group

Click on **Create Group** button



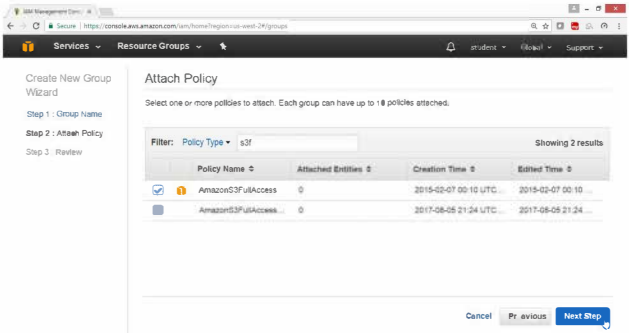
To create a group With S3FullAccess



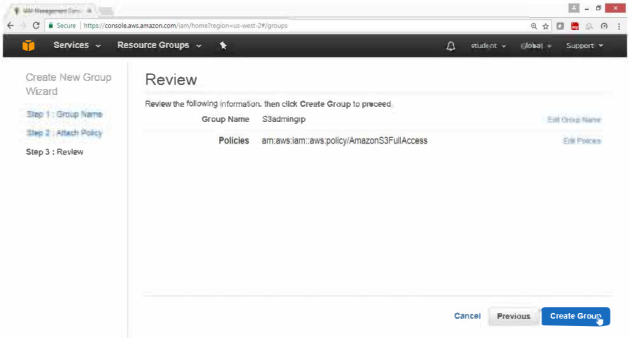
In Filter type → S3f

Select check box for **AmazonS3FullAcess**

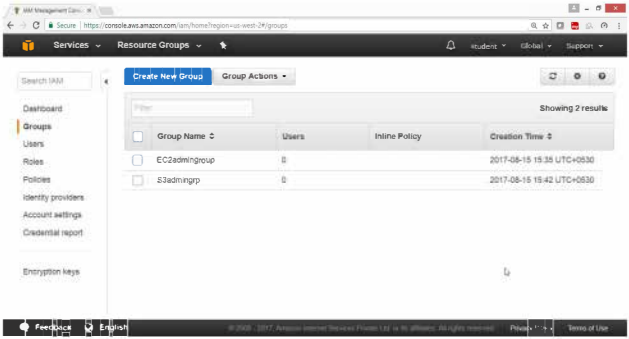
Click on **Next Step** button



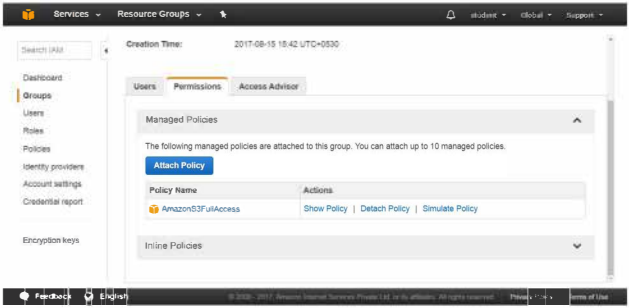
Click on **Create Group** button



Verify EC2admingroup & S3admingr groups got created



Verify S3 policy is attached



Create user tom and join to EC2admingroup

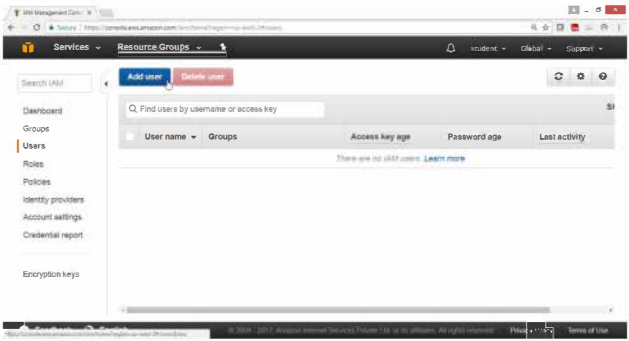
Create user john and join to S3admingroup

Create a user sai add Ec2fullaccess and S3fullacces Policy

From IAM dashboard

Select **Users**

Click on **ADD Users** button



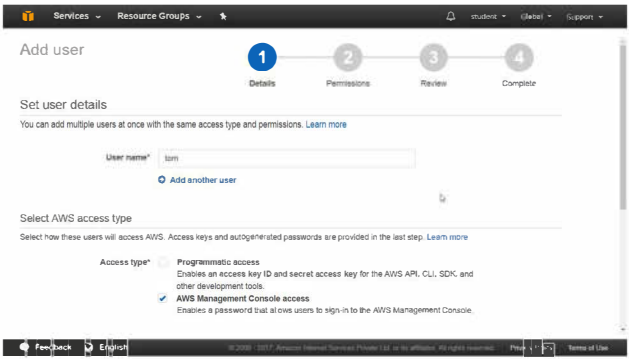
Scenario 1)

Create user tom and join to EC2admingroup

For User name → tom

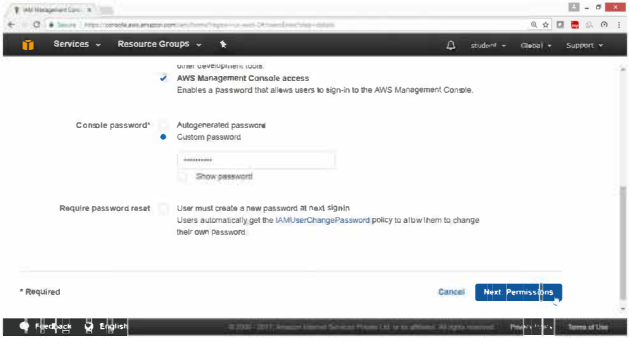
For Access type → AWS Management Console access

Drag down



For Console password → ******

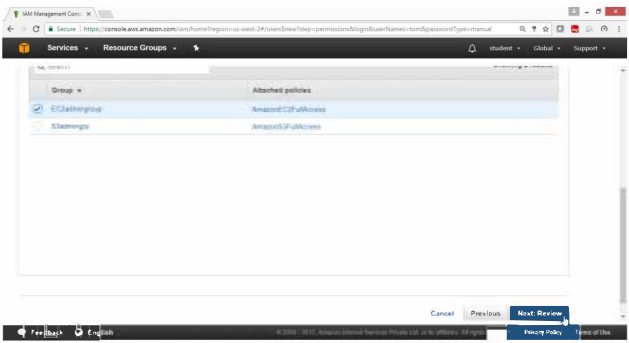
Click on Next **Permissions** button



Under Group column

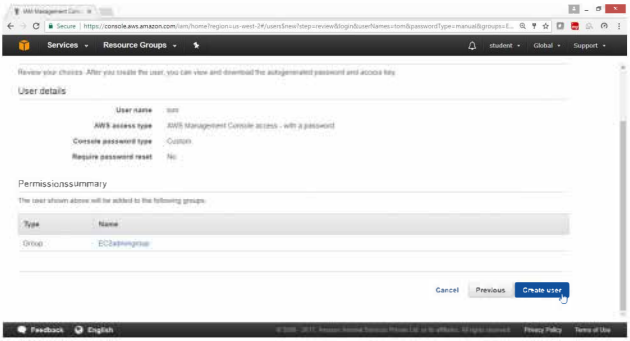
Select EC2admingroup

Click on **Next Review**

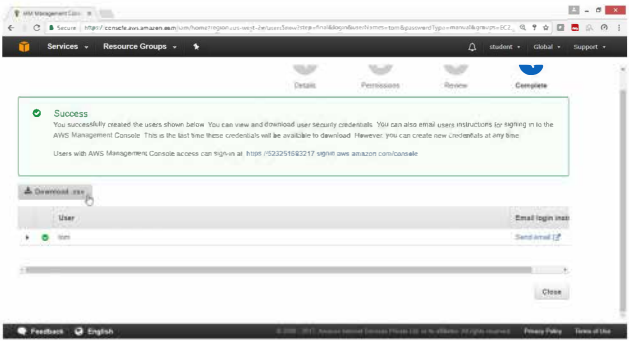


Verify users detail

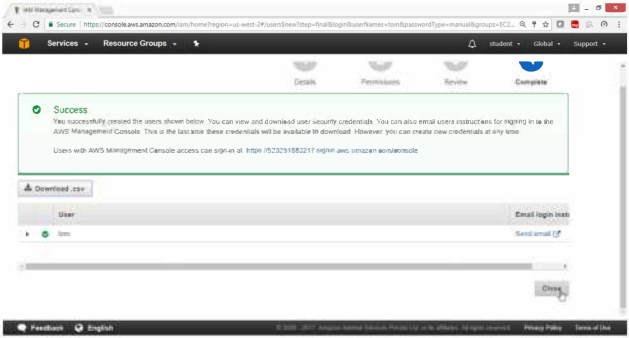
Click on **Create user** button



Down the .csv file



Click on **close** button

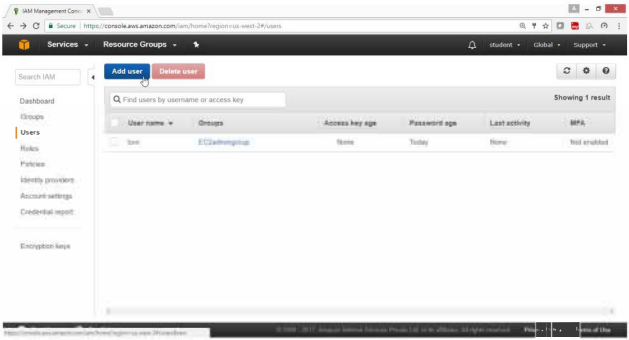


Scenario 2)

Create user john and join to S3admingroup

Select user

Click on **Add user** button

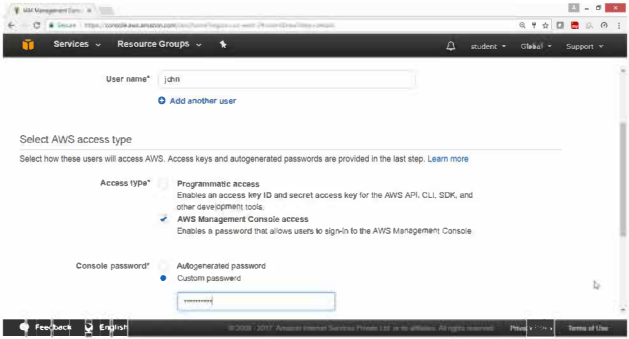


For user name → john

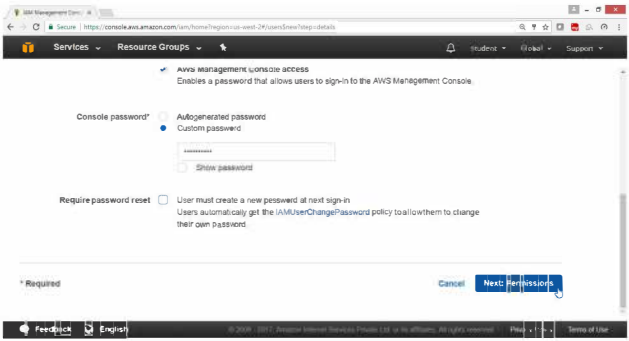
For Access type → AWS Management Console access

For console password → ******

Drag down

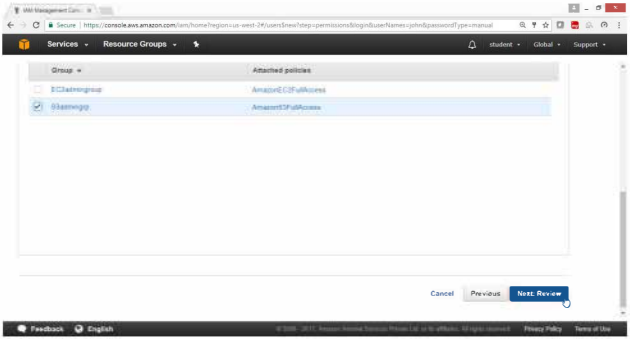


Click on **Next Permission** button



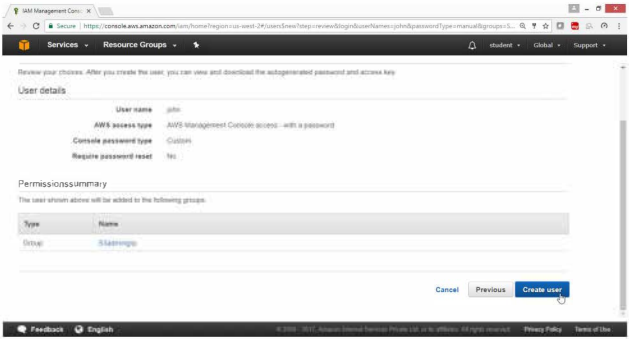
Select S3admingrp

Click on **Next Review** button



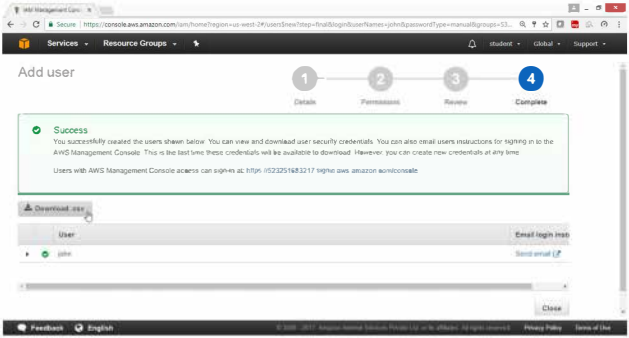
Verify user details

Click on **Create user** button



Download .csv file

Click on **Close** button



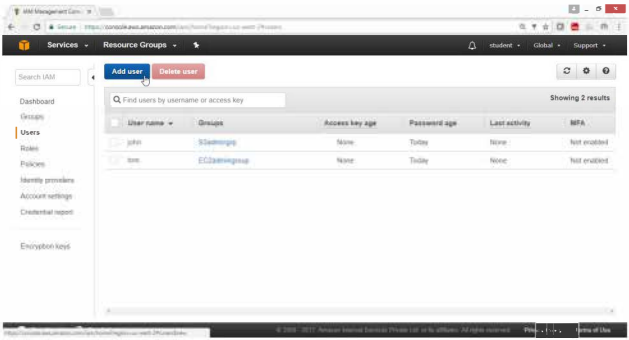
Scenario 3)

Add a user individual user sai without joining to any group

Attach EC2FullAccess and S3FullAccess policy

Select User

Click on **Add user** button

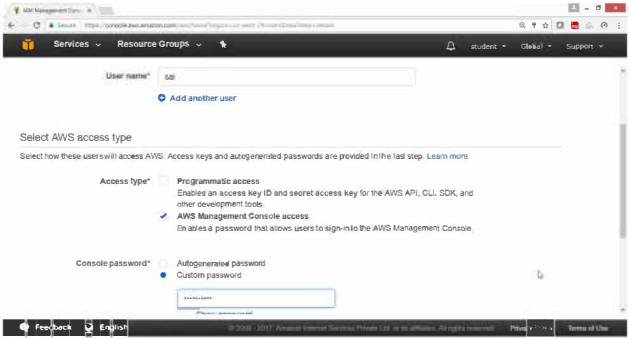


For User name → sai

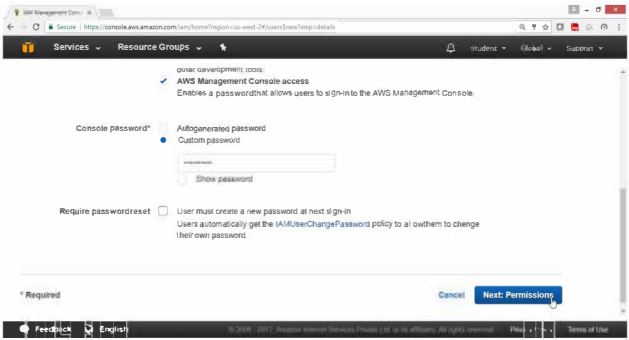
For Access type → AWS Management Console access

For Console password → ******

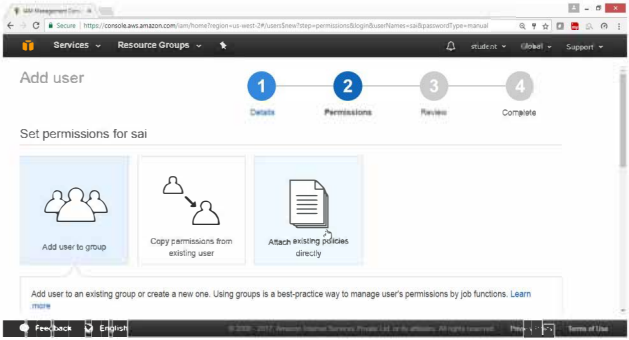
Drag Down



Click on **Next permission** button

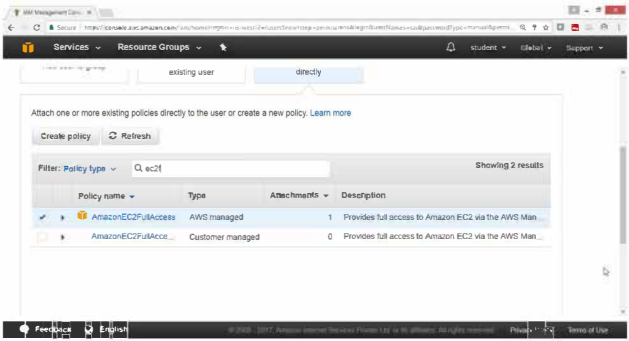


Click on **Attach existing policies directly** box



In Filter type search for ec2f

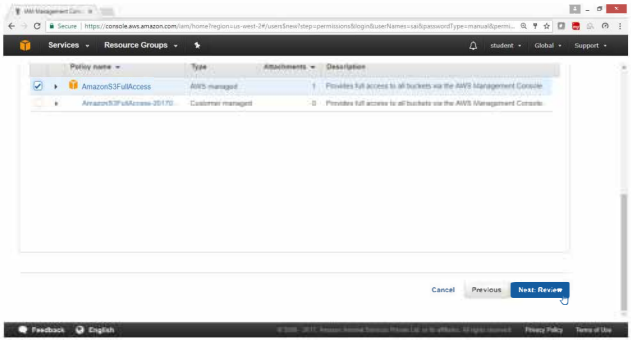
Select AmazonEC2FullAccess check box



In Filter type search for s3f

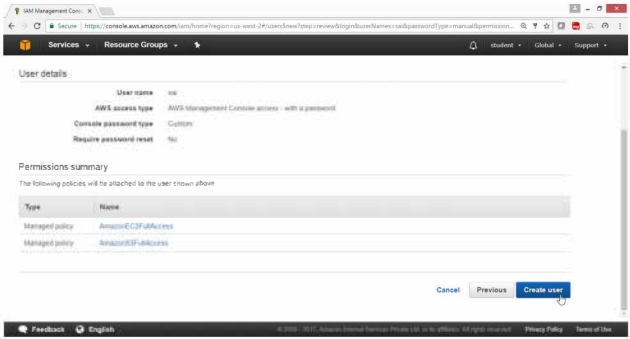
Select AmazomS3FullAccess check box

Click on **Next Review** button



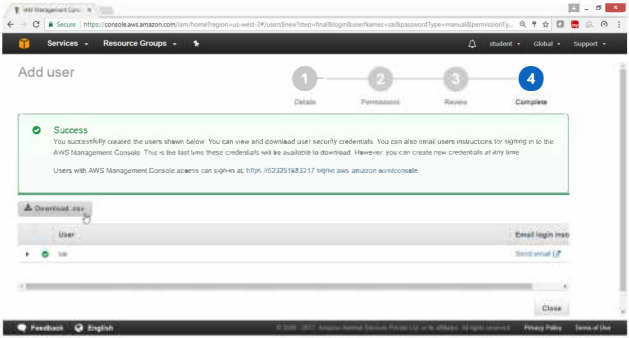
Verify users detail

Click on Create user button



Download .csv file

Click on **Close** button



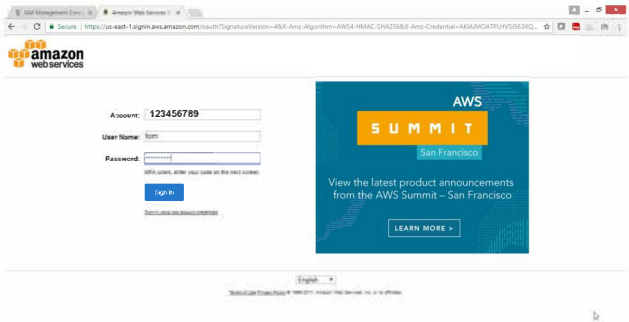
To verify wheather users can access particular Service

Login as tom user

Provide the following url in Browser

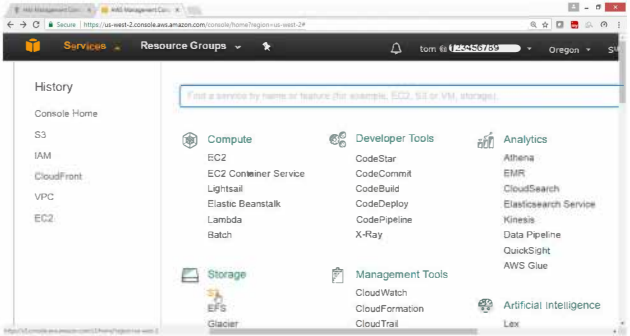
https://123456789.signin.aws.amazon.com/console

Click on Sign in button



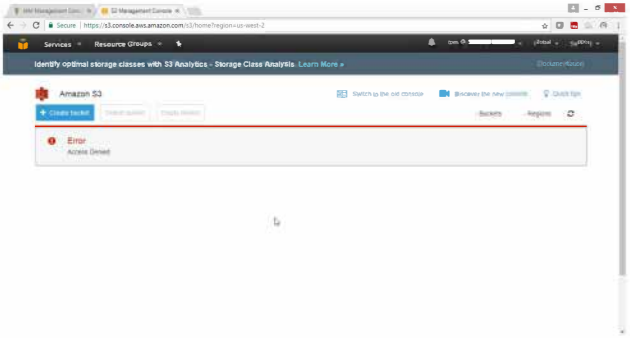
User tom is not having S3 access

Click on S3 verify the access

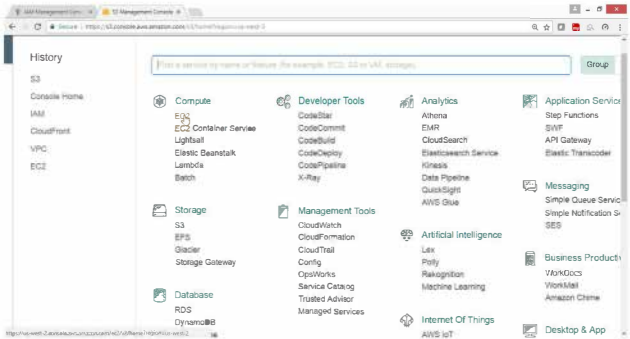


Verification

Error Access Denied

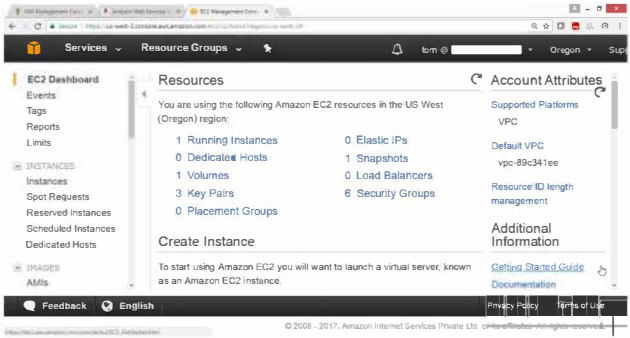


Now select EC2 service



Verification

User tom can access EC2 service.



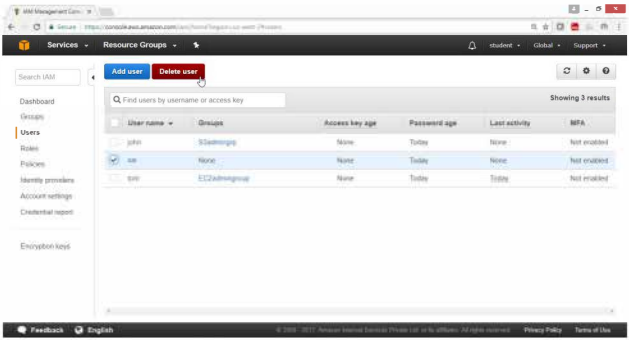
Similarly check for user john

To Delete users and groups

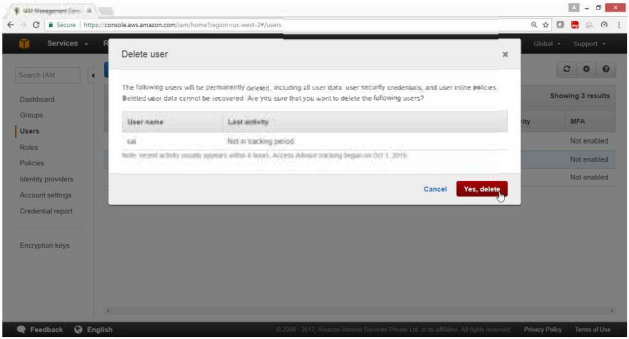
From IAM dashboard, select **Users**

Select the users, drop down **Action** button

Click on **Delete Users** button

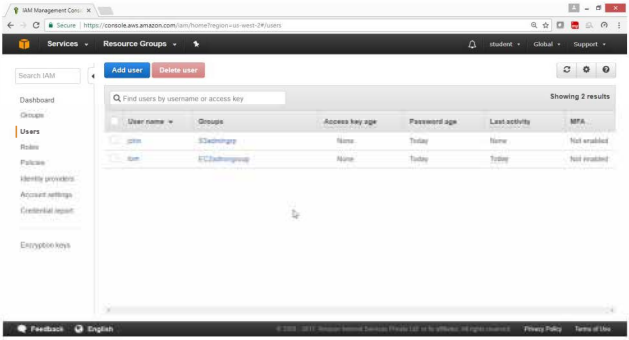


Click on **Yes, delete** button



Verification

User sai is deleted



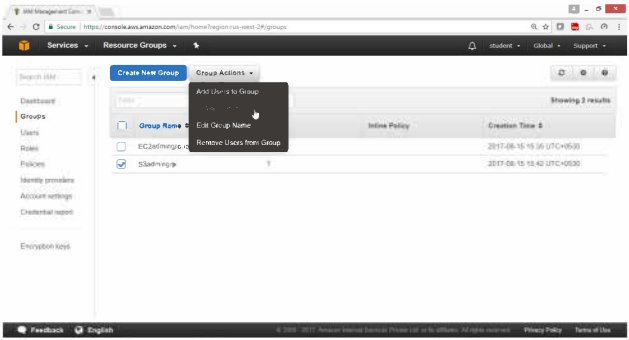
To Deleting Groups

From IAM Dashbaoard

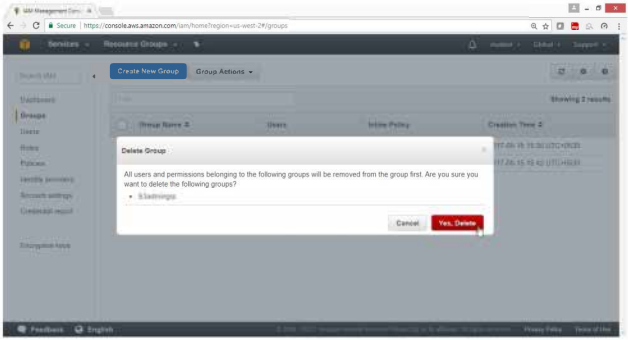
Select the **Groups**

Drop down **Group Action** button

Select **Delete Group**

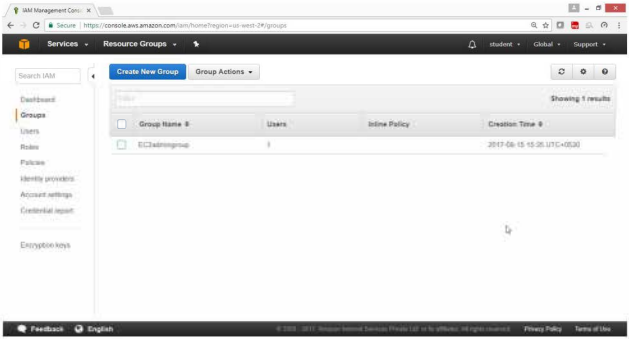


Click **Yes, Delete** button



Verification

Group is deleted



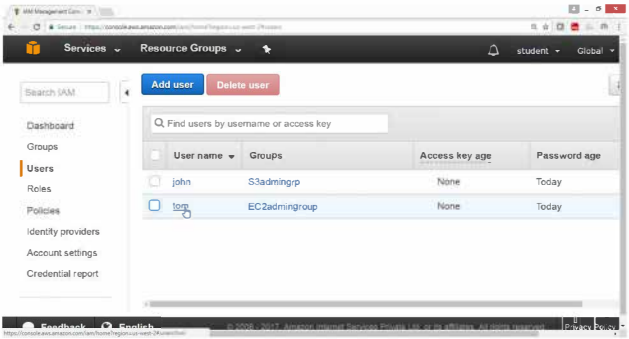
To Create Multifactor Authentication

Install Google authenticator in your Android Mobile

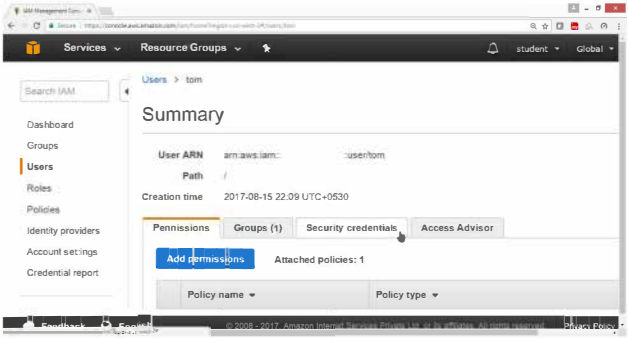
On the **IAM Dashboard** panel

Click on Users

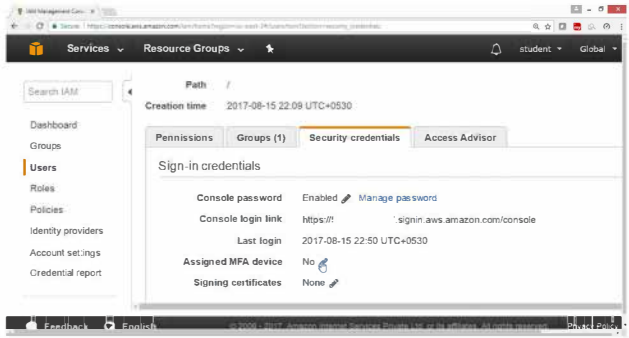
Click on the user tom



Click on Security credentials

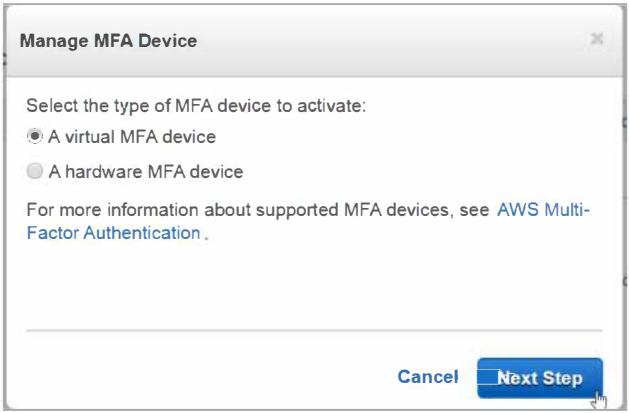


Click on pen sign for **"Assigned MFS device"**

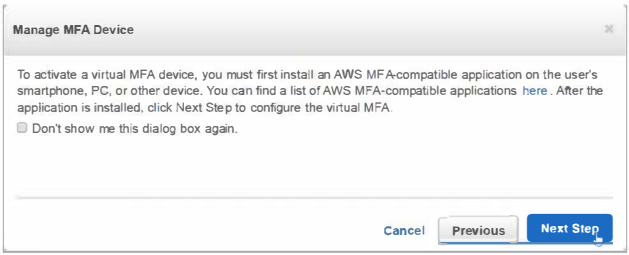


Select →**"A virtual MFS device"**

Click on **Next Step** button



Click on **Next Step** button



Bar code will be created

Scan this bar code from your mobile Google Authenticator application.

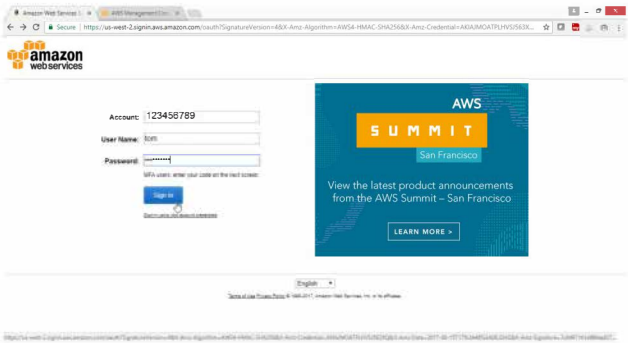
Now type 6 digit bar code in Authetication code 1

Once the bar code changes

Retype 6 digit bar code in Authetication code 2

Click on Finish

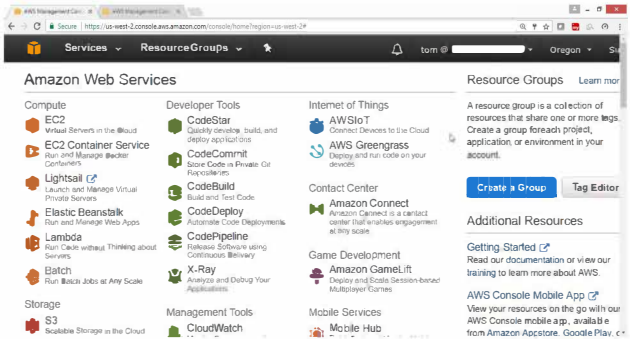
Now login as tom user



Once the user types the MFA 6 digit coder

Click on submit

Verify user had successfully logged in.

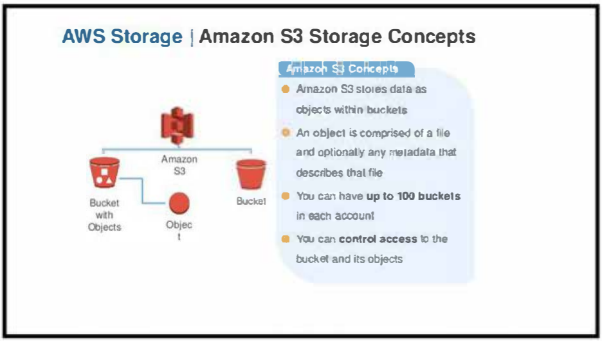


Lab 7: To Configure Amazon Simple Storage Service (Amazon S3)

OBJECTIVE

To configure and use AWS S3 service

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with AmazonS3FullAccess

To Configure S3 with following task:

Sign Up for Amazon S3

Create a Bucket

Add an Object to a Bucket

Add an folder to Bucket

View an Object

Move an Object

Delete an Object and Bucket

To empty a bucket

To delete a bucket

Hosting a Static Website on Amazon S3

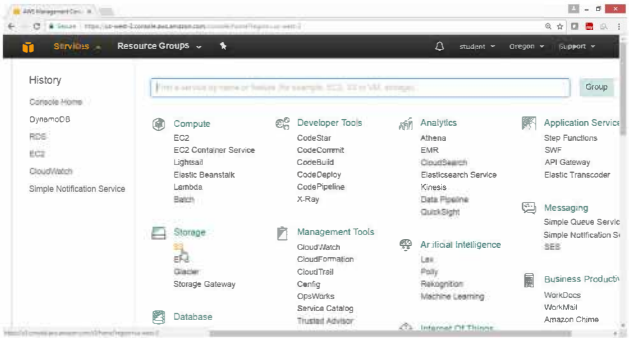
AWS user to control S3

1. To create S3 bucket for storing objects that is files and folders

Open AWS console

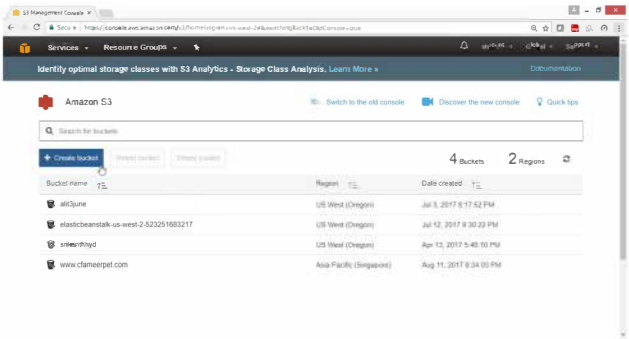
Select **"Storage"** service

Click on **S3**



On Amazon **S3** page

Click on **Create Bucket**



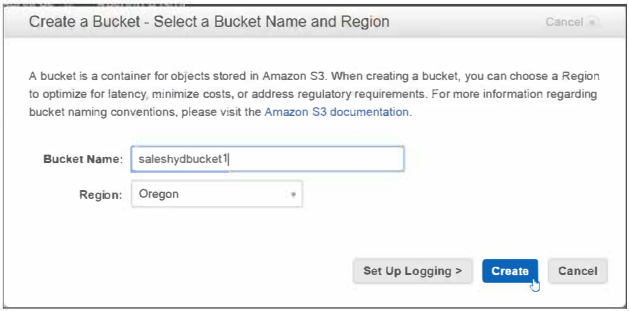
On **"Create Bucket - Select a Bucket Name and Region" box**

Provide following values

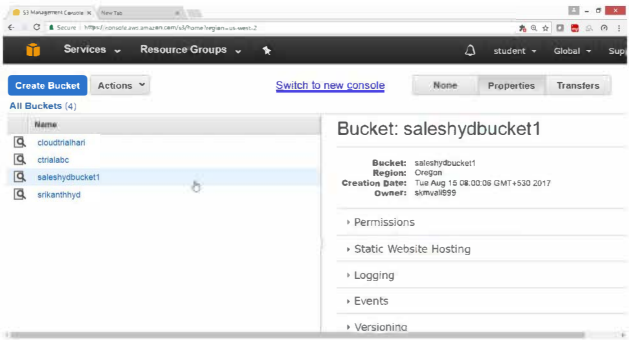
Bucket Name → saleshydbucket

Region → Oregon

Note: A bucket name in region must contain only lower case characters and should be unique in entire Amazon bucket names from all the region.



Verify that bucket is created.



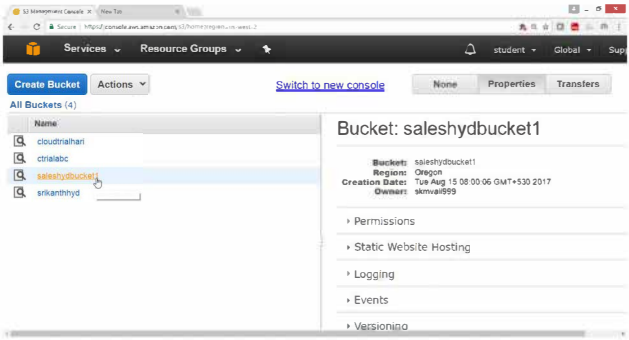
To upload files of any type

Right click in empty space, select **Upload**

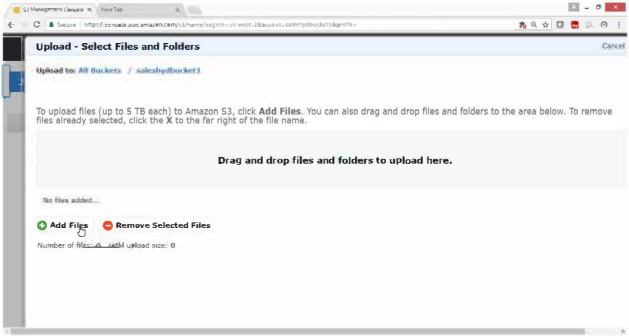
Note: 5 GB can be uploaded

It will be charged if crossed free tier usage.

Click on Created bucket



Click on **Add files**

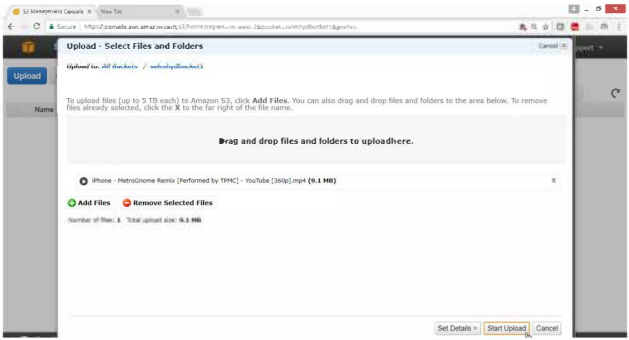


In the upload Wizard

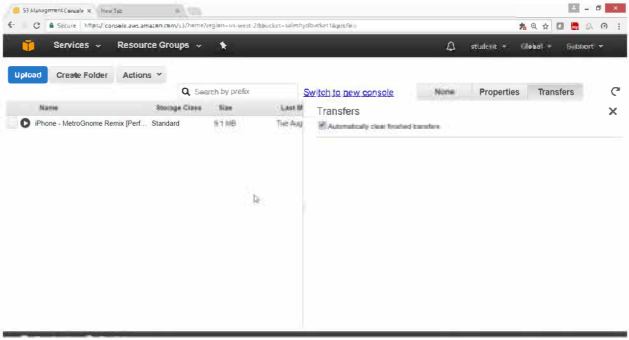
Click on **Add files**

Select some txt, pdf, video files

Click "**start upload**" button

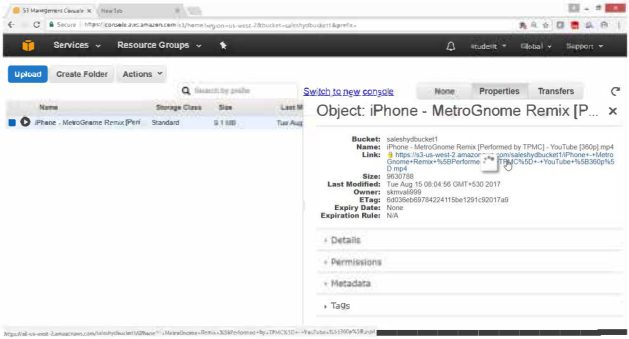


Verify that the file got uploaded.



Select the file, Click on Properties on Right Panel,

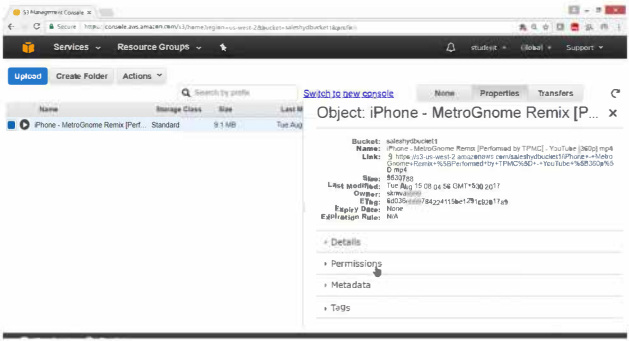
Click on the **link**



Verification : Cannot access due to lack of permission

To allow users to Download, or view give permission

Select, **Permission** tag



Click on **Plus Radio** button for **Add more permissions**

Drop down Grantee Button

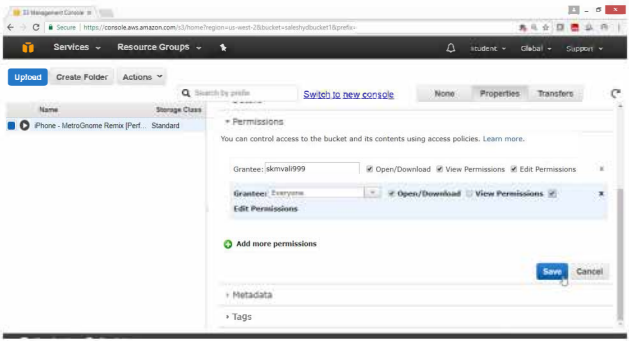
Select **Everyone** to make it public

Enable the check box to **Open/Download**

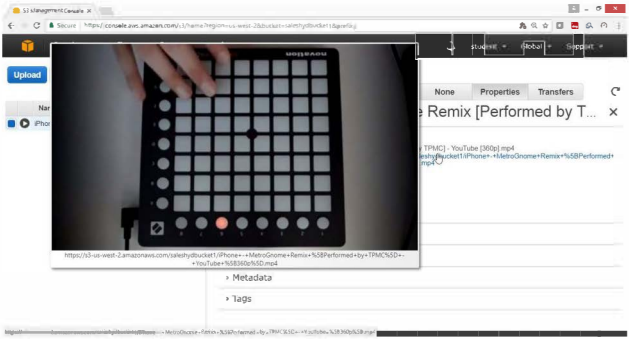
Enable the check box to **View Permission**

Enable the check box the **Edit View Permission**

Click on **Save** button



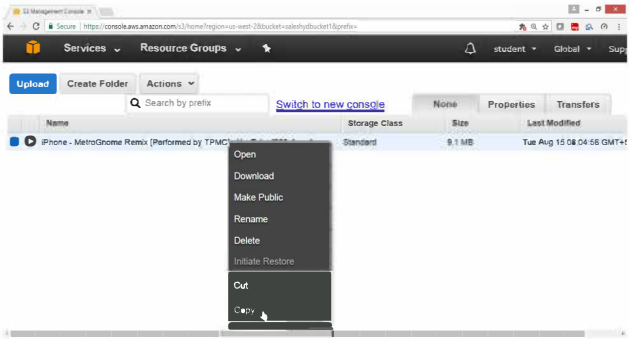
Verify file is accessible.



2) To copy or move files from one bucket to another.

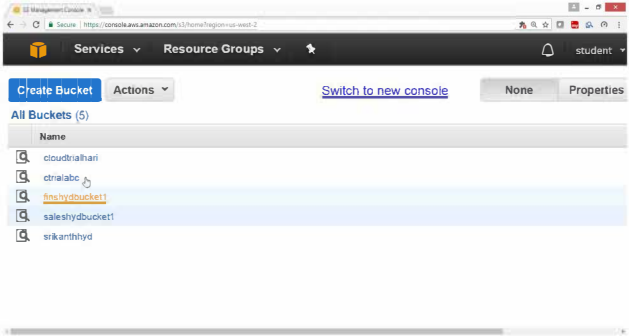
Select the file from Bucket or Folder, right click,

now select copy/cut

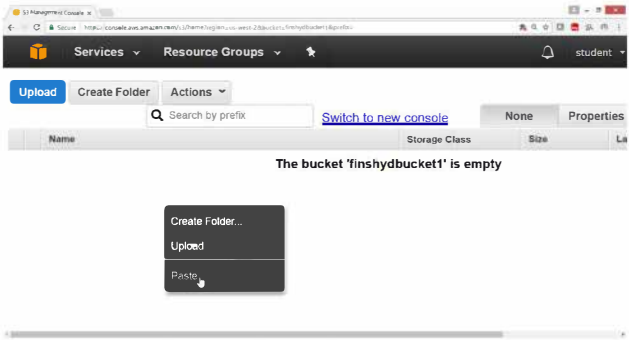


2.2 Select the Bucket or Folder, where you want to paste.

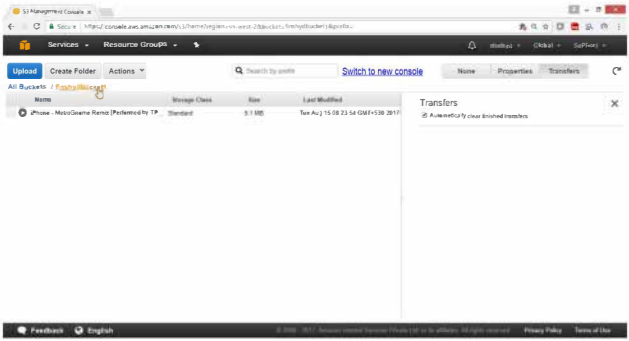
Click on the Bucket → finshydbucket1



Click on **Paste**

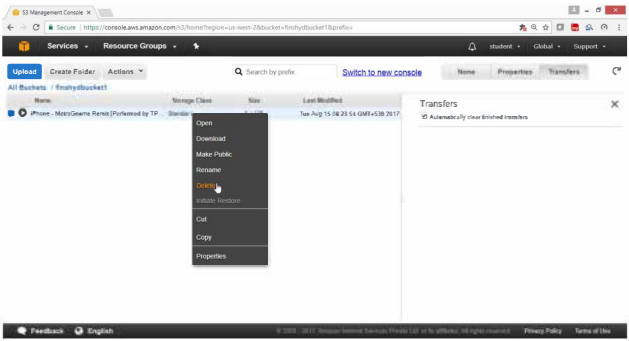


Verify that the file is copied in another bucket i.e finshydbucket1



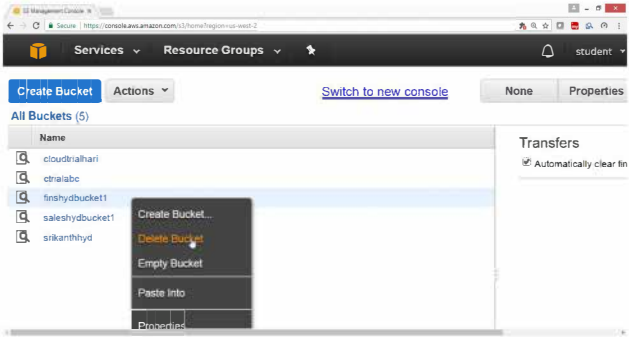
3) To **delete** a file from a bucket

Right click on it, select Delete



To Delete a bucket

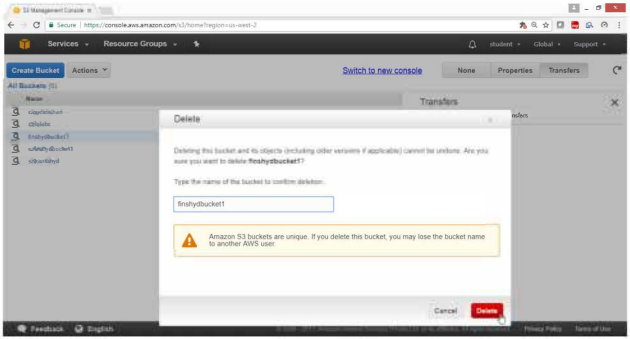
Select the bucket, right click select **Delete Bucket**



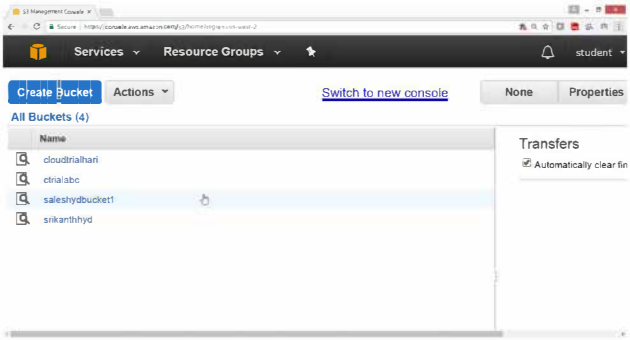
To **Delete Bucket**

Provide exact bucket name

Click on **Delete** button



Verify that the bucket **finshydbucket1** is deleted



4) To Host a Static Website using Amazon s3 Bucket

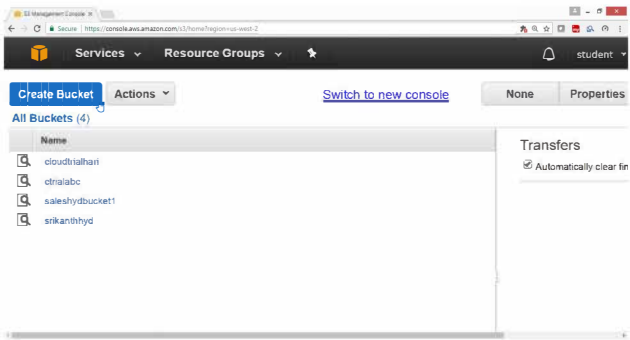
To Host a Static Website using Amazon s3 Bucket

Open AWS console

Select **Storage**

Click on **S3** service

Click on "**Create Bucket"**

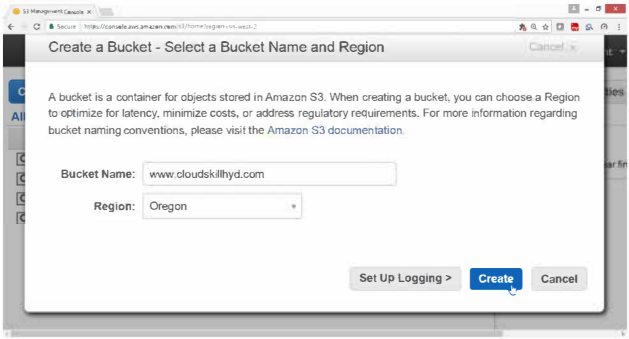


On "**Create a Bucket - Select a Bucket Name and Region**" page

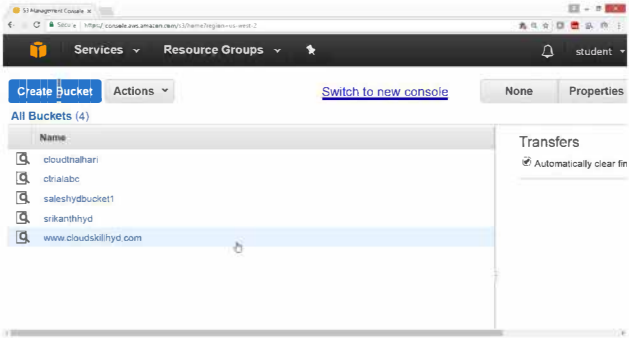
Provide following values for

Bucket Name	→	www.cloudskillhyd.com
Region	→	Oregon

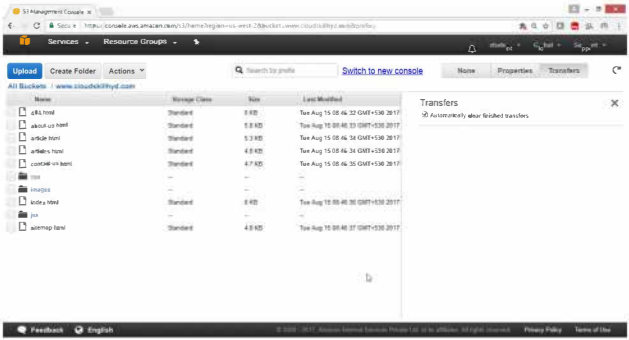
Click on **Create** button



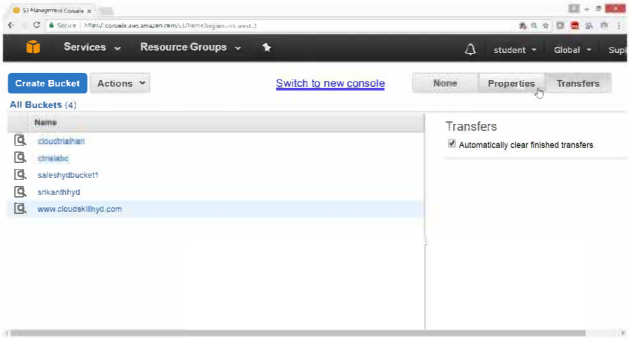
Verify Bucket got created



Upload all website contents in this bucket.



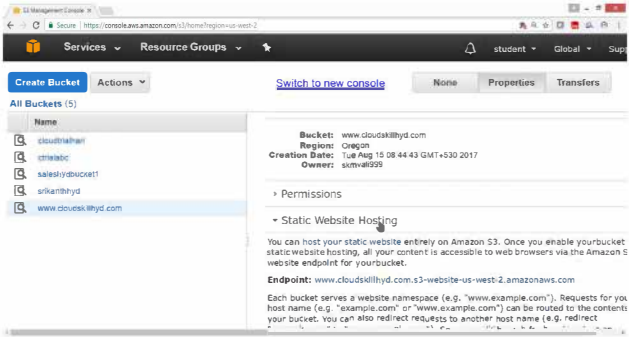
Select the bucket and click on **properties** button



On the **Properties** panel

Click **Static Website Hosting**

Drag Down



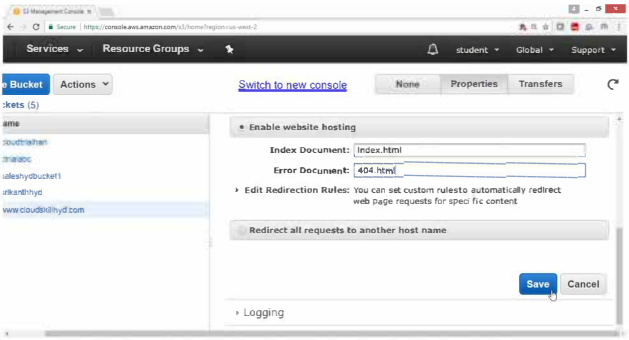
Select the **Enable website hosting**

Provide following values for

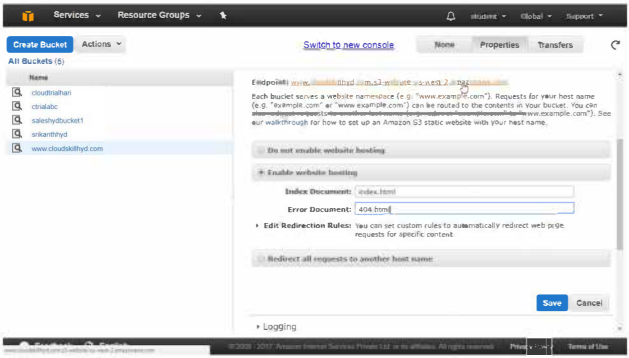
Index Document box → index.html

Error Document box → 404.html

Click on **Save** button



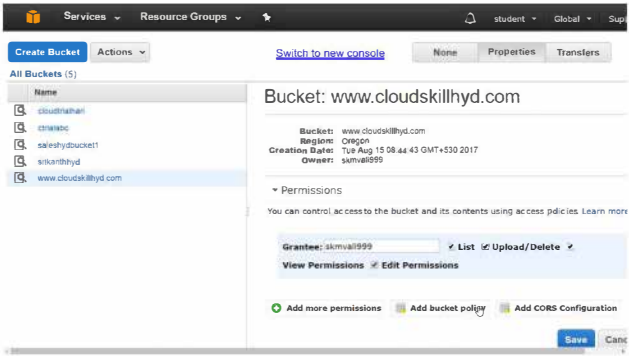
Note down the Endpoint.



2. To add a bucket policy that makes your bucket content publicly available

In the Bucket Properties, click on **Permission**

Click on **Add Bucket Policy.**



Copy the following bucket policy, and then paste it in the Bucket Policy Editor.

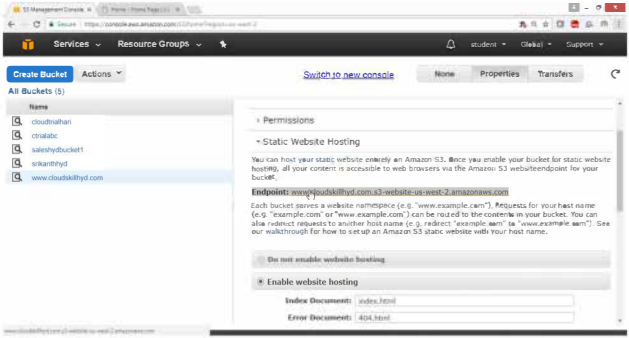
```
{
    "Version":"2012-10-17",
    "Statement":[{
          "Sid":"PublicReadForGetBucketObjects",
        "Effect":"Allow",
            "Principal": "*",
       "Action":["s3:GetObject"],
       "Resource":["arn:aws:s3:::cloudskillhyd.com/*"
      ]
     }
    ]
  }
```

Click on **Save** button

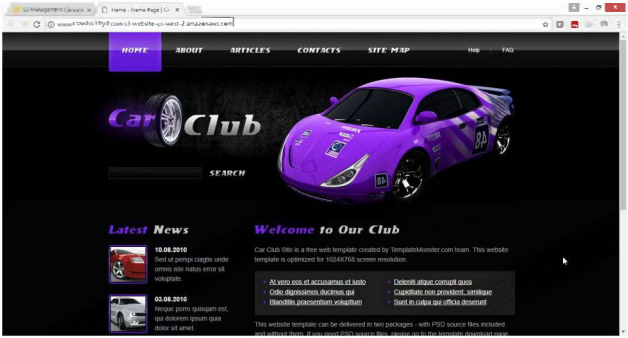
Verify your website

Click on Endpoint Under **Static Website Hosting**

Endpoint: www.cloudskillhyd.com.s3-website-us-west-2.amazonaws.com



Verify the website which is coming from S3 Bucket

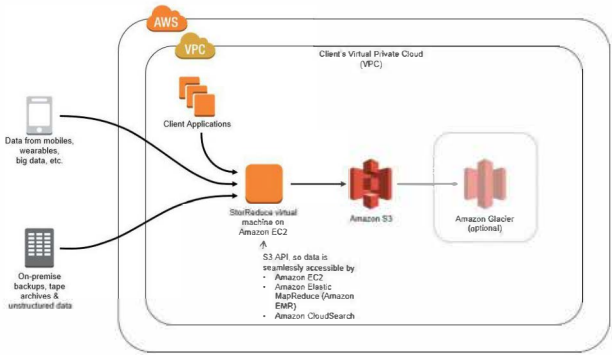


Lab 8: To configure Amazon Glacier

OBJECTIVE

To configure and use AWS Glacier Service.

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with AmazonGlacierFullAccess policy.

To configure Glacier with following task.

Transfer files from S3 to Glacier

Note: Amazon does not allows files to be directly loaded on Glacier

use s3 or third party tools to archive or restore.

1.Using s3 bucket & s3 lifecycle permission to archive in glacier

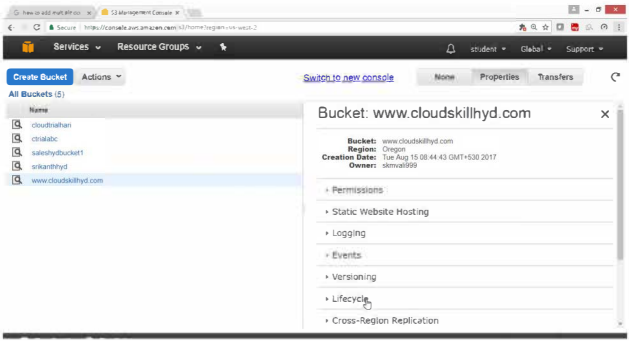
Select S3 bucket

[refer s3 topics how to create bucket and upload files]

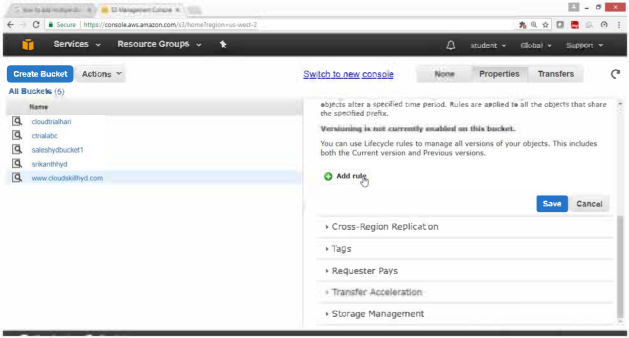
Select the bucket,

Go to properties

Click on **Lifecycle**



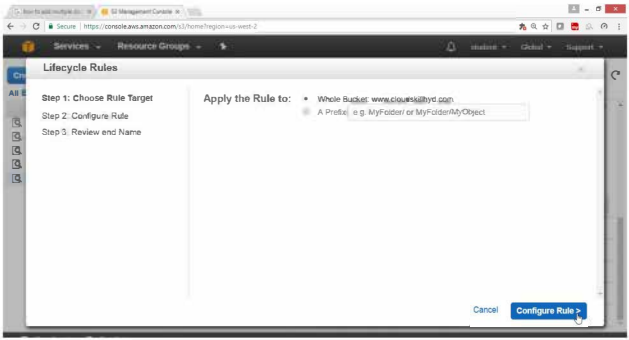
Click on **Add rule**



Under Lifecycle Rules

select **Choose Rule Target**

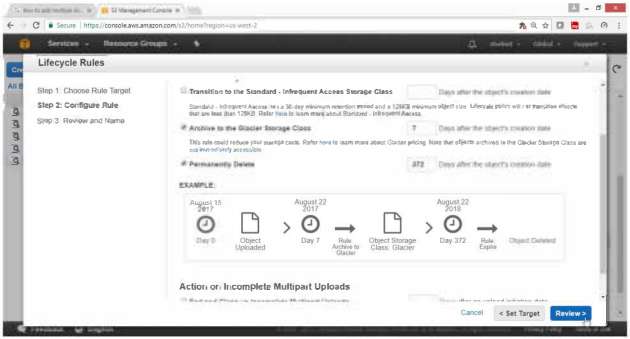
Apply the Rule to → Whole Bucket



Select check box **Archieve to the Glacier Storage Class→** 7

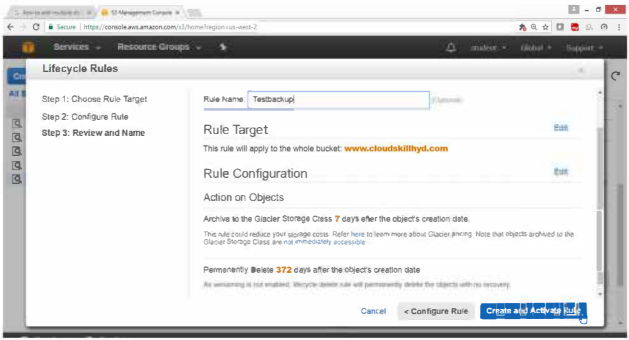
Select the check box **Permanently Delete → 372**

click on **Review**

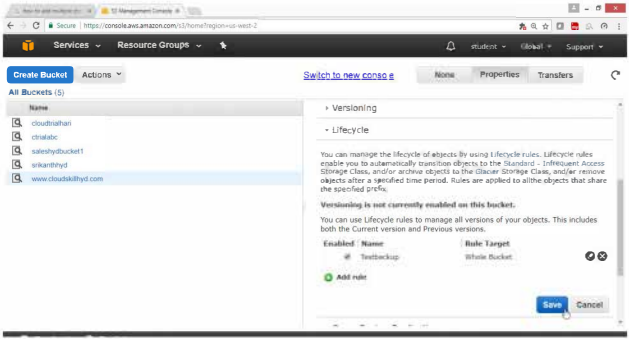


Provide Rule Name → Testbackup

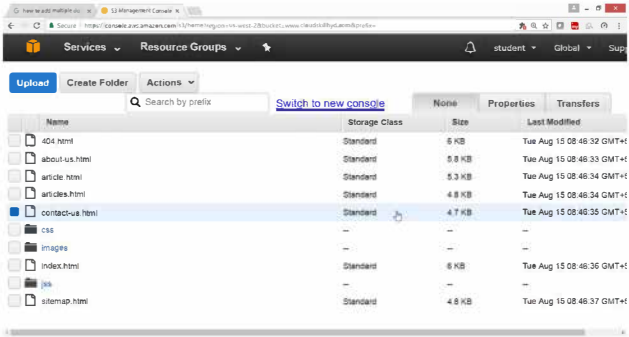
click on **"Create and Activate Rule"** button



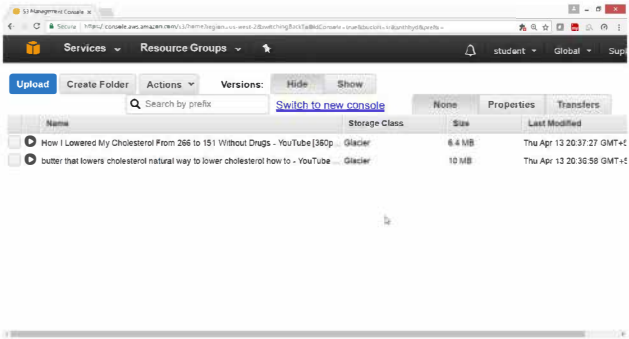
Click on **Save** button



Verify Storage Class is **Standard**

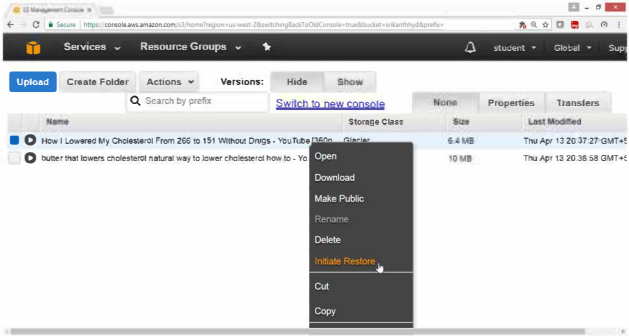


Verify Once the file goes to Glacier then Storage Class is **Glacier**



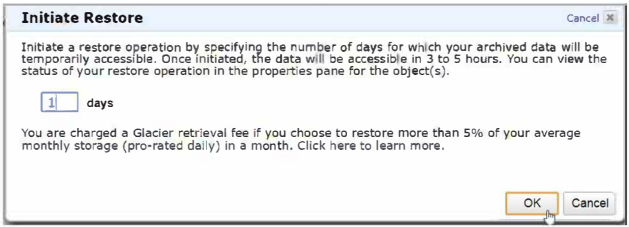
To Restore go to the bucket select the file

Right click and select **Initiate Restore**



Provide number of days → 1

Click on **OK**



Verify

File will get restored after 1 Day

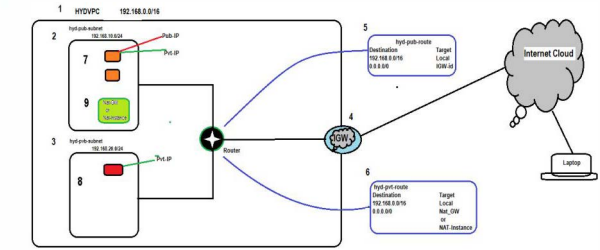
Storage class will become Standard.

Lab 9: To Configure Amazon Virtual Private Cloud (VPC)

OBJECTIVE

To configure Amazon Virtual Private Cloud with public and private subnet

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with VPCfullaccess

TASK

Create your own VPC

Create Public subnet

Create Private subnet

Create Internet Gateway

Attach Internet Gateway to your VPC

Create Public Routing Table, associate subnet and add routing rules

Create Private Routing table, associate subnet and add routing rules

Launch an instance in Public network

Launch an instance in Private network

Create Nat Gateway

Connect to public instance and check internet connectivity

Connect to private instance and check internet connectivity

Amazon Virtual Private Cloud (Amazon VPC) enables you to launch Amazon Web Services (AWS) resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

1) To create your own VPC

Open **AWS** console

Click on **Services**

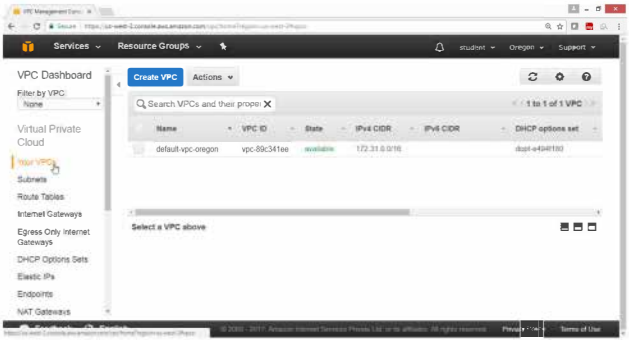
Select **Networking and Content Delivery**

Click on **VPC**

On **VPC Dashboard** panel

Click on **Your VPC**

Click on **Create VPC** button



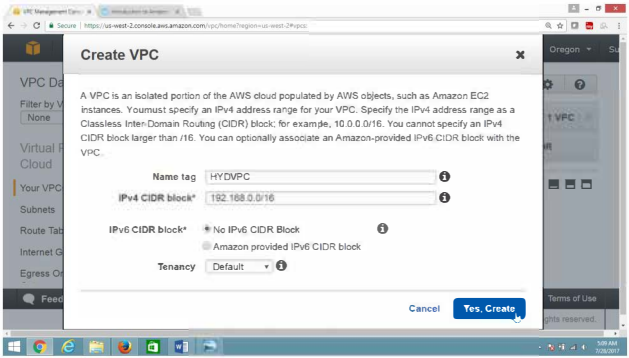
On **"Create VPC"**, page

For **Name tag→HYDVPC**

For **IPv4 CIDR block→192.168.0.0/16**

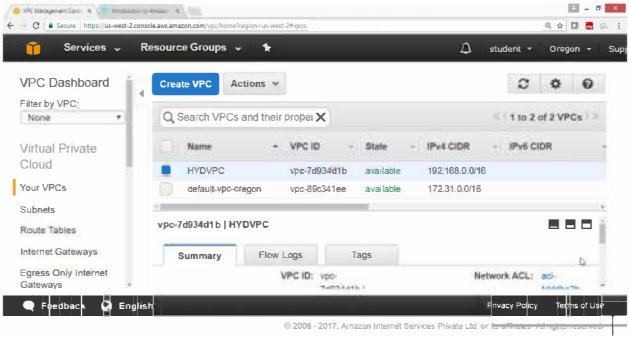
Leave remaining field as default

Click on **"Yes Create"** button



Verify

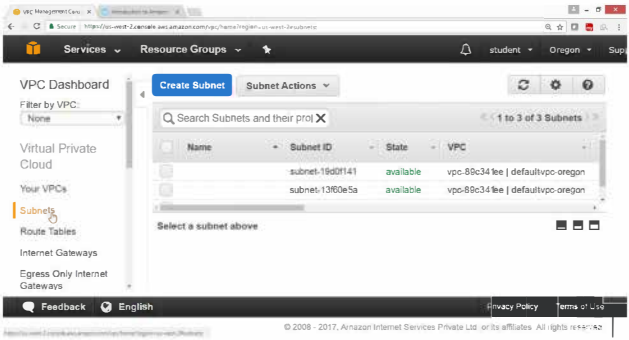
HYDVPC is created



2) To create public subnet

Click on **Subnet**

Click on **Create Subnet** button



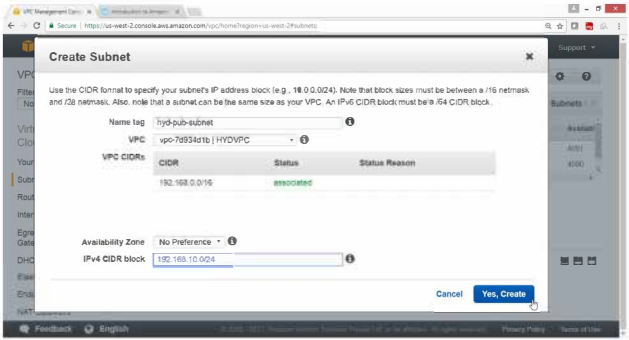
On **Create Subnet**, page

For **Name tag** → **hyd-pub-subnet**

For **VPC** → **HYDVPC**

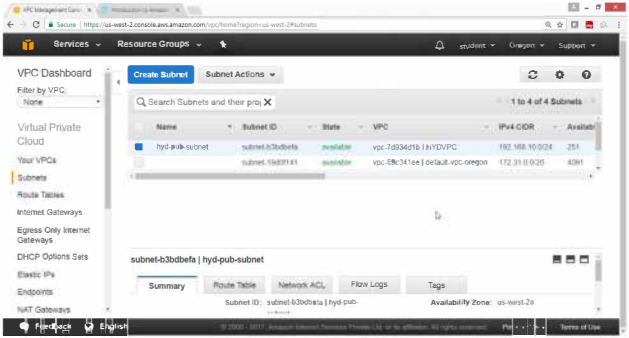
For **IPv4 CIDR block** → **192.168.10.0/24**

Click on **Yes Create** button



Verify

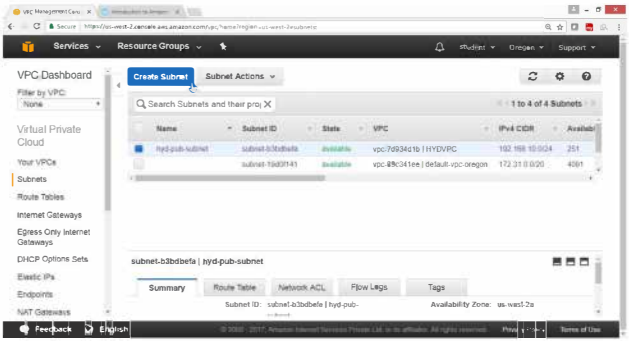
hyd-pub-subnet got created



3) To create private subnet

Click on **Subnet**

Click on **Create Subnet** button



On **Create Subnet**, page

For **Name tag** → **hyd-pvt-subnet**

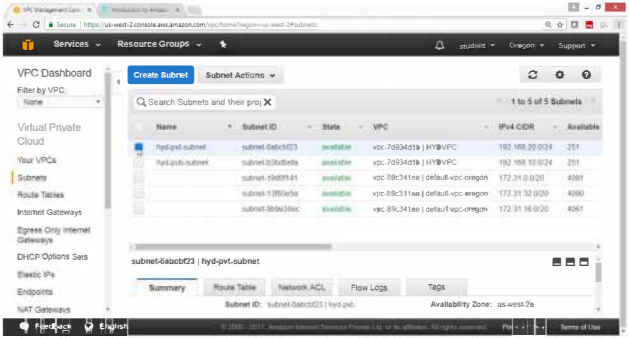
For **VPC** → **HYDVPC**

For **IPv4 CIDR block** → **192.168.20.0/24**

Click on **Yes Create** button

Verify

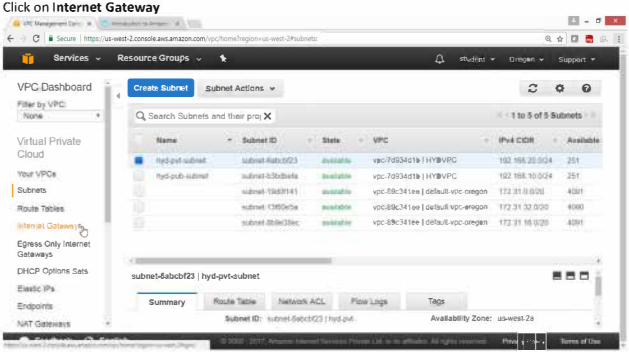
hyd-pvt-subnet got created



4) Create a Internet Gateway and attach to your VPC.

In VPC **Dashboard** panel

Click on **Internet Gateway**

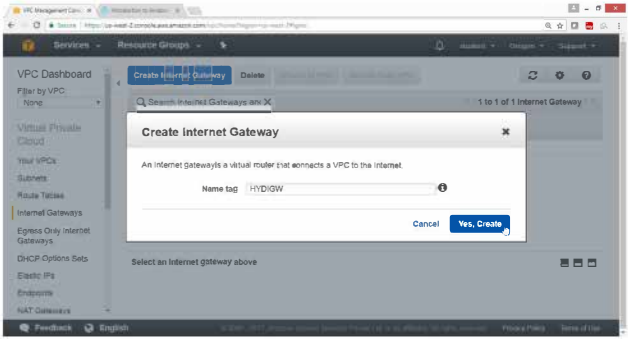


Click on **Create Internet Gateway** button

In **Create Internet Gateway**, box

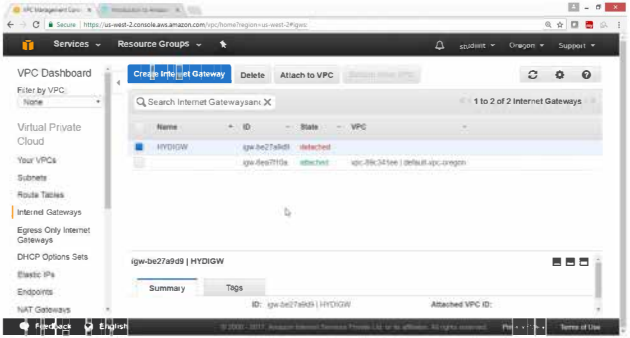
For **Name tag → HYDIGW**

Click on **"Yes, Create"** button



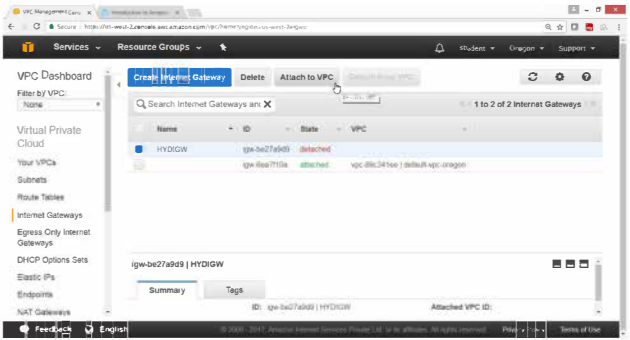
Verify

Internet gateway is created



Select HYDIGW

Click **"Attach to VPC"**



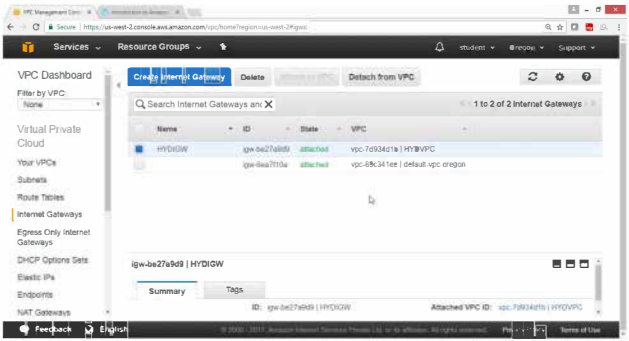
In **"Attach to VPC"** box

For VPC →**HYDVPC**

click on **"Yes, Attach"** button

Verify

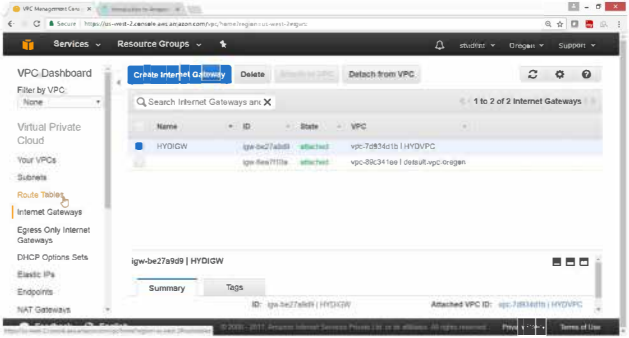
Internet gateway is connected to your VPC



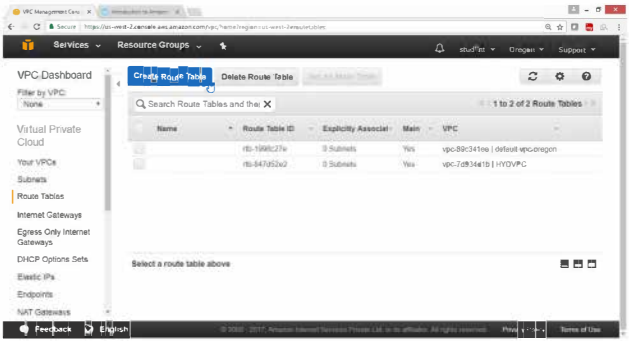
5) Create Public Routing Table, associate subnet and add routing rules

On **VPC Dashboard** panel

Click on **Route Table**



Click on **"Create Route table"** button

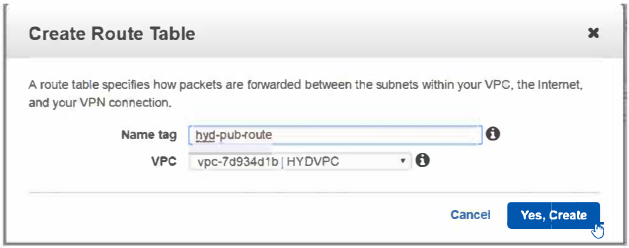


On **"Create Route Table"** box

For **Name tag** → hyd-pub-route

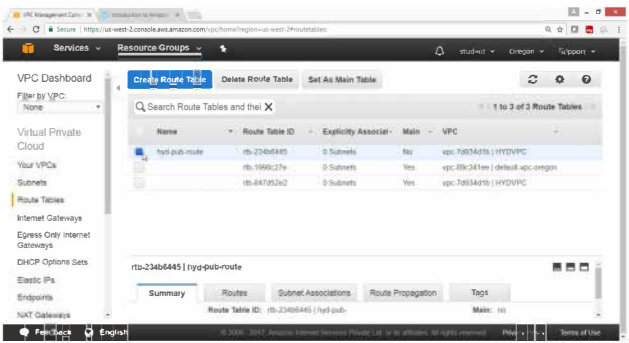
For VPC → HYDVPC

Click on **"Yes, Create"** button

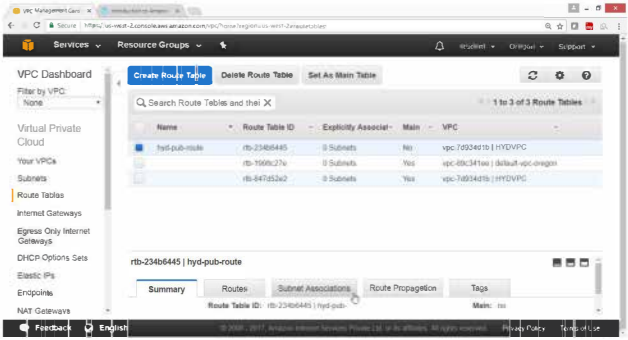


Verify

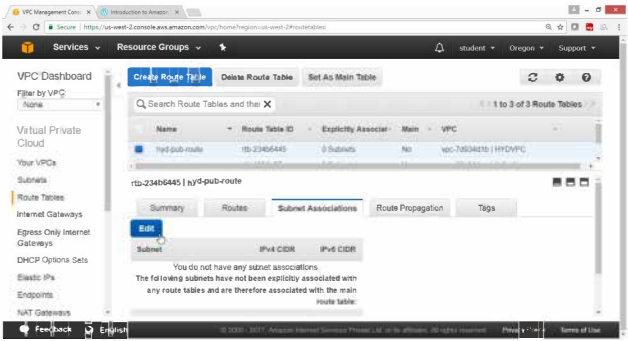
hyd-pub-route table is created



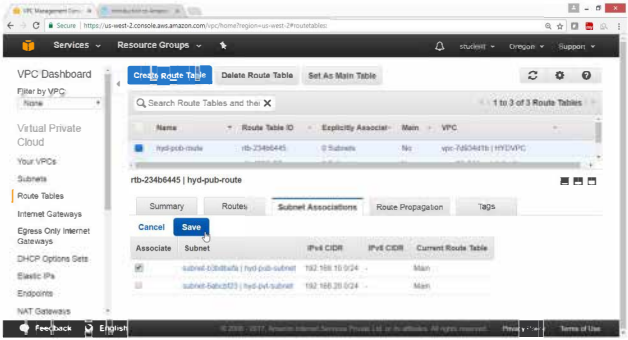
Click on **"Subnet Association"** button



Click on **Edit** button

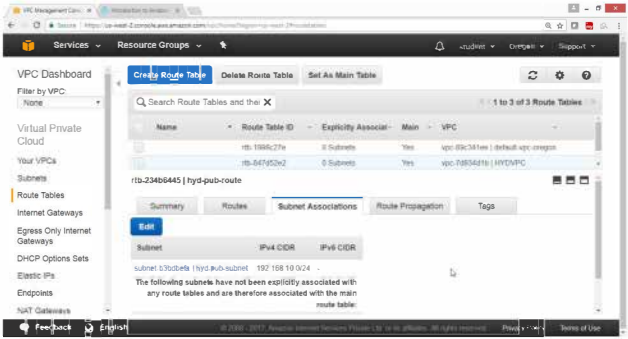


Select check box of hyd-pub-subnet → 192.168.10.0/24



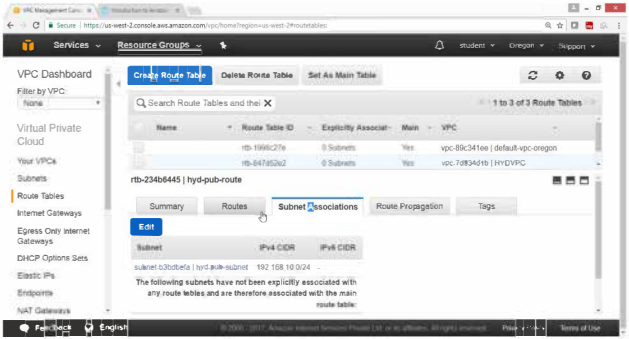
Verify

hyd-pub-subnet is associated with routing table

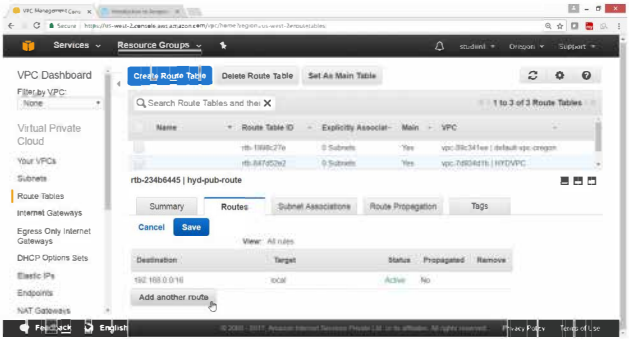


Click on **Route** Button

Click on **Edit** button



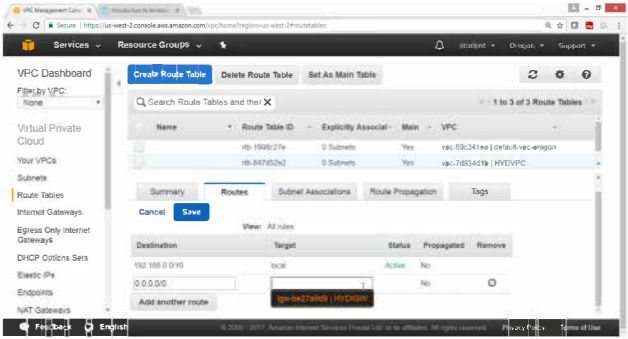
Click on **"Add another route"** button



For Destination → 0.0.0.0/0

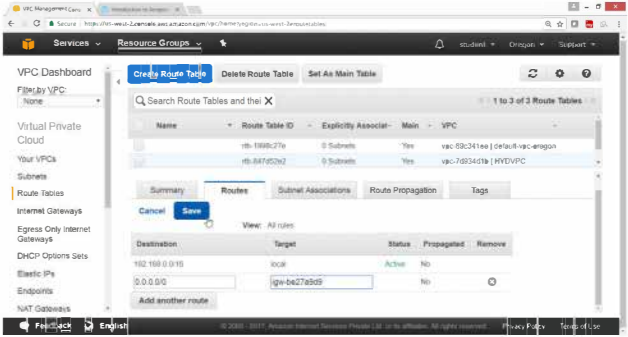
For Target → select HYDIGW

Click on **Save** button



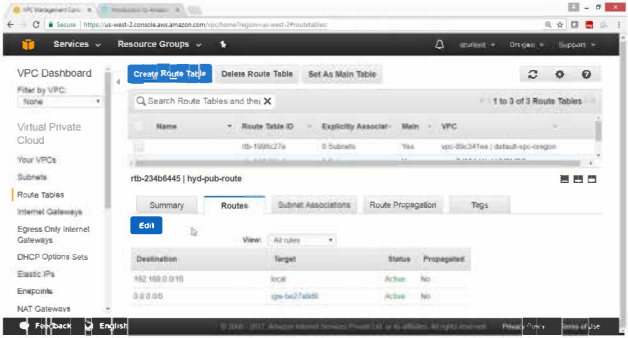
Verification

Public route is added through internet gateway



Verify

Status column show Active

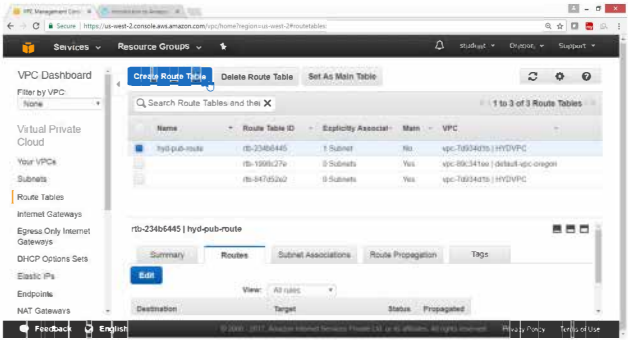


6) Create Private Routing Table, associate subnet and add routing rules

On **VPC Dashboard** panel

Select **Route Tables**

Click on **" Create Route Table"**

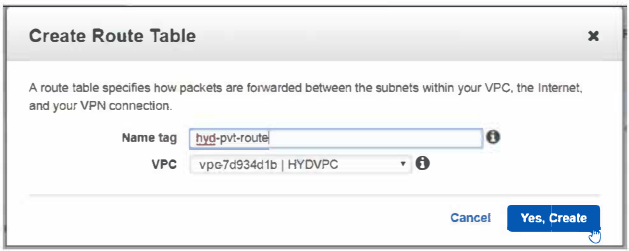


On **"Create Route Table"** box

For **Name tag->** hyd-pvt-route

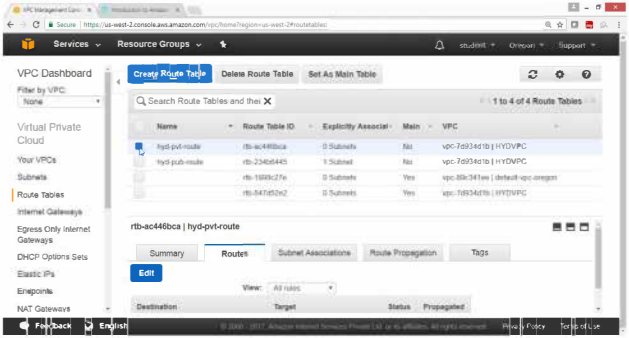
For VPC -> HYDVPC

Click on **"Yes, Create"** button

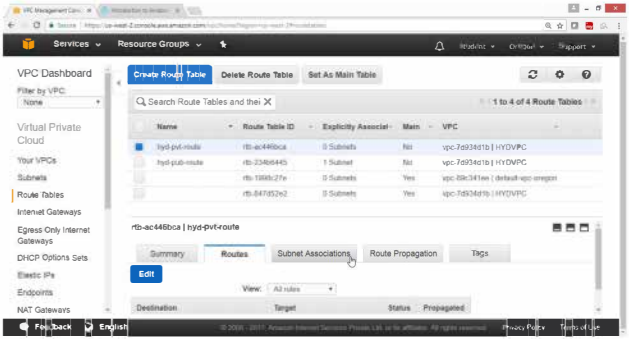


Verify

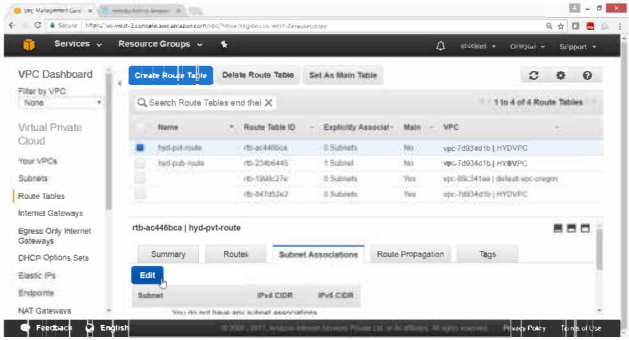
hyd-pvt-route table is create



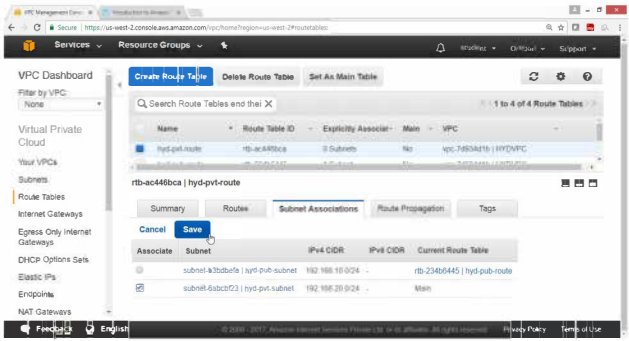
Click on **Subnet Association** button



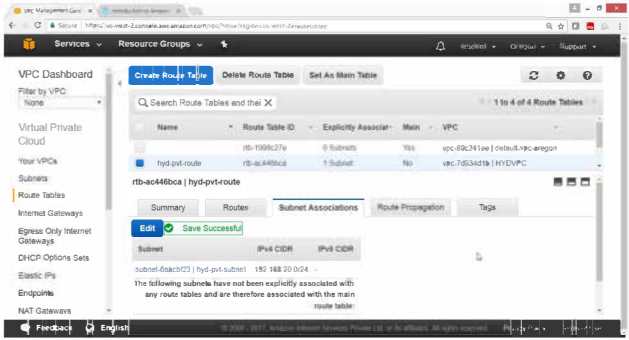
Click on **Edit** button



Select check box hyd-pvt-subnet → 192.168.20.0/24

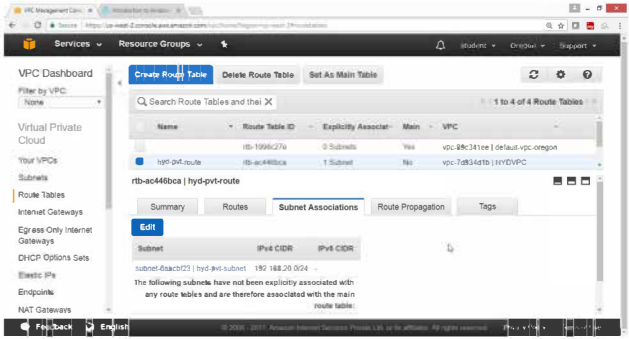


Click on **Save** button

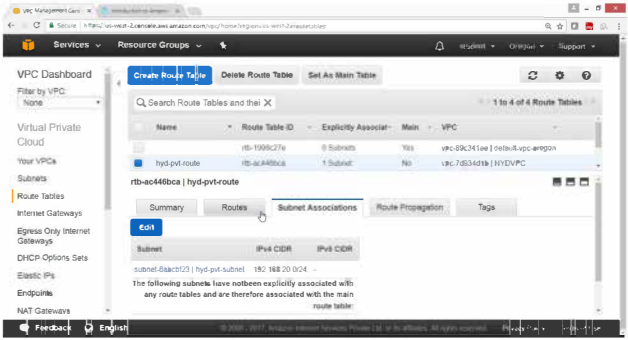


Verify

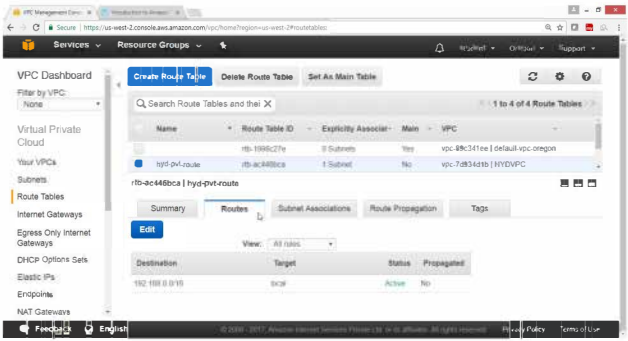
Hyd-pvt-subnet is associated with hyd-pvt-route table



Click on **Route** button



Note: No need to add IGW in pvt route

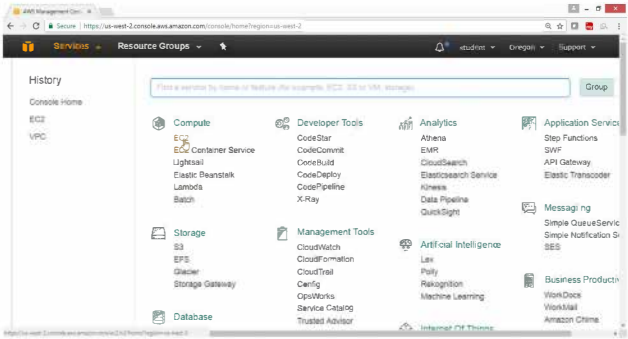


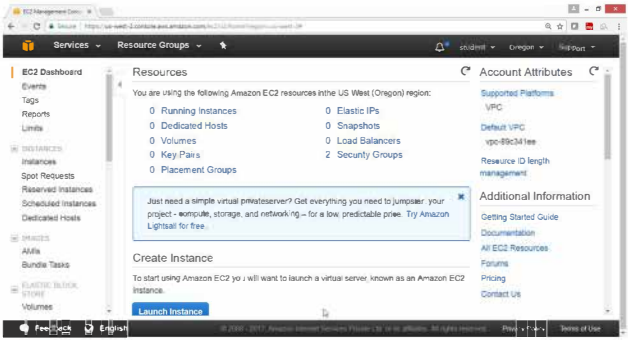
7) To launch Windows instance in Public subnet

Open the AWS console

Click on Services

Click on Ec2 services

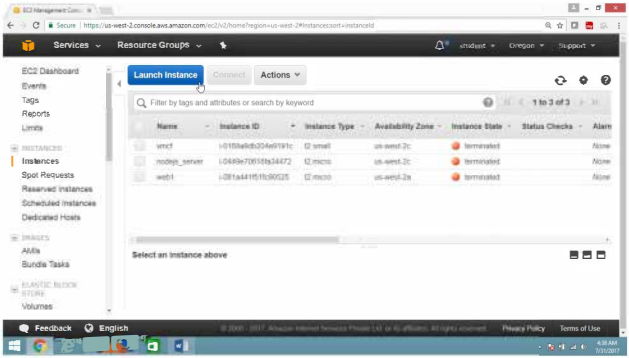




On the EC2 dashboard panel

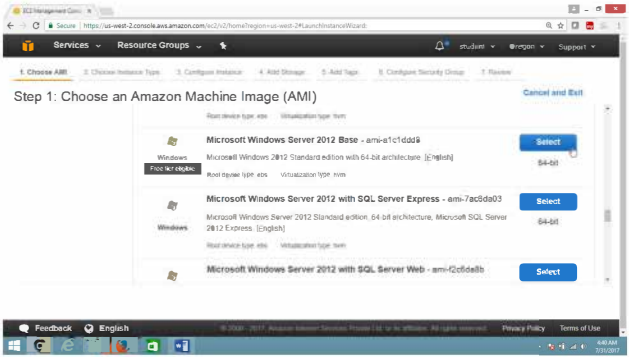
Click on **instance**

Click on **Launch instance** button



Select AMI **"Microsoft Windows Server 2012 Base** - ami-a1c1ddd8"

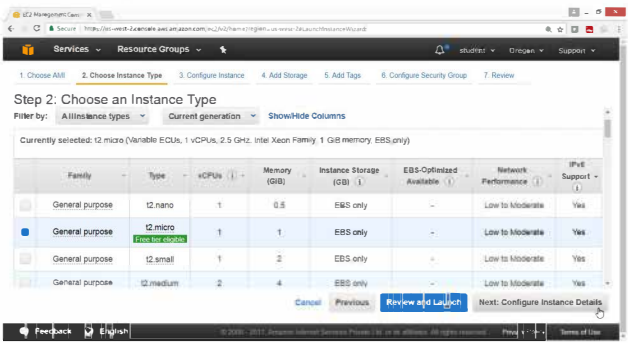
Free tier elegible



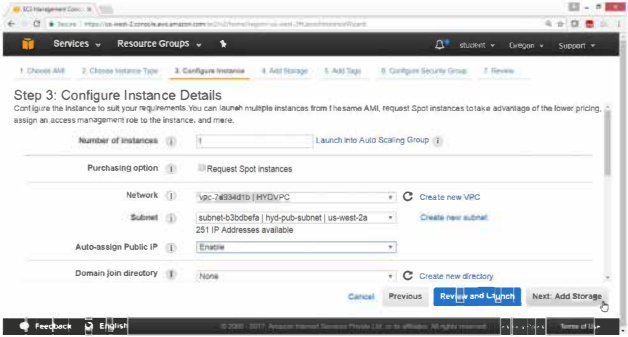
On the **"Choose an Instance Type"** page

Select **"General purposet2.micro"**

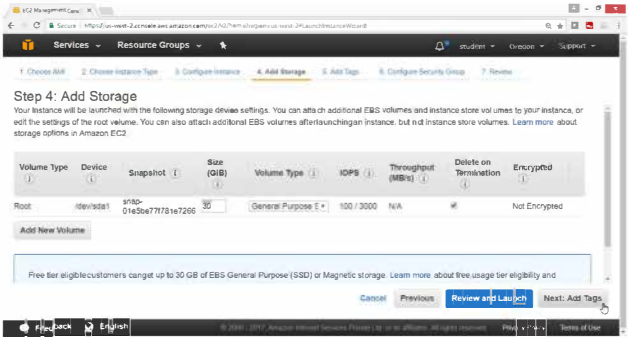
Click on **"Next Configure Instance Details"** button



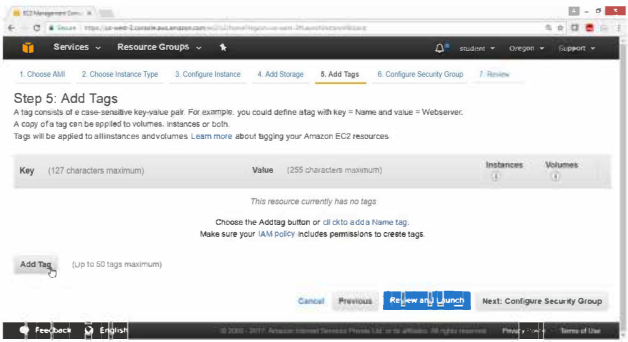
On the **"Configuration Instance Details"** page
For "Number of instances" -> **1**
For "Network" → HYDVPC
For "Subnet" → hyd-pub-subnet
For "Auto-assign Public IP" -> Enable
Click on **"Next: Add Storage"** button



On the **"Add Storage"** page
Take default values
Click on **"Next: Add tags"** button



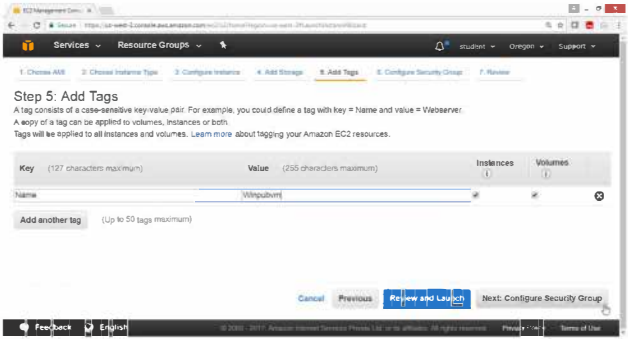
Click on **"Add tag"** button



For "Key" → Name

For Value → **W**inpubvm

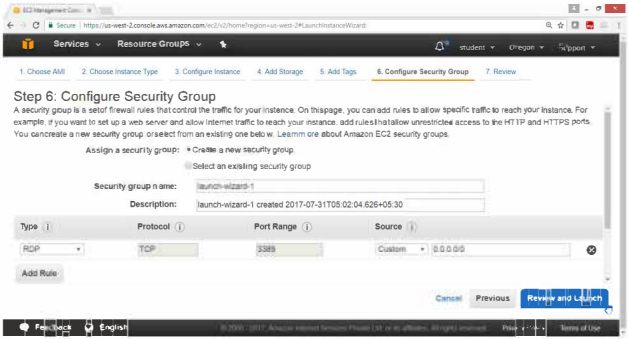
Click on **"Next: Configure Security Group"**



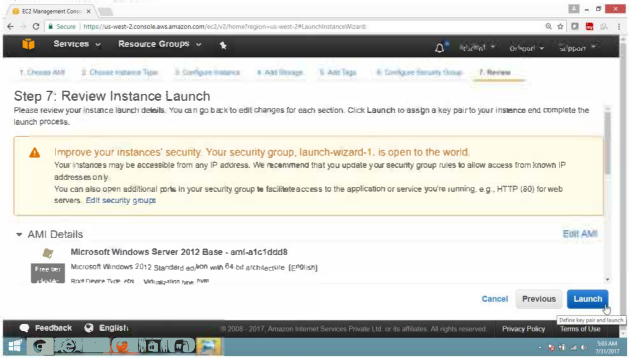
On the “Configure Security Group” page

Take Default Values

Click on **“Review and Launch”** button



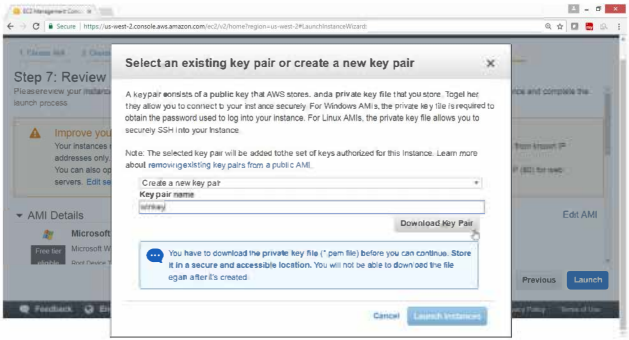
Click on **“Launch”** button



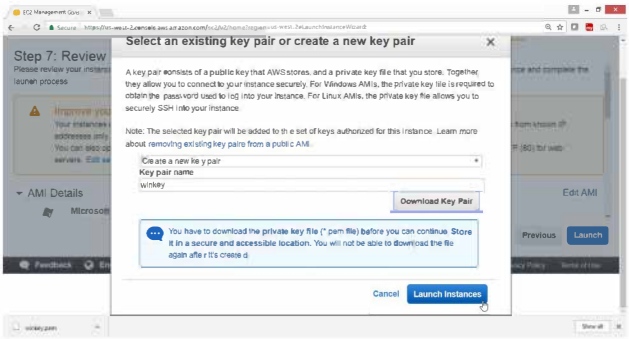
Select **"Create a new key pair"**

For **"Key pair name"** → **winkey**

Click on **"Download Key Pair"** button

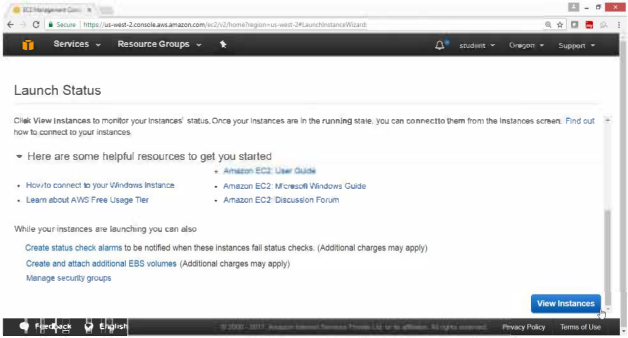


Click on **"Launch Instance"** button

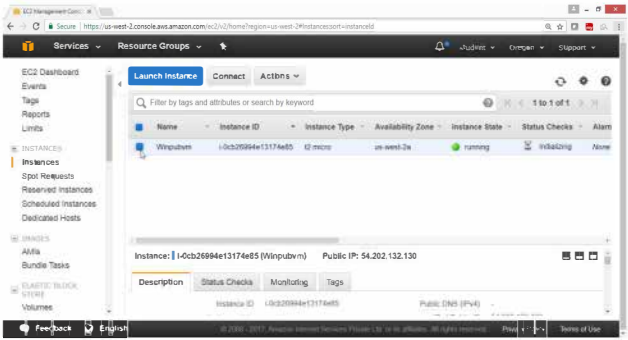


Check summary, Drag down

Click on **"View Instance"** button



Verify that instance is Running

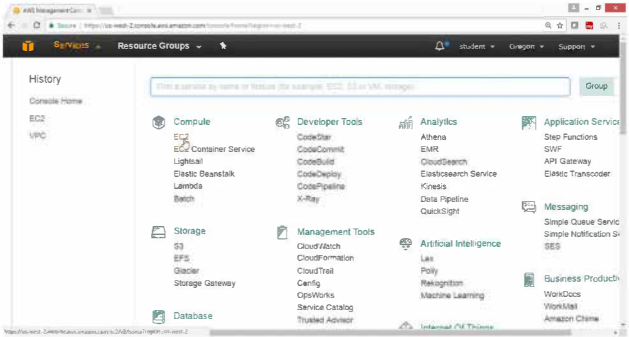


8) To Launch Windows instance in Private Subnet under HYDVPC VPC

Open the AWS console

Click on Services

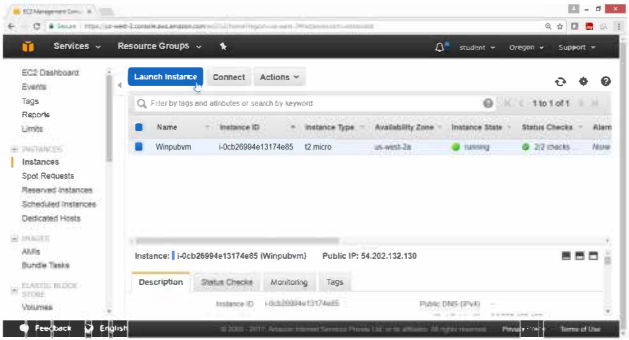
Click on **Ec2** services



On the **EC2 Dashboard** panel

Click on **Instance**

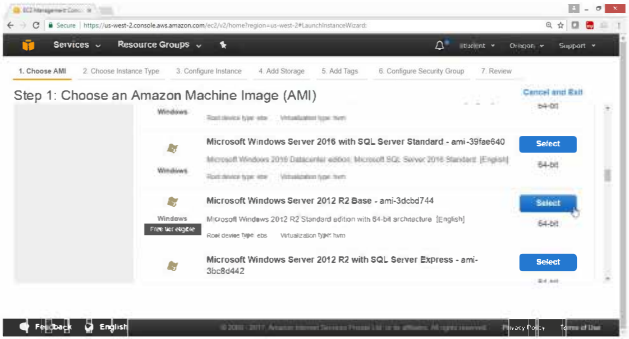
Click on "**Launch instance"** button



On the **"Choose an Amazon Machine Image (AMI) "** page

Select AMI **"Microsoft Windows Server 2012 R2 Base** - ami-a1c1ddd8"

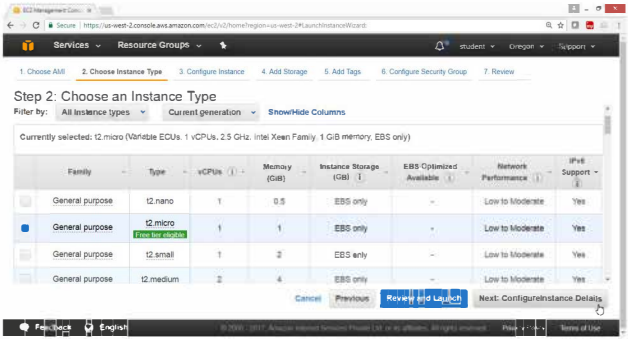
Free tier elegible



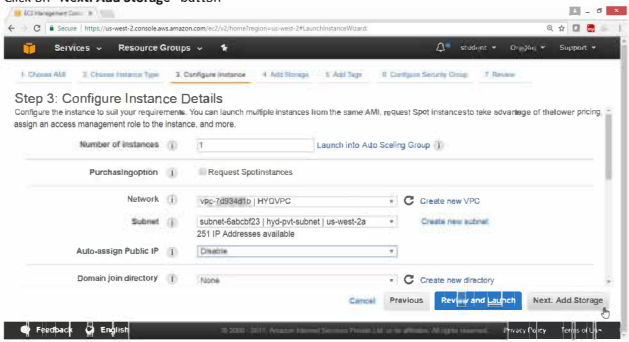
On the **"Choose an Instance Type"** page

Select **"General purposet2.micro"**

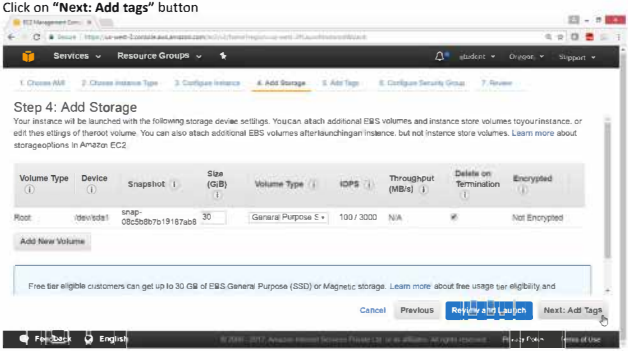
Click on **"Next Configure Instance Details"** button



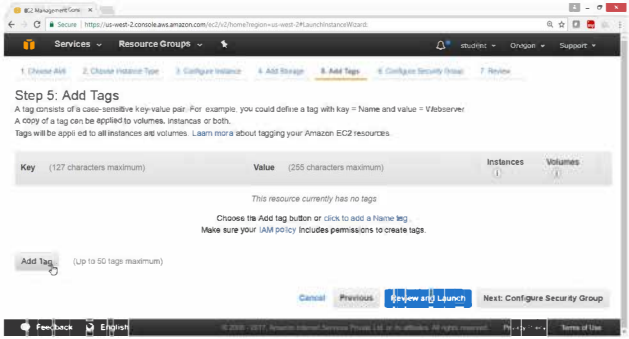
On the **"Configuration Instance Details"** page
For "Number of instances" -> **1**
For "Network" → HYDVPC
For "Subnet" → hyd-pvt-subnet
For "Auto-assign Public IP" -> Disabled
Click on **"Next: Add Storage"** button



On the **"Add Storage"** page
Take default values
Click on **"Next: Add tags"** button



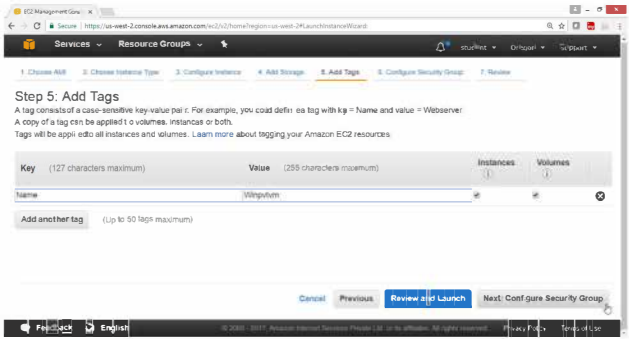
Click on **"Add tag"** button



For "Key" → Name

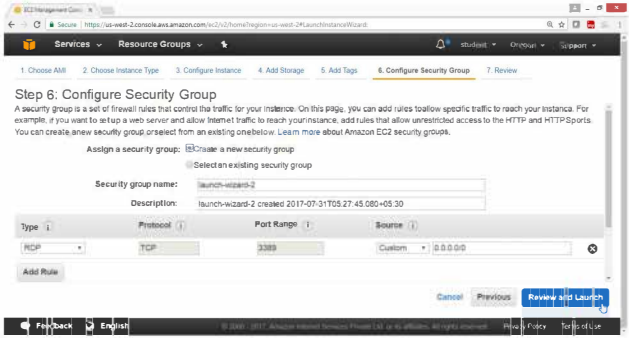
For Value → Winpvtvm

Click on **"Next: Configure Security Group"** button



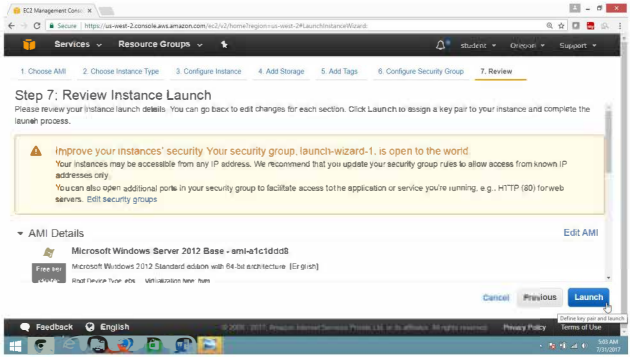
Take Default Values

Click on **"Review and Launch"** button



Drag down

Click on **"Launch"** button

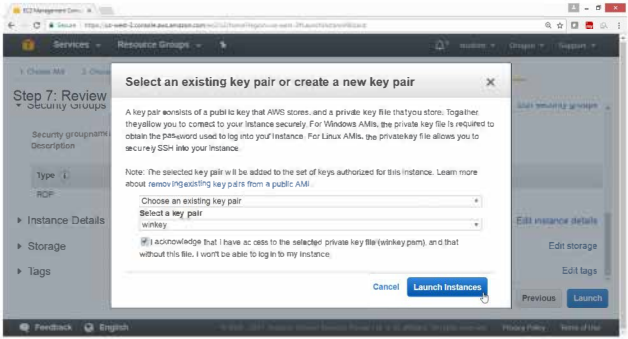


Select **"Choose an existing key pair"**

For **"Key pair name"** → **winkey**

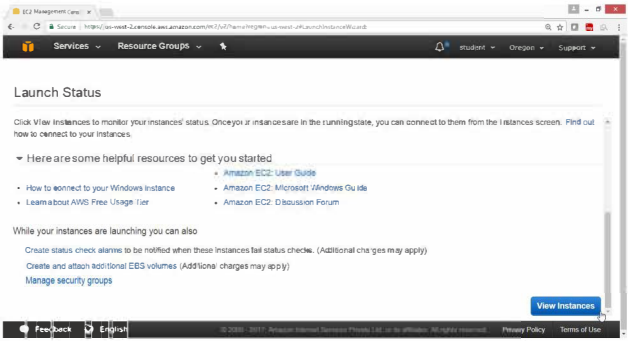
Select I acknowledge check box

Click on **"Launch Instance"** button

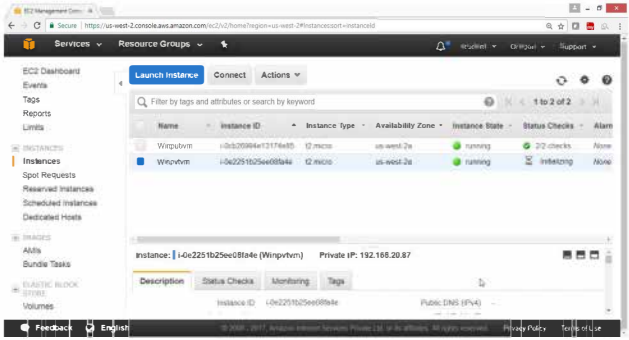


Check summary, Drag down

Click on **"View Instance"** button

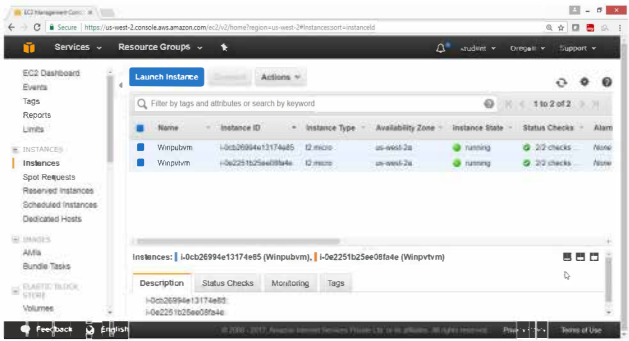


Verify that instance is Running

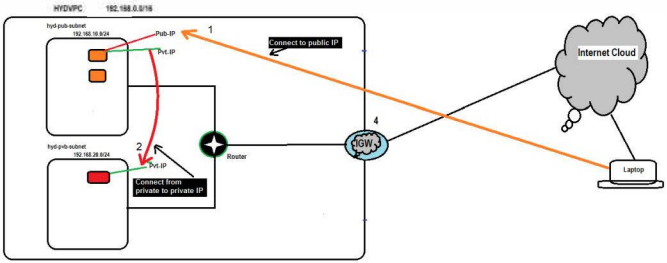


Verification

Output shows that both instances in public & private subnet are running.

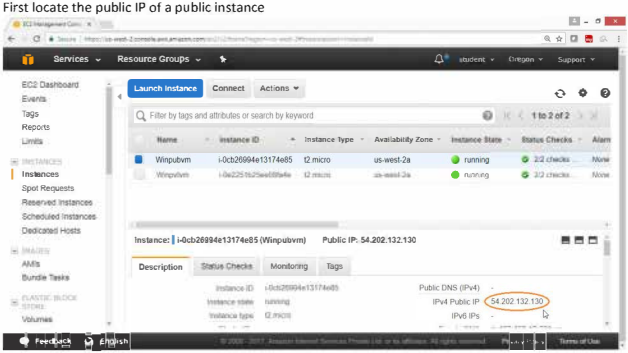


Now to connect an instance in private subnet first connect an instance in public network then from there connect to an instance in private subnet as shown in diagram

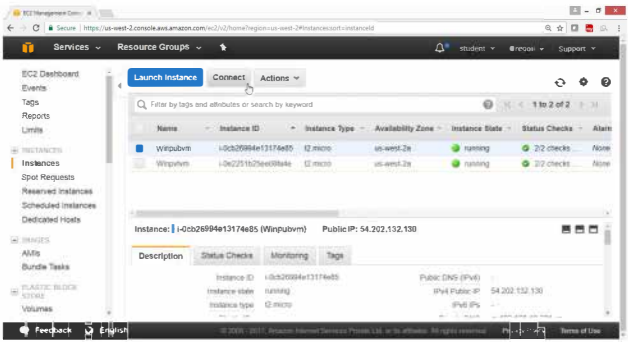


9) To Connect to Public subnet instance

First locate the public IP of a public instance

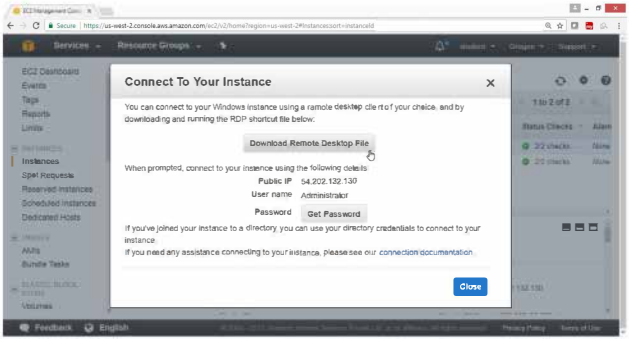


Click on **"Connect"** button



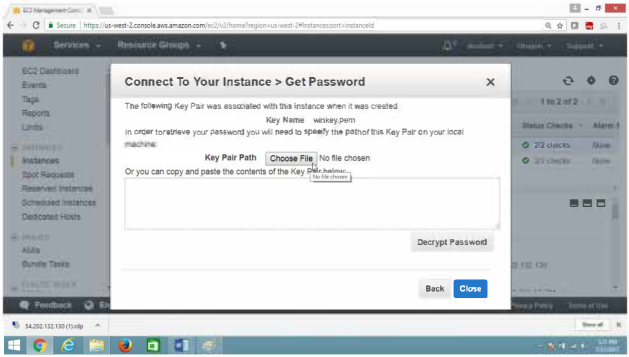
Click on **"Download Remote Desktop file"**

Click on **"Get Password"**



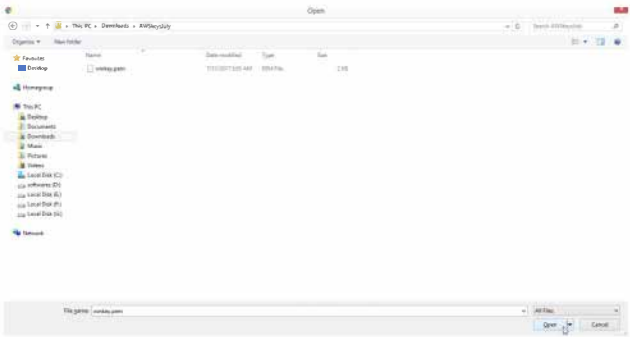
Provide the path of key file

Click on **Choose file** button

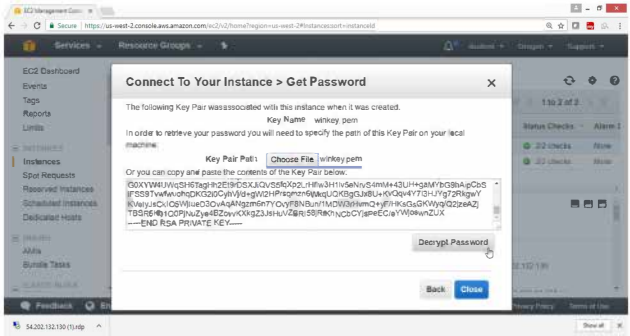


Select the key file

Click on **Open** button



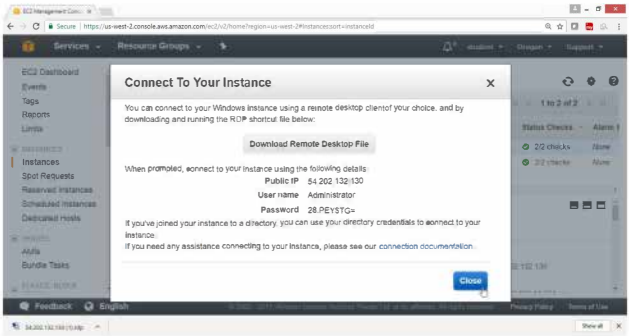
Now click on **"Decrypt Password"** button



Verification

Password is generated copy in notepad

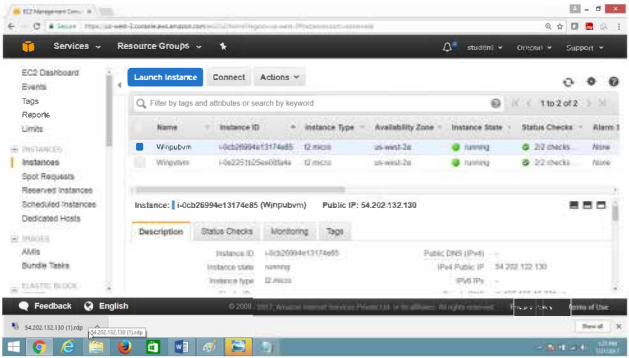
Click on **Close** button



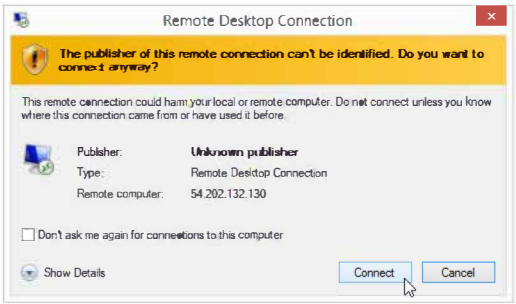
Double Click on RDP file

Provide Windows Username → Administrator

Password → "28.PEY$TG=" , as shown above

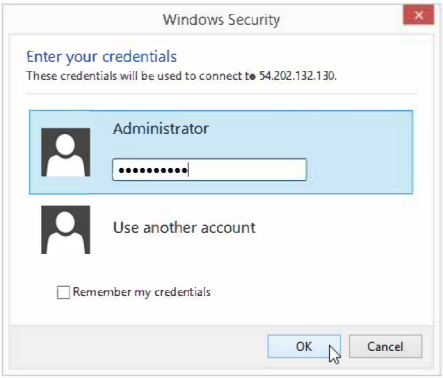


Click on **"Connect"** button

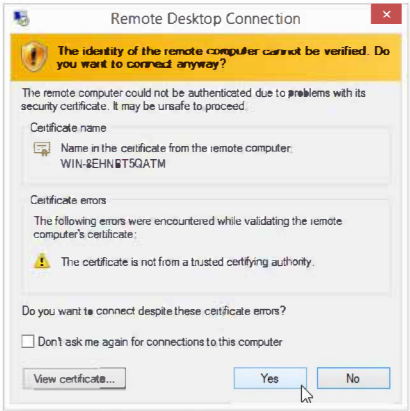


Paste the password

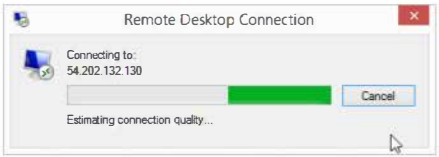
Click on **OK** button



Click on **Yes** button



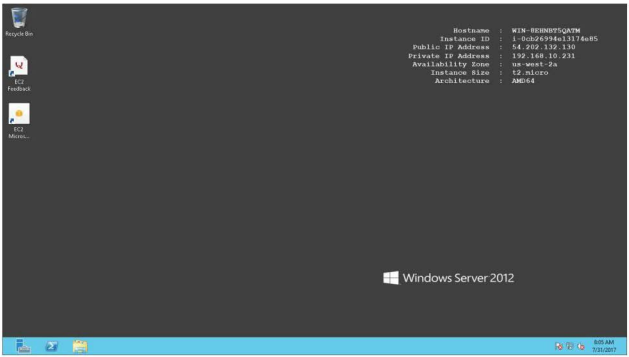
Verify



Verification

Now you are connected to Windows Public instance

On Windows Desktop public and private both IP's are displayed

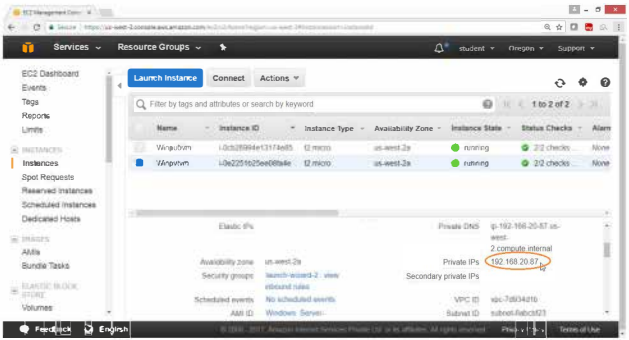


10) To Connect to Private subnet instance

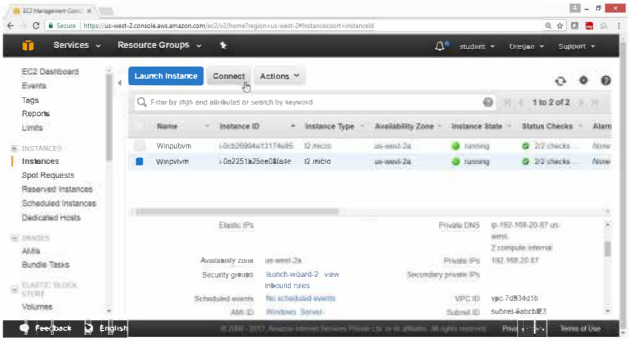
Go to Ec2 Dashboard

Select private instance

Get the private IP of the instance

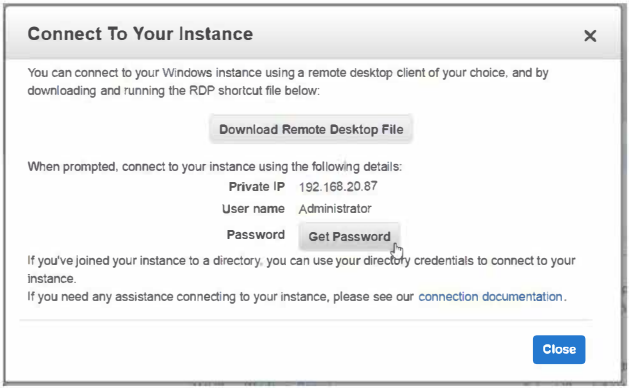


Click on **Connect** button



To get the password

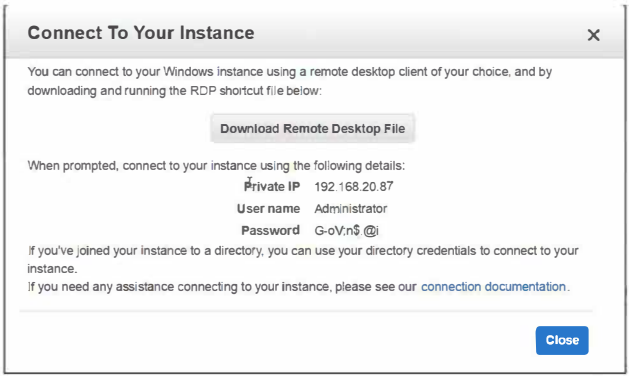
Click on **"Get Password" button**



Click on **"Decrypt Password"**

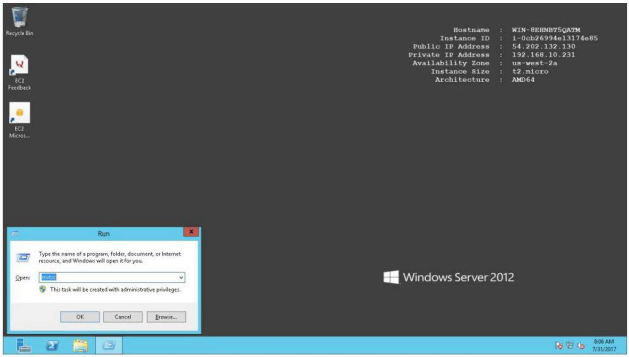
Verify

IP and password of private subnet instance is provided



Now logging to public instance

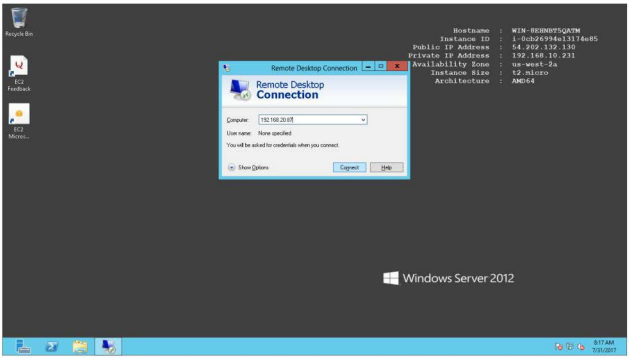
Open Run and type mstsc to connect to window private instance



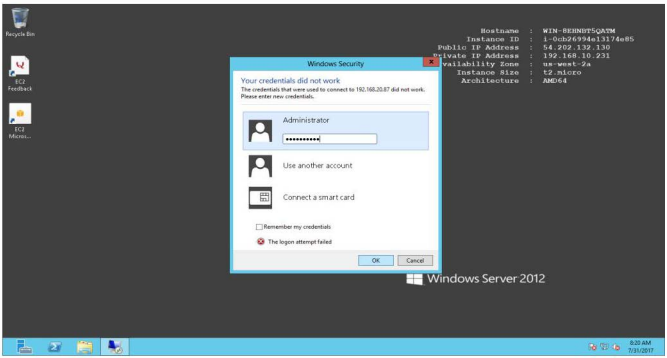
Provide private instance

Private IP →192.168.20.87
Username → Adminsitrator

Password →G-oV;n$.@i



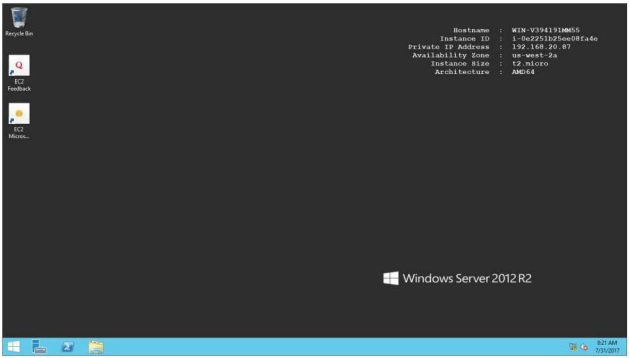
Now Provide Username & password



Verification

Check private IP at Right top corner

Now you are connected to windows private instance.



11) To connect to linux instance in private subnet

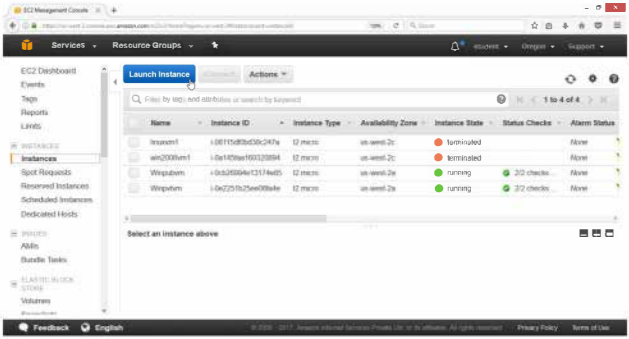
Launch linux instance in public subnet → hyd-pub-subnet

Open the AWS console

Click on Services

Click on Instance

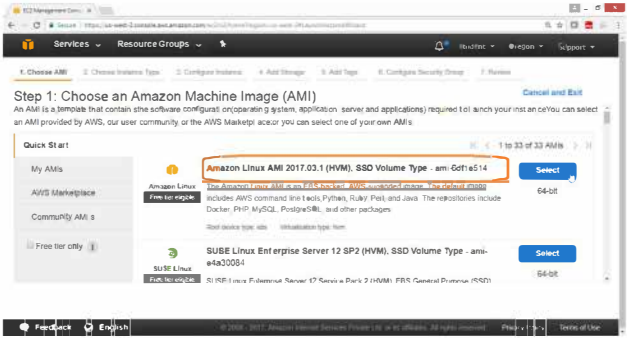
Click on **"Launch Instance"** button



On the **"Choose an Amazon Machine Image (AMI)"** page

Select AMI **"Amazon Linux AMI 2017.03.1 (HVM), SSD Volume Type** - ami-6df1e514

Click on **Select** button

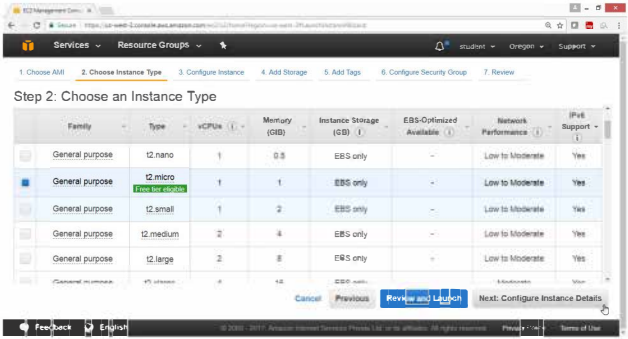


On the **"Choose an Instace Type"** page

Select **"General purpose"**

Type →**t2.micro**

Click on **"Next: Configure Instance Details"**



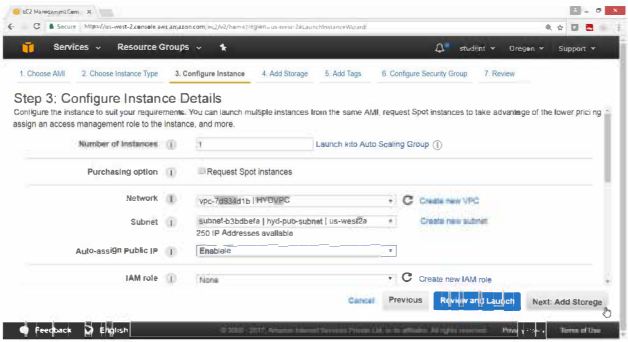
On the **"Configure Instance Details" page**

Number of instance → 1

Network → HYDVPC

Subnet → hyd-pub-subnet

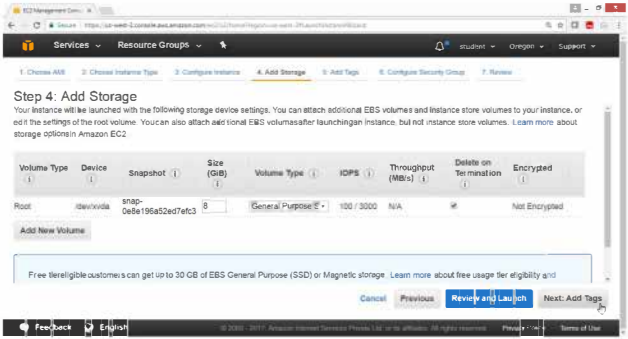
Auto-assign Public IP → Enable



On the **"Add Storage"** page

Leave the values as default

Click on **"Next: Add Tags"** button

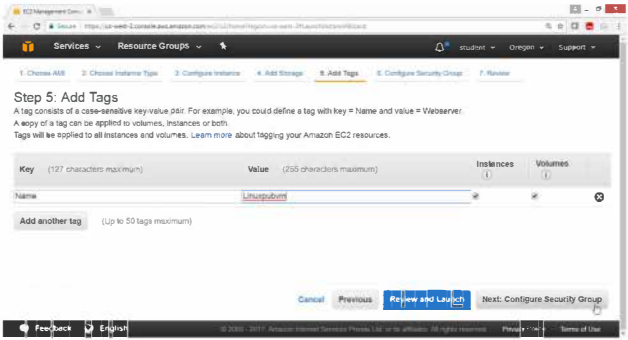


On the **"Add Tags"** page

Key → Name

Value → Linuxpubvm

Click on **"Next: Configure Security Group"** button

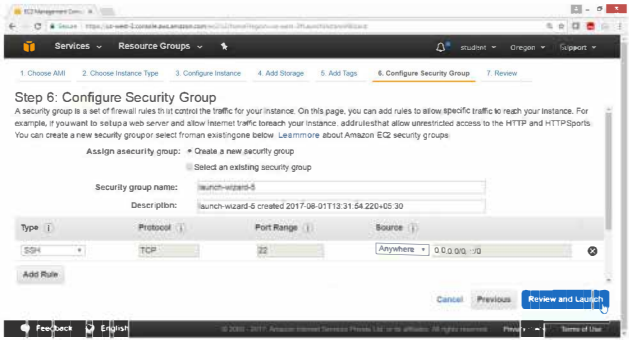


On the **"Configure Security Group"** page

Assign a security group → Create a new security group

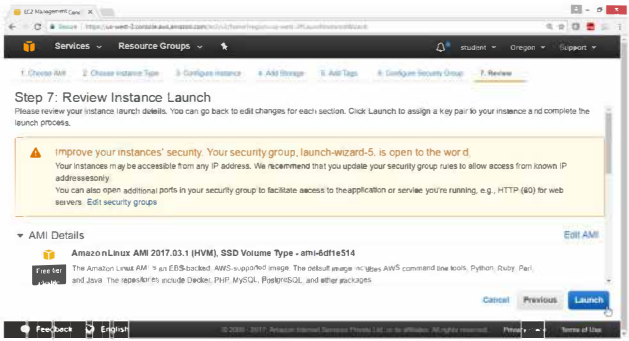
Leave remaining values as default

Click on **Review and Launch** button



On the **"Review Instance Launch"** page

Click on **Launch** button

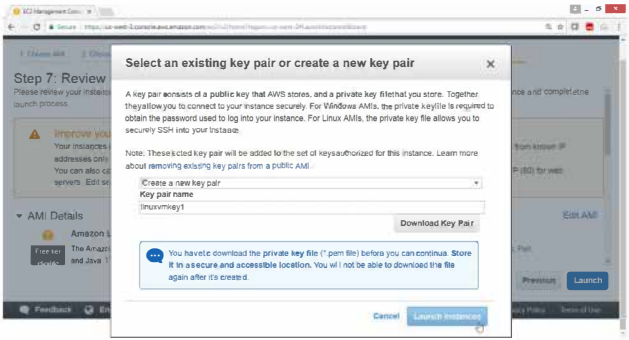


On the "Select an existing key pair or create a new key pair" page

Select **Create a new key pair**

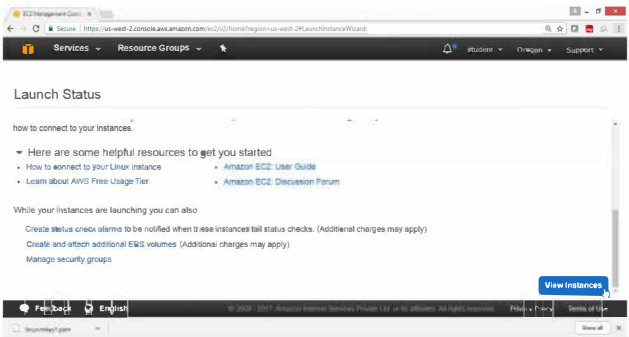
Key pair name -→ linuxvmkey1

Click on **"Launch Instance"** button



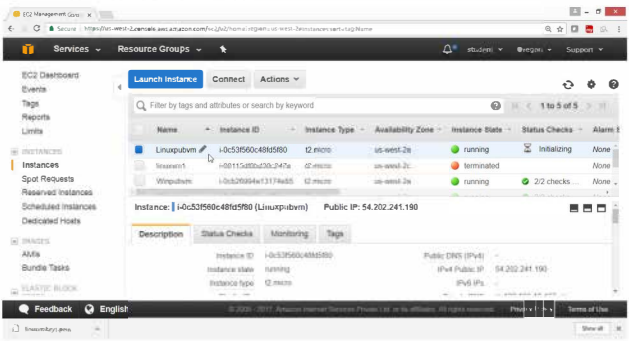
Check the summary

Click on **View Instance** button



Verification

Linux instance in public subnet is launched



12) To connect to linux instance in private subnet

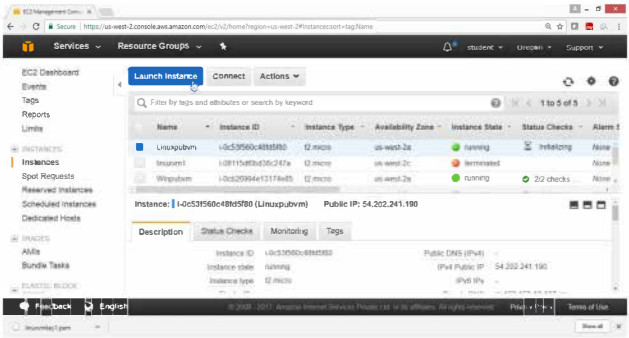
Launch linux instance in private subnet → hyd-pvt-subnet

Open the AWS console

Click on Services

Click on Instance

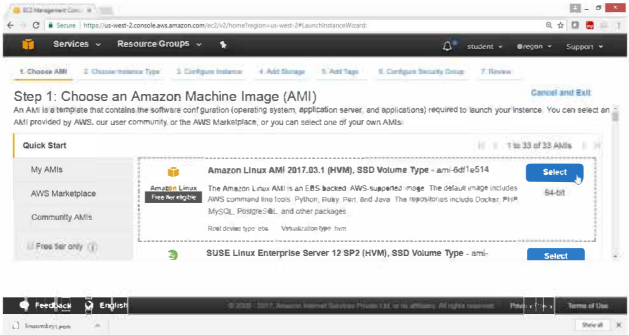
Click on **"Launch Instance"** button



On the **"Choose an Amazon Machine Image (AMI)"** page

Select AMI **"Amazon Linux AMI 2017.03.1 (HVM), SSD Volume Type** - ami-6df1e514

Click on **Select** button

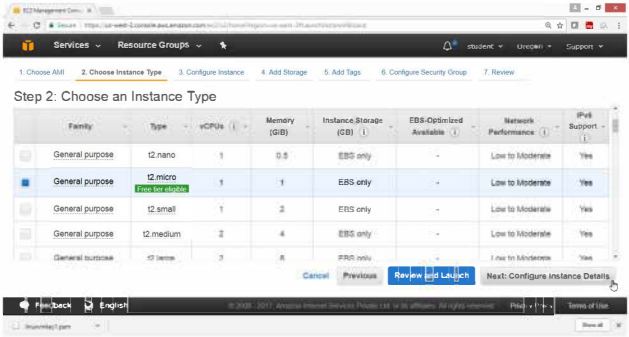


On the **"Choose an Instace Type"** page

Select **"General purpose"**

Type **→t2.micro**

Click on **"Next: Configure Security Group"** button



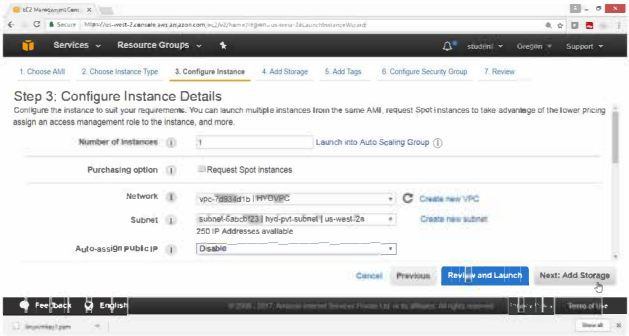
On the **"Configure Instance Details" page**

Number of instance → 1

Network → HYDVPC

Subnet → hyd-pvt-subnet

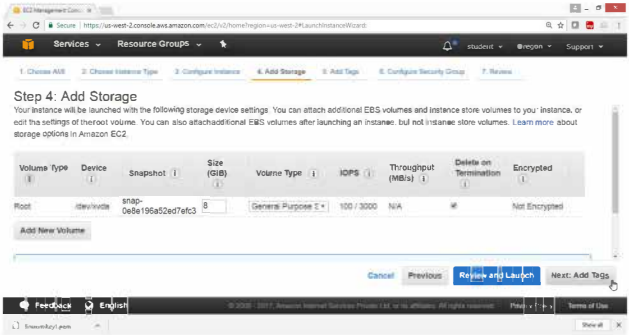
Auto-assign Public IP → Disable



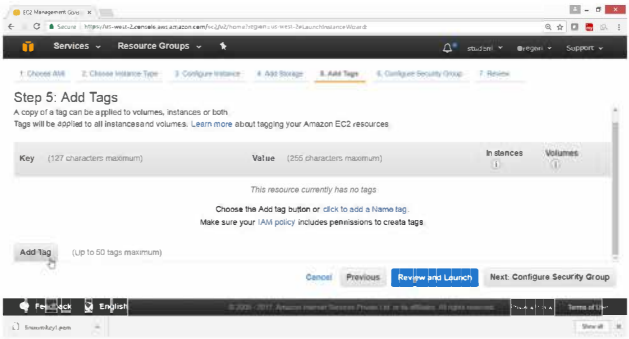
On the **"Add Storage"** page

Leave the values as default

Click on **"Next: Add Tags"** button



Click on **Add Tag**

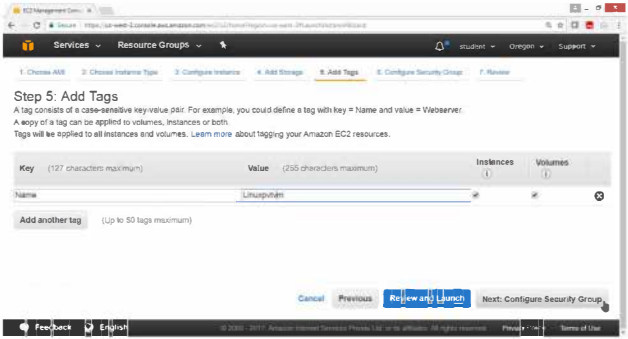


On the **"Add Tags"** page

Key → Name

Value → Linuxpvtvm

Click on **"Next: Configure Security Group"** button

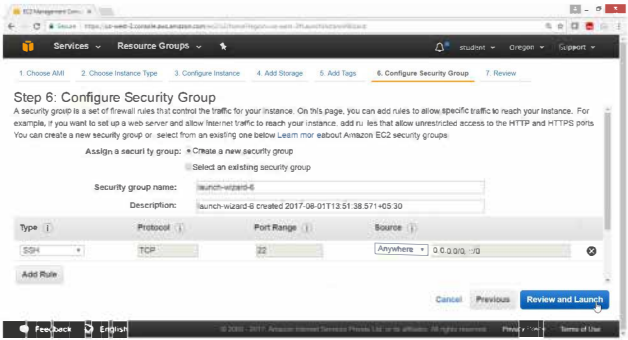


On the **"Configure Security Group"** page

Assign a security group → Create a new security group

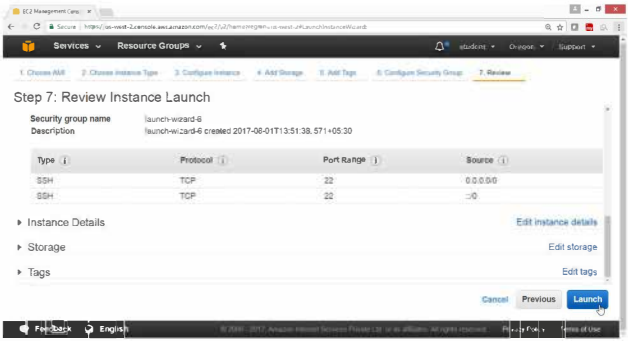
Leave remaining values as default

Click on **"Review and Launch"** button



On the **"Review Instance Launch"** page

Click on **Launch** button

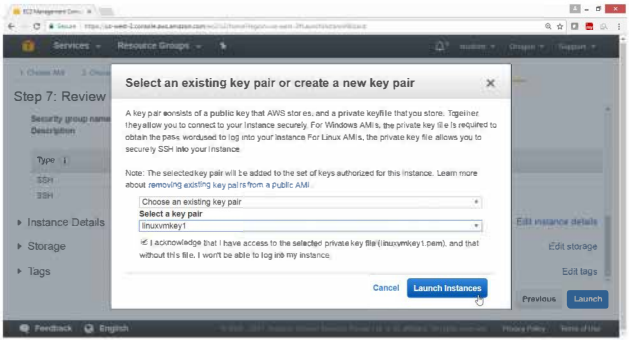


On the **"Select an existing key pair or create a new key pair"** box

Select **Create a new key pair**

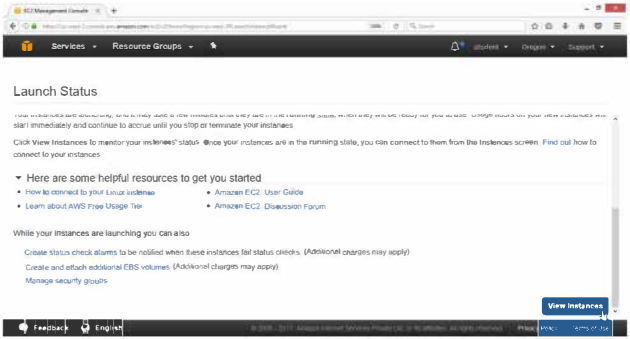
Key pair name → linuxvmkey1

Click on **"Launch Instance"** button



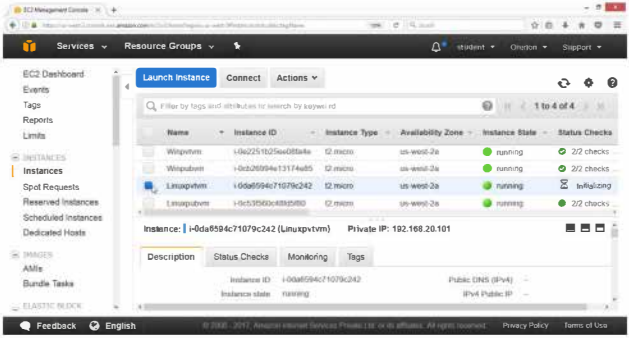
Check the summary

Click on **View Instance** button



Verification

Linux instance in public subnet is launched



To connect to linux private instance

First copy the key to linux instance in public subnet

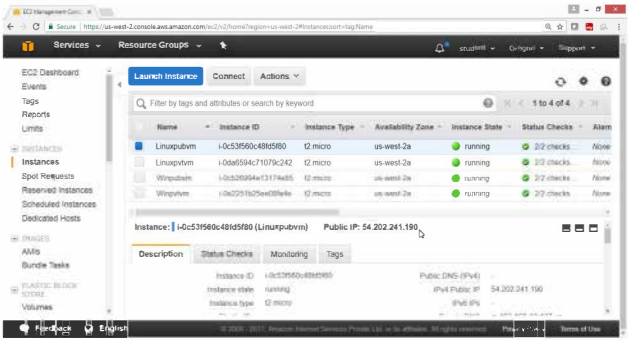
Now connect to linux instance in public

Then connect to linux instance in private

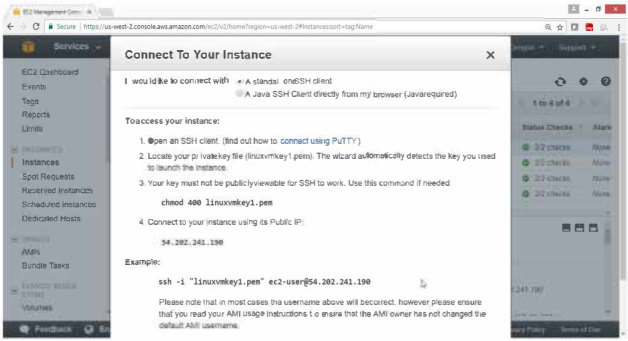
Open Mobaxterm

Coping *.pem file to linux instance in public

Select public linux instance click on connect

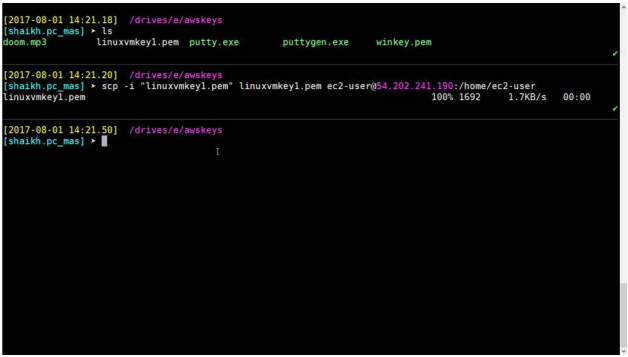


View the guide lines



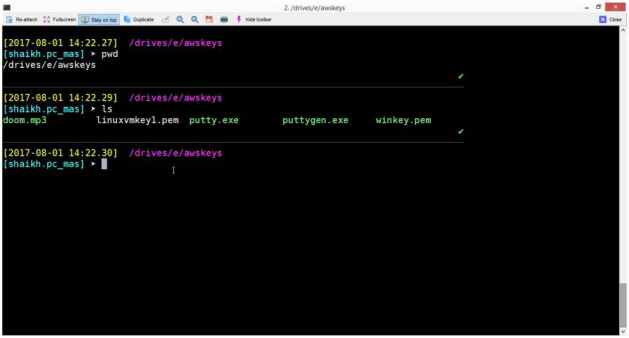
Use the above public ip of linux instance in mobaxterm

Copy *.pem file to pun linux instance using scp command

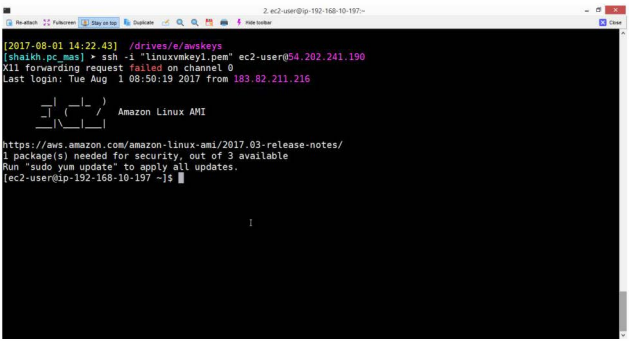


Verify

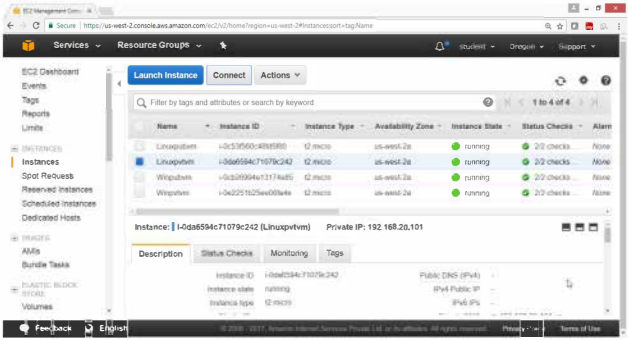
Use commands , pwd, ls to check *.pem file



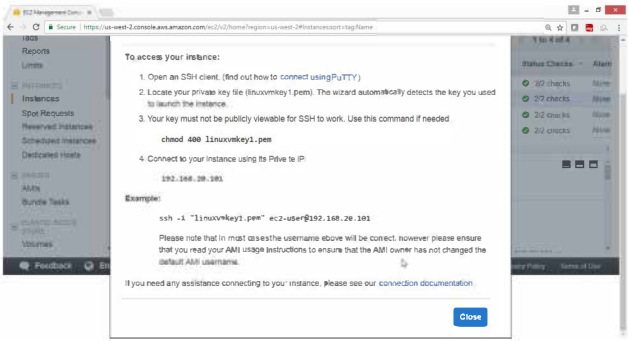
Now connect to public instance using ssh command



Select private instance and get private ip



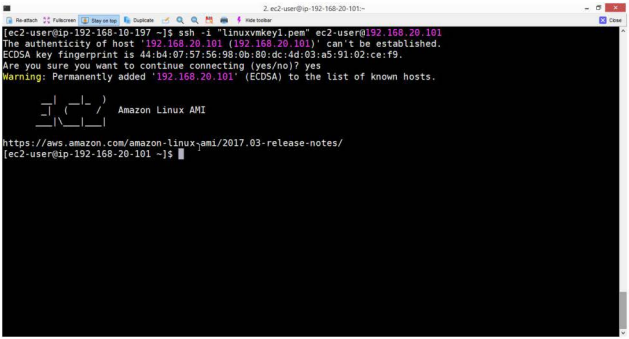
View the details of private instance



Verification

Run ssh command to login to private instance

Now you are connected to private instance in private subnet

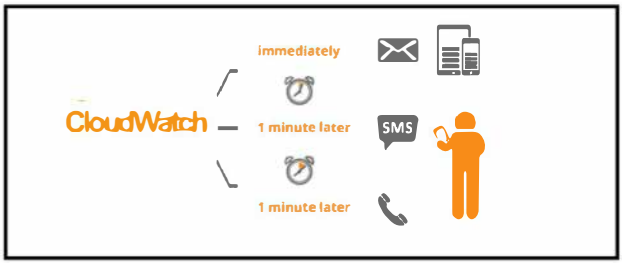


Lab 10: To Configure Amazon CloudWatch

OBJECTIVE

To configure CloudWatch to monitor CPU Utilization

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK :

Creating Alarm

Select Notification

Check mail to verify

1) To Configure Amazon CloudWatch Service

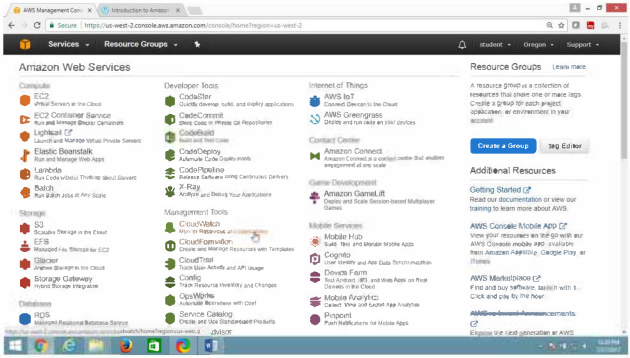
Launch a Amazon linux instance, then

Open **AWS** Console

Click on **Services**

In the **Management Tools** section

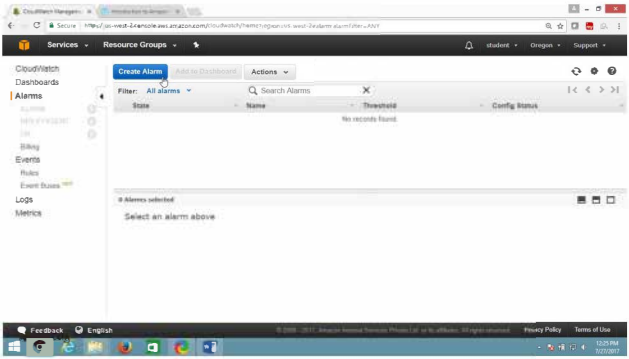
Click on **CloudWatch**



On "**CloudWatch"**, panel

Select **Alarms**

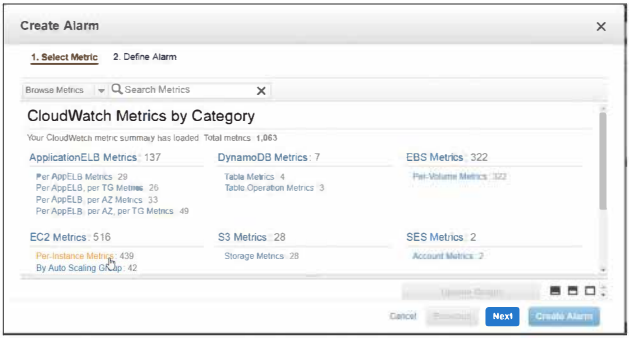
Click on "**Create Alaram"** button



In **"Create Alarm"** page

Select **"EC2 Metrics"**

Click on **"Per-instance Metrics"**



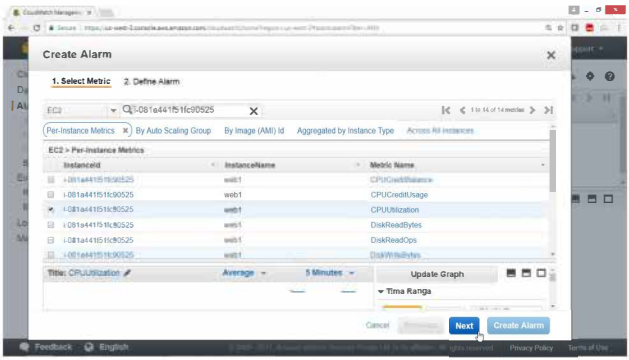
From **"Create Alarm"** page

Select **"1. Select Metric"**

In search box provide instance ID or Name

Under Metric Name, select **CPUUtilization** checkbox

Click on **Next** button



On **Create Alarm** page

Select " **2. Define Alarm**"

Under Alarm Threshold

Name → testcpuutlization

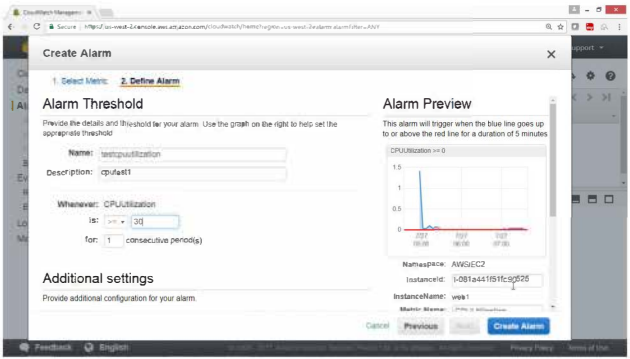
Description→ cputest

Under **W**henever CPUUtilization

is >= **30**

for **1** consecutive periods

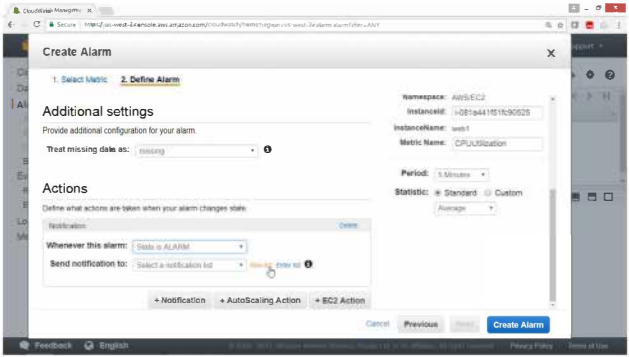
Drag Down



Under Actions

Whenever this alarm → State is Alaram

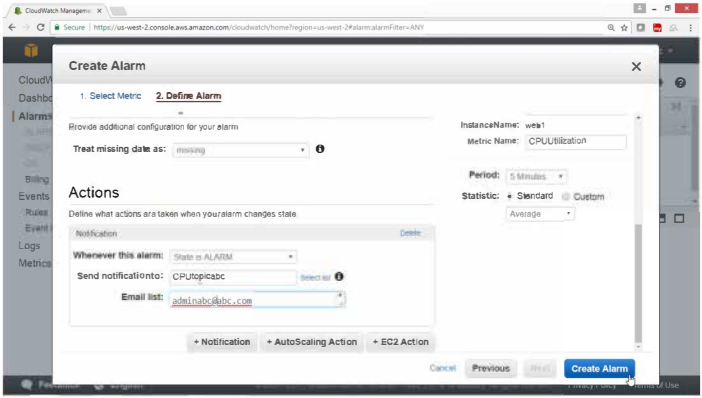
Send notification to → Click on New list



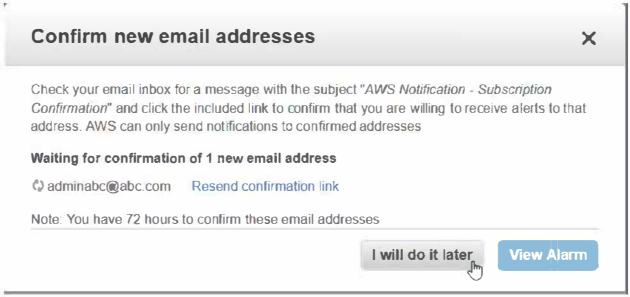
Send notification to → CPUtopicabc

Email → adminabc@abc.com

Click on **"Create Alaram"** button



Click on "**I will do it Later**" button.

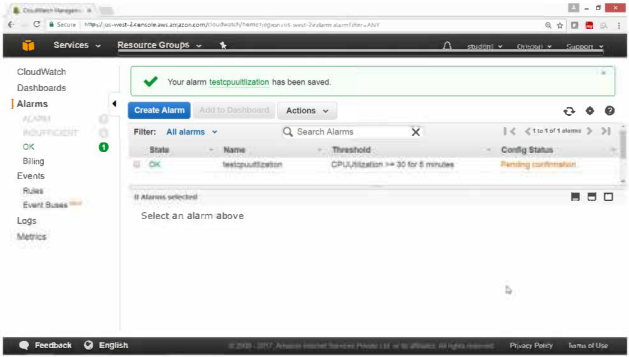


Go to your Email account and check the Mail

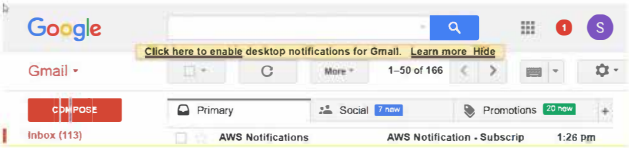
Once mail is been checked

Config status → Pending confirmation

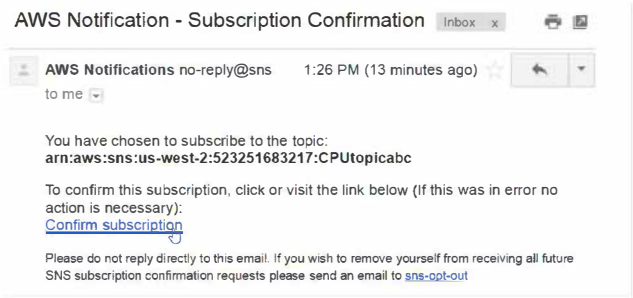
Verify the link from your Email



Open your email

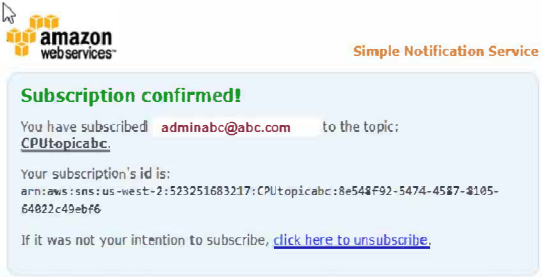


Click on **"Confirm subscription"**



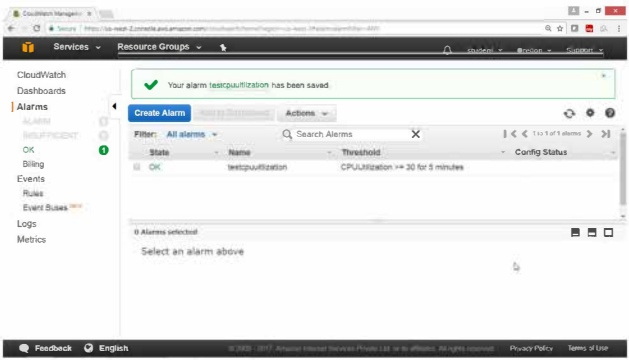
Verified by this output

==========

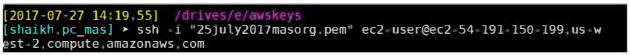


==========

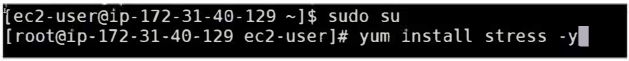
After confirmation from email **Config status** has become blank



Now login to Instance using mobaxterm



Switch to root user and install stress command



Login to another terminal-2

Run top command

Verify output

CPU status is 100% idle

```
top - 08:56:26 up  1:53,  2 users,  load average: 0.00, 0.00, 0.00
Tasks:  94 total,   1 running,  93 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   1017372k total,   166080k used,   851292k free,     9224k buffers
Swap:        0k total,        0k used,        0k free,    90380k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      20   0 19628 2420 2108 S  0.0  0.2   0:00.90 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.00 ksoftirqd/0
    4 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kworker/0:0
    5 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kworker/u30:0
    7 root      20   0     0    0    0 S  0.0  0.0   0:00.03 rcu_sched
    8 root      20   0     0    0    0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      RT   0     0    0    0 S  0.0  0.0   0:00.00 migration/0
   10 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 lru-add-drain
   11 root      20   0     0    0    0 S  0.0  0.0   0:00.00 cpuhp/0
   12 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kdevtmpfs
   13 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   16 root      20   0     0    0    0 S  0.0  0.0   0:00.01 xenwatch
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.02 kworker/u30:2
   21 root      20   0     0    0    0 S  0.0  0.0   0:00.00 xenbus
  139 root      20   0     0    0    0 S  0.0  0.0   0:00.00 khungtaskd
  140 root      20   0     0    0    0 S  0.0  0.0   0:00.00 oom_reaper
  141 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 writeback
  143 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kcompactd0
  144 root      25   5     0    0    0 S  0.0  0.0   0:00.00 ksmd
  145 root      39  19     0    0    0 S  0.0  0.0   0:00.00 khugepaged
  146 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 crypto
  147 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kintegrityd
```

Run this command in terminal -1 which will increase the load

 # stress --cpu 40 --timeout 1000

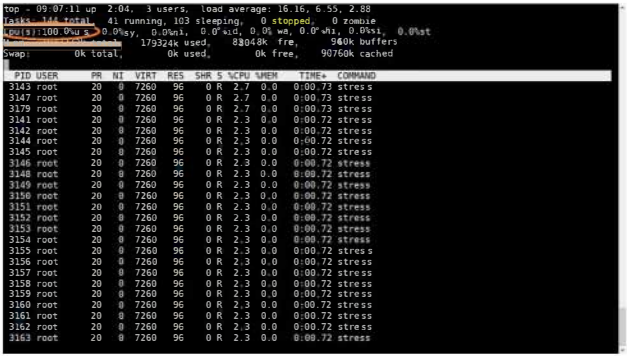
```
[root@ip-172-31-40-129 ec2-user]# stress  --cpu  40  --timeout 1000
stress: info: [3095] dispatching hogs: 40 cpu, 0 io, 0 vm, 0 hdd
```

Now check the status in another terminal-2 by running top command

 # top

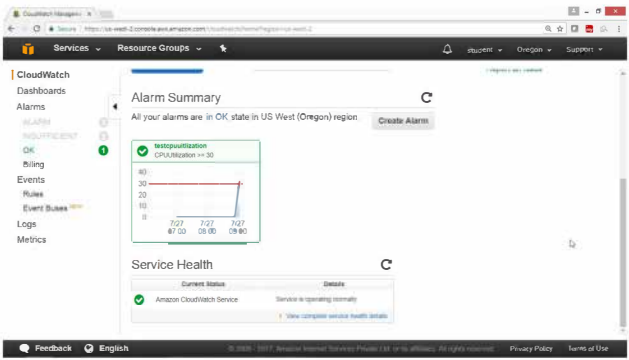
Verify the output

Cpu load is 100%

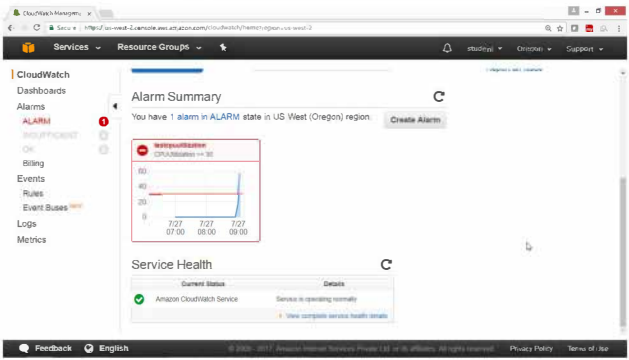


Go to CloudWatch service

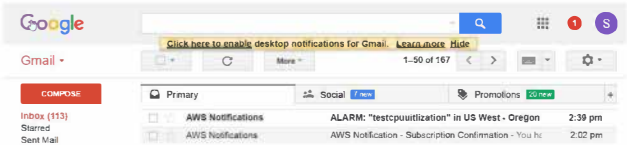
Check the status



After 5 minutes Alarm is generated



Go to email and check mail



Click on mail

Verify output

==========

AWS Notifications no-reply@sns.a 2:39 PM (2 minutes ago)

to me

You are receiving this email because your Amazon CloudWatch Alarm "testcpuuitlization" in the US West - Oregon region has entered the ALARM state, because "Threshold Crossed: 1 datapoint [46.236000000000004 (27/07/17 09:04:00)] was greater than or equal to the threshold (30.0)." at "Thursday 27 July, 2017 09:09:58 UTC".

View this alarm in the AWS Management Console:
https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#s=Alarms&alarm=testcpuuitlization

Alarm Details:
- Name: testcpuuitlization
- Description: cputest
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 datapoint [46.236000000000004 (27/07/17 09:04:00)] was greater than or equal to the threshold (30.0).
- Timestamp: Thursday 27 July, 2017 09:09:58 UTC

- Timestamp: Thursday 27 July, 2017 09:09:58 UTC
- AWS Account: 523251683217

Threshold:
- The alarm is in the ALARM state when the metric is GreaterThanOrEqualToThreshold 30.0 for 300 seconds.

Monitored Metric:
- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-081a441f51fc90525]
- Period: 300 seconds
- Statistic: Average
- Unit: not specified

State Change Actions:
- OK:
- ALARM: [arn:aws:sns:us-west-2:523251683217:CPUtopicabe]
- INSUFFICIENT_DATA:

State Change Actions:
- OK:
- ALARM: [arn:aws:sns:us-west-2:523251683217:CPUtopicabe]
- INSUFFICIENT_DATA:

--
If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.us-west-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-west-2:523251683217:CPUtopicabe:e8d238f8-8e77-46ec-8b2f-609f9ba26876&Endpoint= adminabc@abc.com

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support

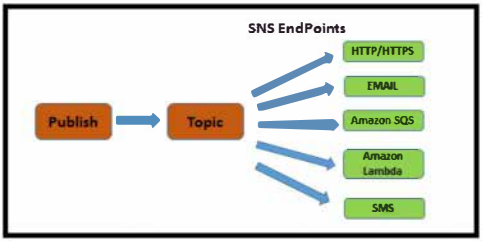
========== **END OF OUTPUT** ==========

Lab 11: To Configure Amazon Simple Notification Service (SNS)

OBJECTIVE

To configure Amazon Simple Service (SNS)

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with AmazonSNSFullAccess

TASK :

Create a Topic

Subscribe your topic

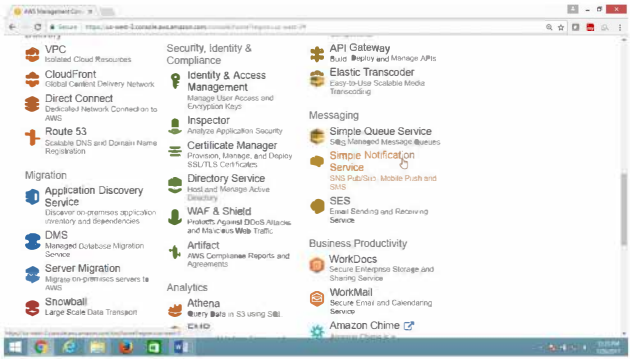
Veriy in your mail account

1) To configure Amazon Simple Notification Service (SNS)

Open AWS console

Select "**Messaging**" service

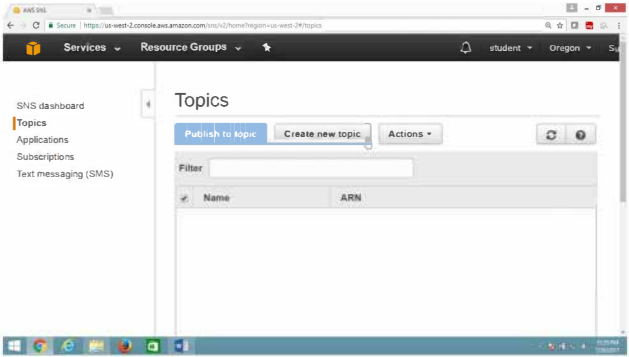
Click on "**Simple Notification service**"



From "**SNS Dashboard"** panel

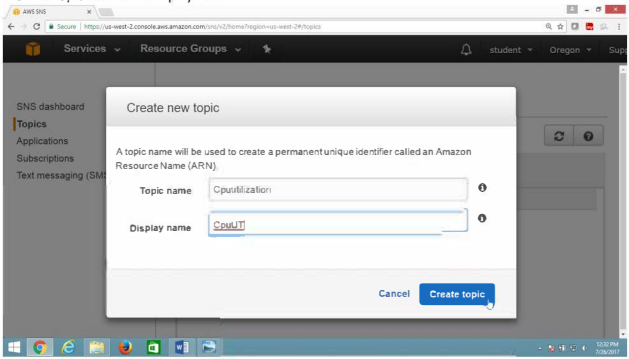
Select **Topic**

Click on "**Create new topic"** button

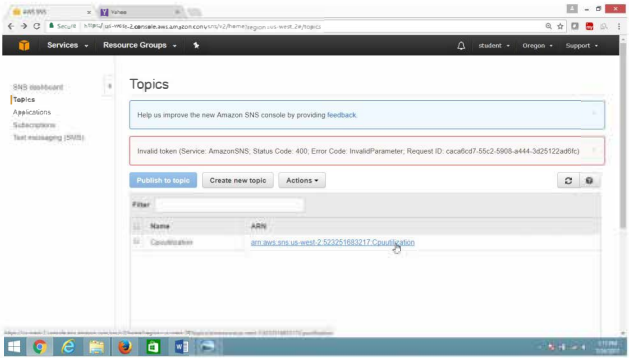


In **"Create new topic"** box

Provide Topic name and Display name

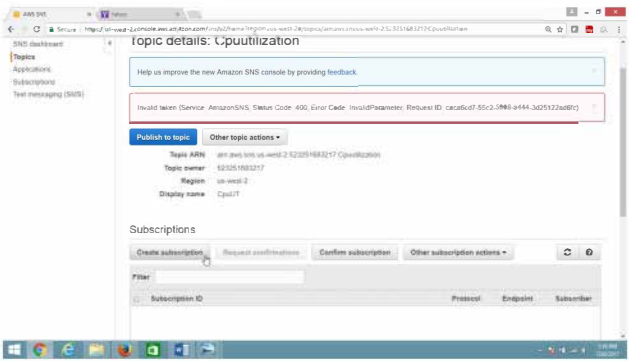


Click of ARN link



2) To create Subscription

Click on "**Createsubscription"** button

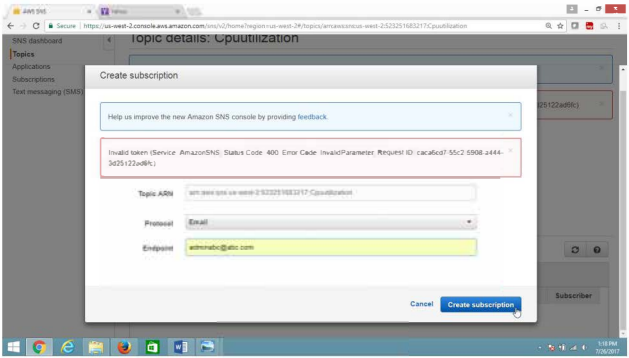


Provide values as

Protocol -→ **EMAIL**

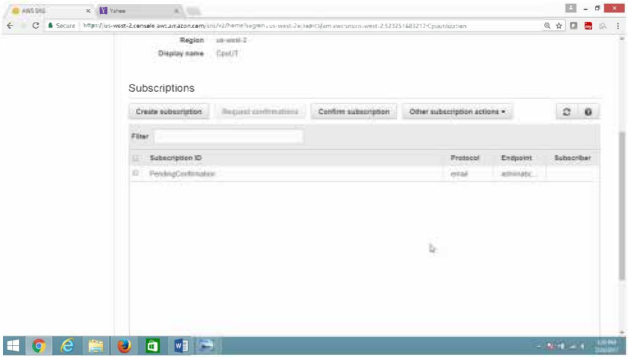
Endpoint -→adminaws@abc.com

Click **"Create subscription"** button



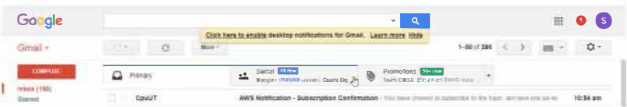
3) **Verification**

Now subscription is in pending state



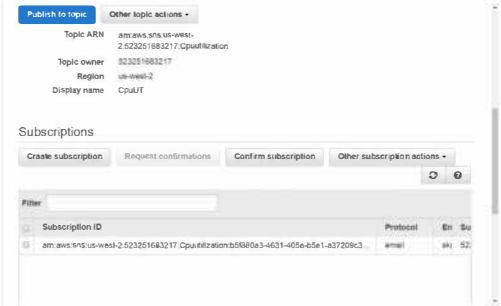
Go to your mail account

Click on the mail



Click on "Confirm message"

Now subscription is verified

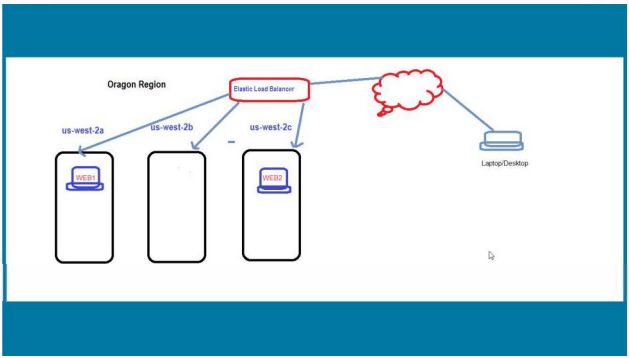


Lab 12: To Configure Amazon Elastic Load Balancer

OBJECTIVE

To configure Elastic load balancer in AWS

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK :

Launch two instance in two separate Availability Zone.

Configure httpd (Apache) webserver in each instances.

Verify Webserver from browser.

Configure Elastic Load Balancer.

Verify Webserver through ELB

1) Launch two install with apache webserver in two separate Availability Zone,

for example us-west-2a and us-west-2c

Note

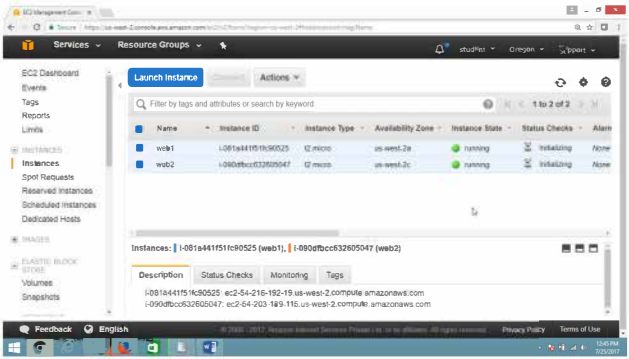
[To configure webserver refer lab – webserver configuration]

2) Check websites are running

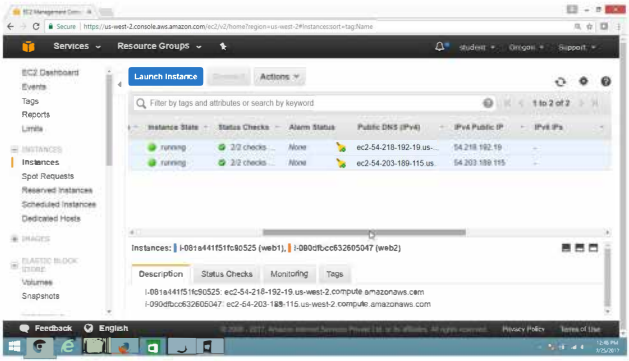
Open the browser

Provide public ip of both instances

Verify both website are running.



Verify Public IP of both instance



Verify

Output of Webserver one

Verify

Output of Webserver two

3) To Configure Elastic Load Balancer.

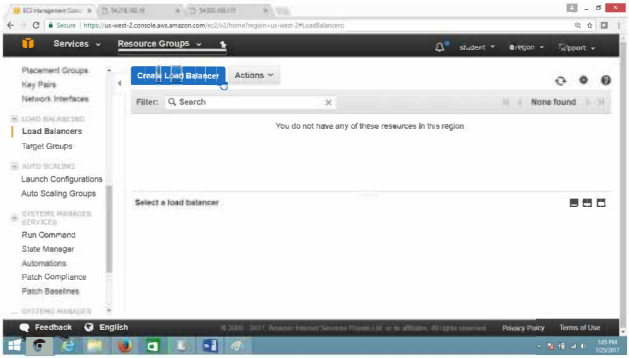
Open the AWS console

On **EC2 Dashboard** panel

Expanding **"LOAD BALANCING"**

Select **Load Balancer,**

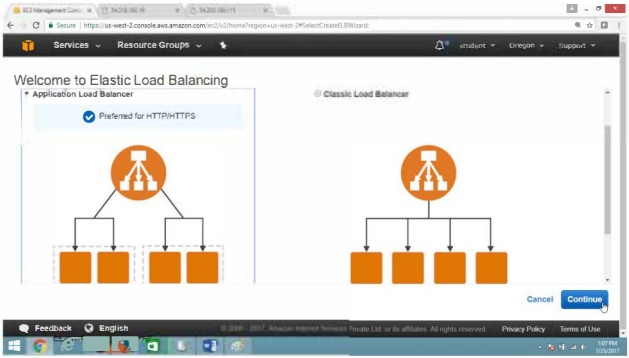
Click on **"Create Load Balancer"** button



On **"Welcome to Elastic Load Balancing"** page

Select "**Application Load Balancer",**

Click **Continue** button



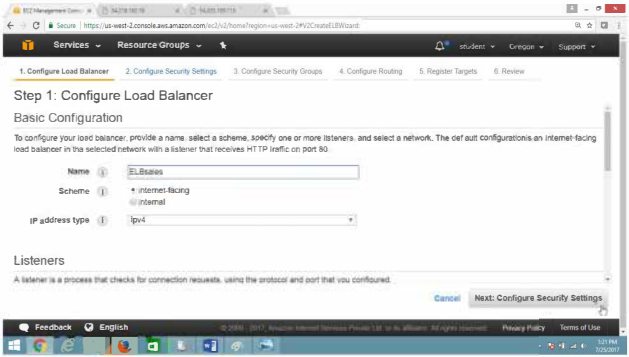
On "**Configure Load Balancer**" page

Provide

Name **→ELBsales**

Schema **→Internet-facing**

Drag down

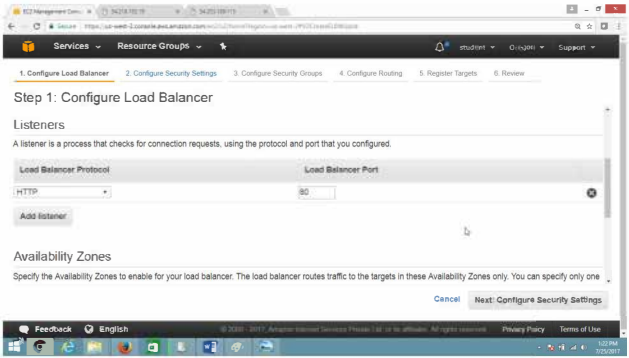


Under **Listeners,** Provide

Load Balancer Protocol →**HTTP**

Load Balancer Port as →**80**

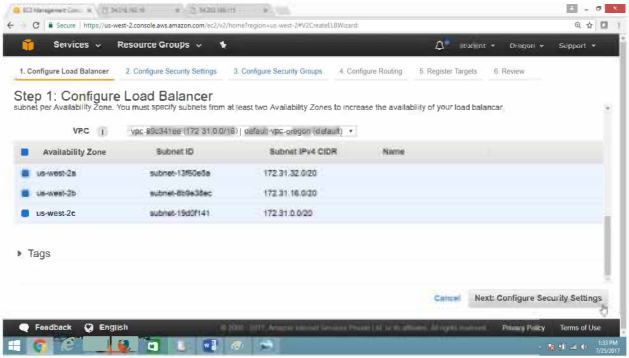
Drag down



Under **Availability Zones**

Select all zones

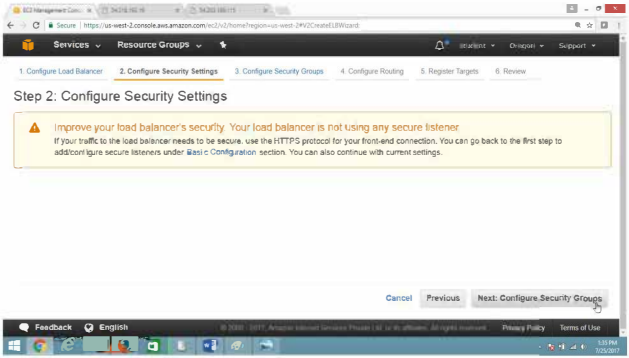
Click on "**Next:Configure Security Settings**" button



On **"Configure Security Settings"** page

Leave values as default.

Click "**Next:Configure Security Groups"** button

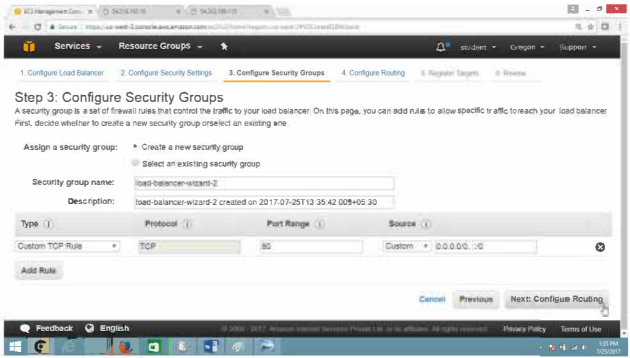


On "**Configure Security Groups**" page

Under Assign a security group

Select **"Create a new security group"**

click on **Configure Routing** button

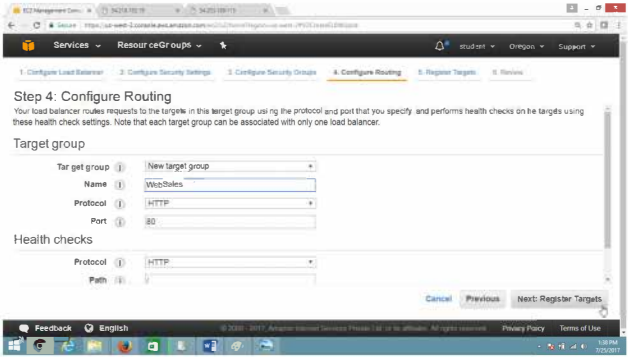


ON **"Configure Routing"** page give following values

Name → **W**ebsales

Leave remaining values as default

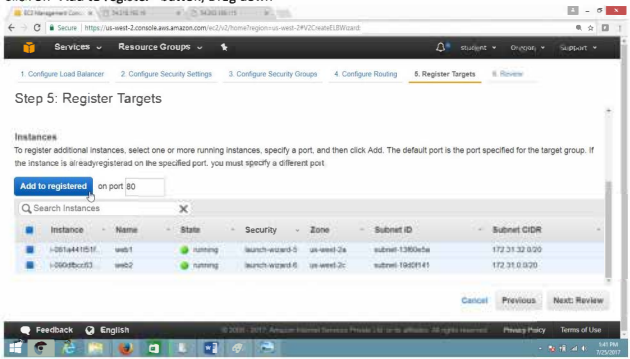
click "**Next: Register Targets"** button



On **Register Targets** page, Drag down

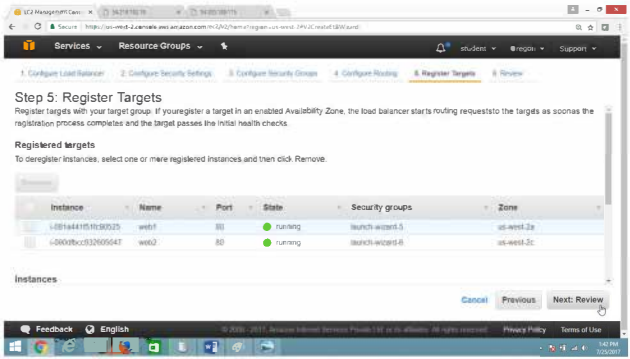
Select the instance which you want to put under load balancer,

Click on "**Add to register**" button, Drag down



Verify that running instances are registered

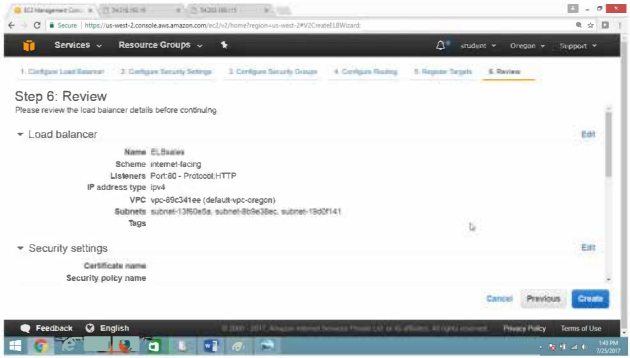
Click on "**Next: Review**" button



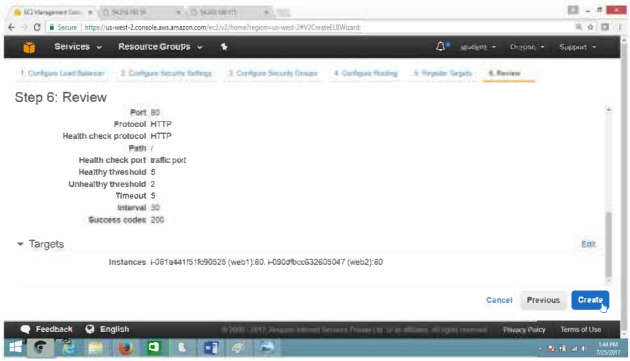
Verify

Check the summary

Drag Down

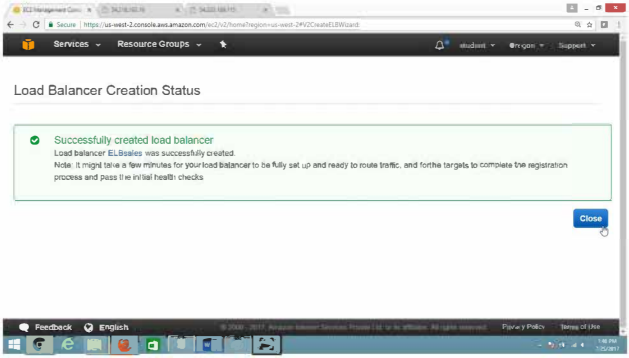


Click on **"Create"** button



Verify

Load balancer successfully created.



4) Verification

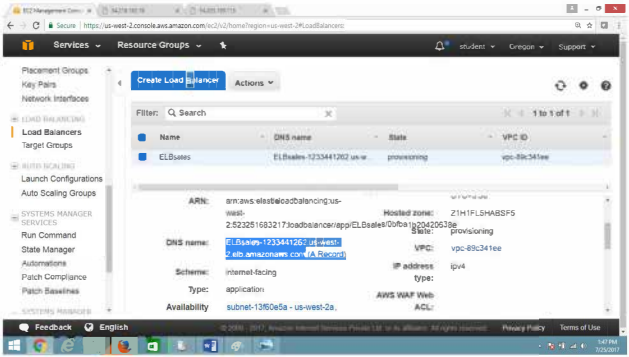
To verify **W**ebsites are coming through Load Balance**r**

Go to **EC2 Dashboard** panel

Expanding **LOAD BALANCING**

Select **Load Balancer.**

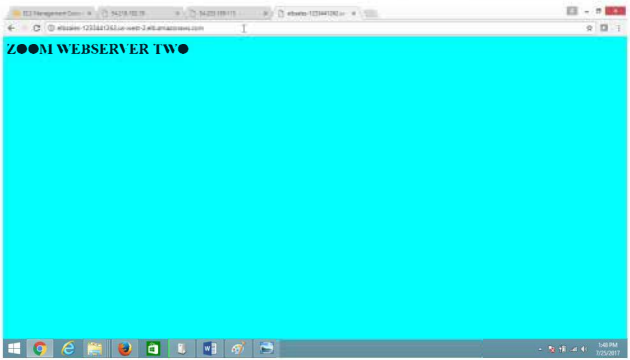
Copy Load Balance**r** DNS Name



In browser type load balancer DNS name

Verify website by frequently refreshing browser (press F5)

On Each Refresh one by one , Webserver 1 and Webserver 2 will be displayed.



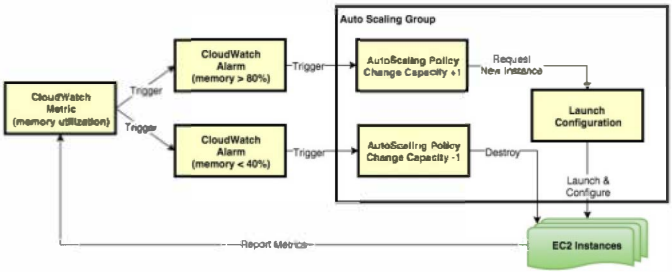
If you get this output, Congratulation your ELB configuration is successful.

Lab 13: To Configure Auto Scaling With Load Balancer

OBJECTIVE

To configure Auto Scaling in AWS

TOPOLOGY



PRE-REQUISTES

User should have AWS account, or IAM user with EC2fullaccess

TASK

Launch Amazon linux instance

Configure web server

Stop the instance

Create AMI image of above instance

Configure Autoscaling launch configuration and autoscaling group

Configure Load balancer with Autoscaling

Practical Steps

1) First launch Amazon linux Instance and configure webserver

2) Create AMI image

To create AMI from this instance

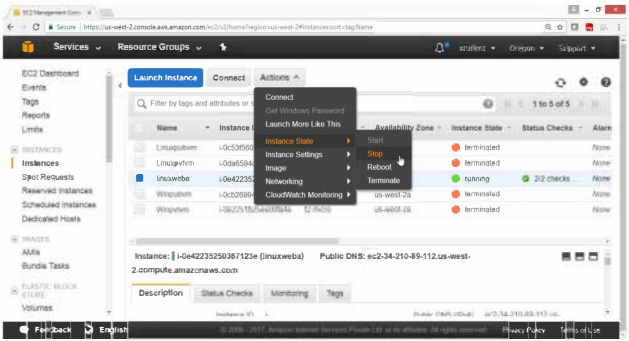
On **"EC2 Dashboard"** panel

Select the instance

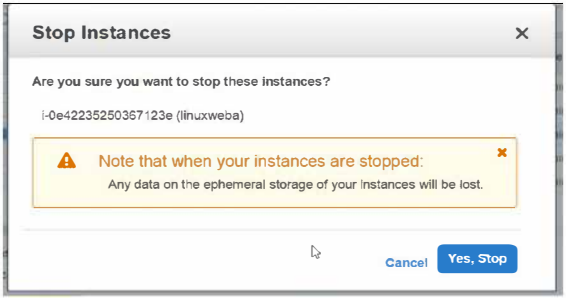
Click on **Action** button

Select Instance state

Click stop



Click on **Yes Stop** button

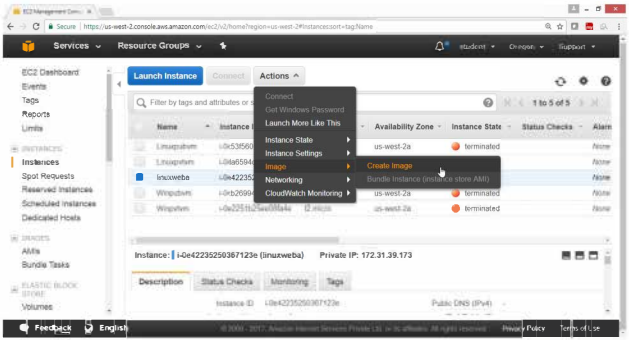


Select the stopped instance

Click on **Action** button

Select **image**

Click on **Create image** button

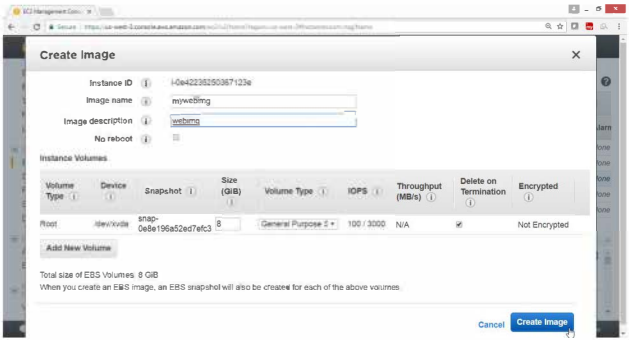


For Image name → mywebimg

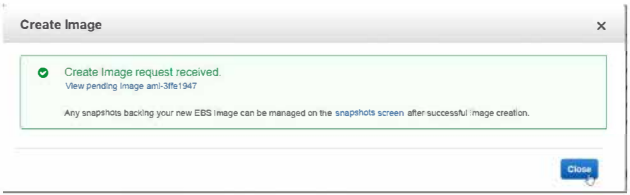
For Image description → webimg

Leave remaining default

Click on **Create image** button



Click on **Close** button



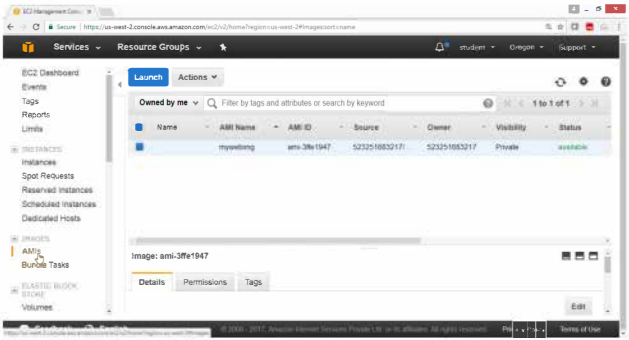
Verify AMI is created

On the **EC2 Dashboard** panel

Select **IMAGES**

Click on **AMIs**

Check the status is **available**



3) To Configure Auto Scaling

On the **EC2 Dashboard** panel

Select **"AUTO SCALING"**