



VPC's PEERING

IN DIFFERENT REGIONS

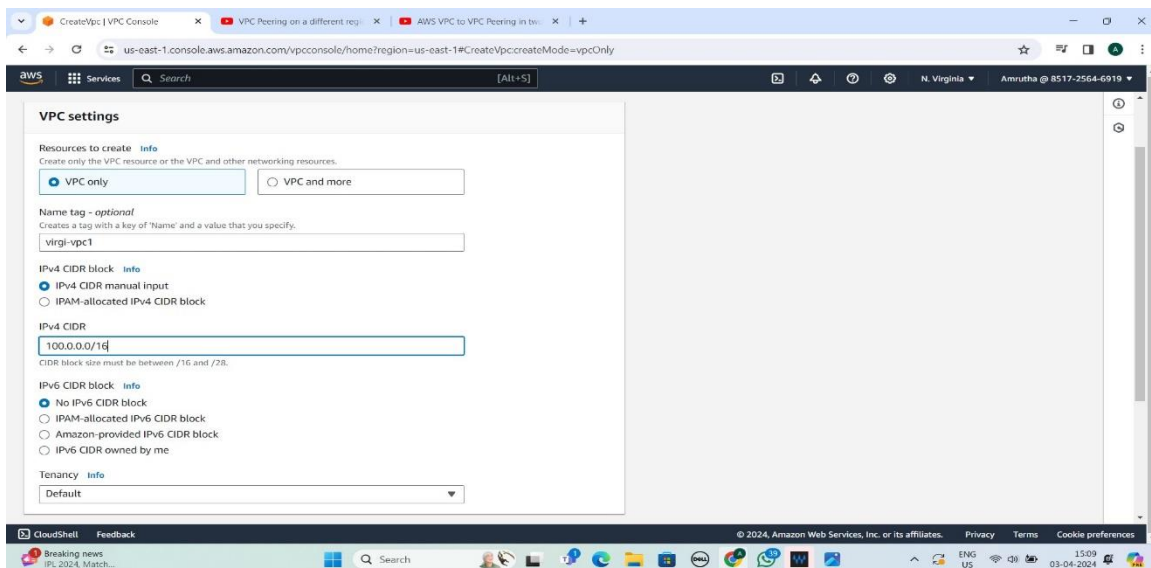
G.T.N.Amrutha

tejasri.cloud@gmail.com

Batch:122

Date:04/03/2023

Creation of two VPC'S in different regions and connecting the two vpc's [peering]



The screenshot displays the AWS Management Console interface. At the top, the breadcrumb navigation shows the path: **VPC** > **Subnets** > **subnet-0e530fd942b1309d8**. The main heading is **subnet-0e530fd942b1309d8 / virgi-subn1**. On the left sidebar, the **Virtual private cloud** section is expanded, showing **Subnets** as the active category. The main content area is titled **Details** and contains a table with the following information:

Details	
Subnet ID subnet-0e530fd942b1309d8	Subnet ARN arn:aws:ec2:us-east-1:851725646919:subnet/subnet-0e530fd942b1309d8
Available IPv4 addresses 251	State Available
Network border group us-east-1	Availability Zone us-east-1a
Default subnet No	Route table -
Customer-owned IPv4 pool -	Auto-assign IPv6 address No
IPv6-only No	IPv4 CIDR reservations -
DNS64 Disabled	Resource name DNS A record Disabled
	Auto-assign customer-owned IPv4 address No
	IPv6 CIDR reservations -
	Resource name DNS AAAA record Disabled

At the bottom of the console, there are tabs for **Flow logs**, **Route table**, **Network ACL**, **CIDR reservations**, **Sharing**, and **Tags**. The footer of the console shows the copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates.

Step3: created 'virgi-igw' internet gateway and attach the internet gateway to vpc

The image consists of two screenshots of the AWS Management Console interface, showing the steps to create and attach an internet gateway.

Top Screenshot: Create internet gateway

- Page Title:** Create internet gateway | VPC
- URL:** us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#CreateInternetGateway
- Section:** Create internet gateway
- Description:** An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.
- Form Fields:**
 - Internet gateway settings:**
 - Name tag:** Creates a tag with a key of 'Name' and a value that you specify. Value:
 - Tags - optional:** A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
 - Key:** Name
 - Value - optional:** virgi-igw
 - Buttons:** Add new tag, Remove
- Buttons:** Cancel, Create internet gateway

Bottom Screenshot: Attach to VPC (igw-00959a422008438ec)

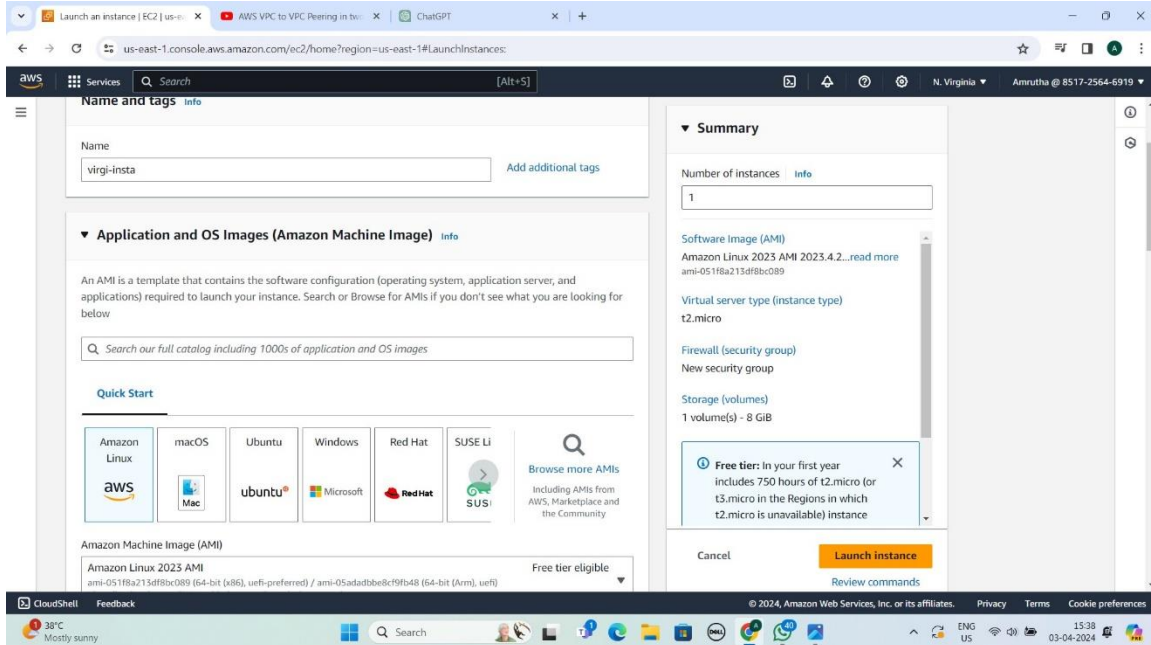
- Page Title:** Attach internet gateway | VPC
- URL:** us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#AttachInternetGateway:internetGatewayId=igw-00959a422008438ec
- Section:** Attach to VPC (igw-00959a422008438ec)
- Description:** Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.
- Form Fields:**
 - Available VPCs:** Attach the internet gateway to this VPC. Value:
 - Section:** AWS Command Line Interface command
- Buttons:** Cancel, Attach internet gateway

Step4: created 'virgi-rout1' route table and attached vpc and in the edit route section we attached the internet gateway we created earlier and associated the subnet

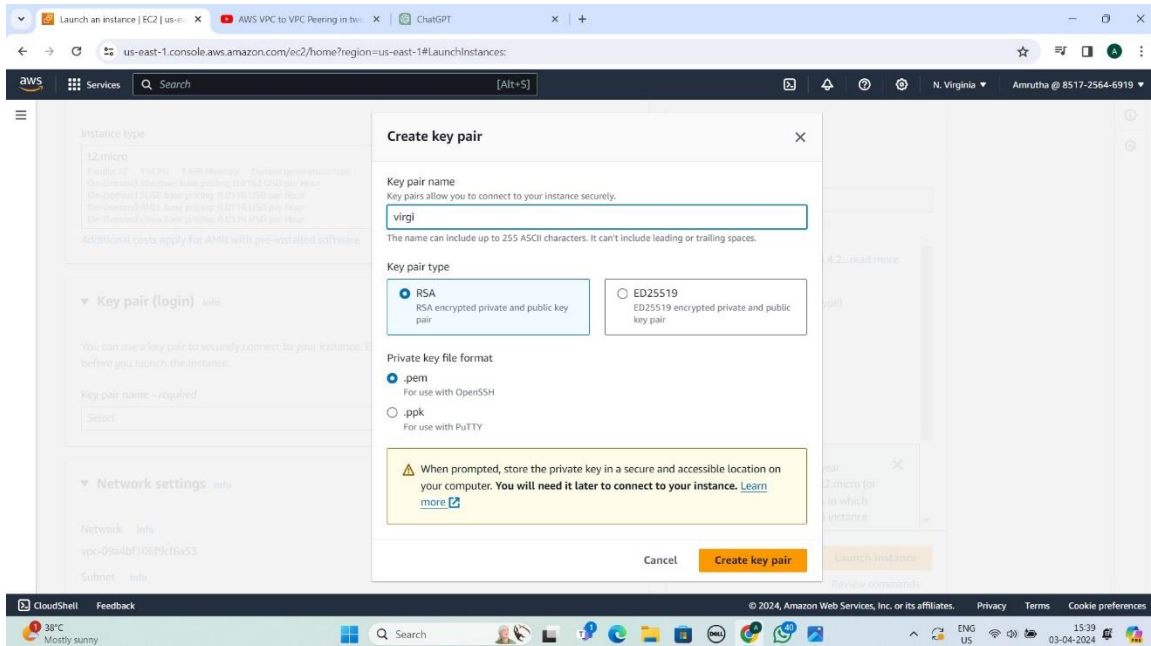
The first screenshot shows the 'Create route table' page in the AWS Management Console. The 'Route table settings' section has 'Name - optional' set to 'virgi-rout1' and 'VPC' set to 'vpc-07b7c8582a898f76c (virgi-vpc1)'. The 'Tags' section has a key 'Name' and value 'virgi-rout1'. The 'Create route table' button is visible at the bottom.

The second screenshot shows the 'Edit routes' page for the route table 'rtb-01d22a6b065d75fd5'. The 'Destination' column shows '100.0.0.0/16' and '0.0.0.0/0'. The 'Target' column shows 'local' and 'Internet Gateway' (with ID 'igw-00959a422008438ed'). The 'Status' column shows 'Active' and '-'. The 'Propagated' column shows 'No'. The 'Add route' button is visible at the bottom left, and 'Cancel', 'Preview', and 'Save changes' buttons are at the bottom right.

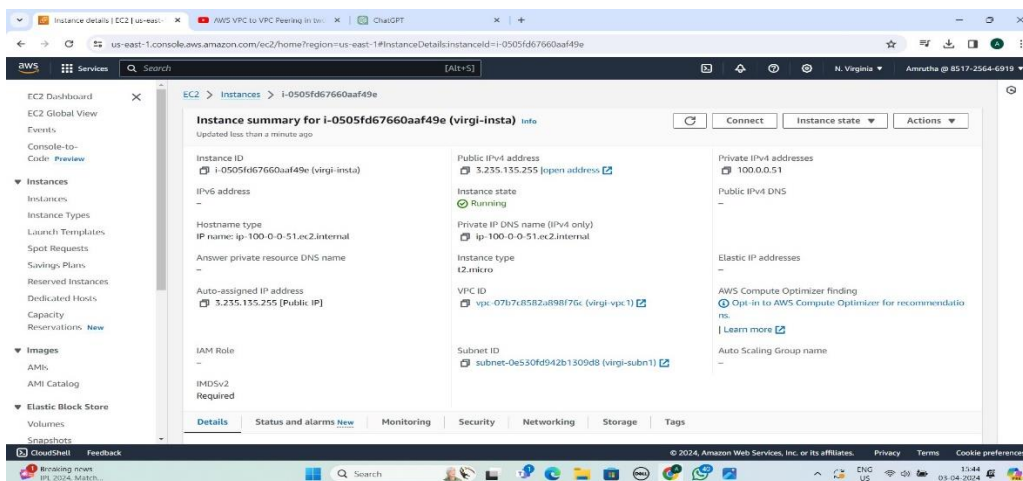
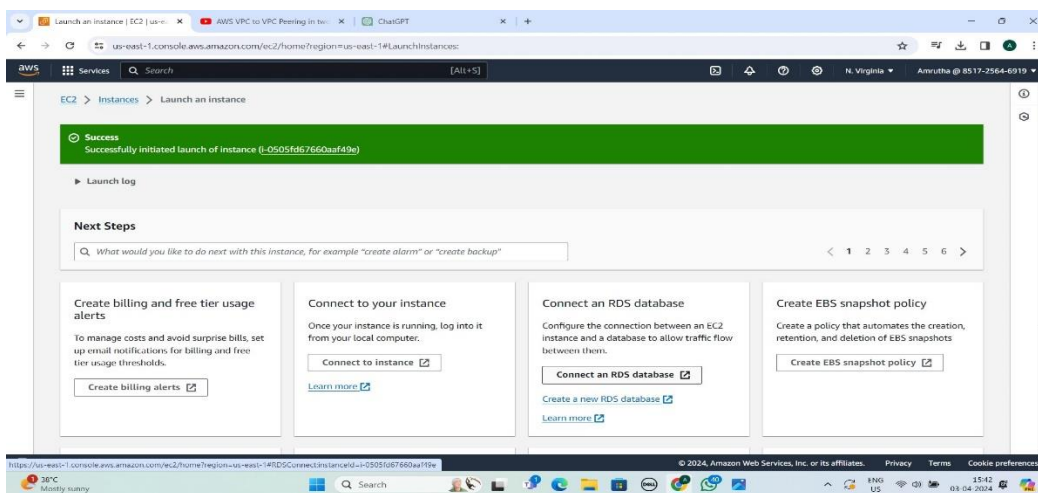
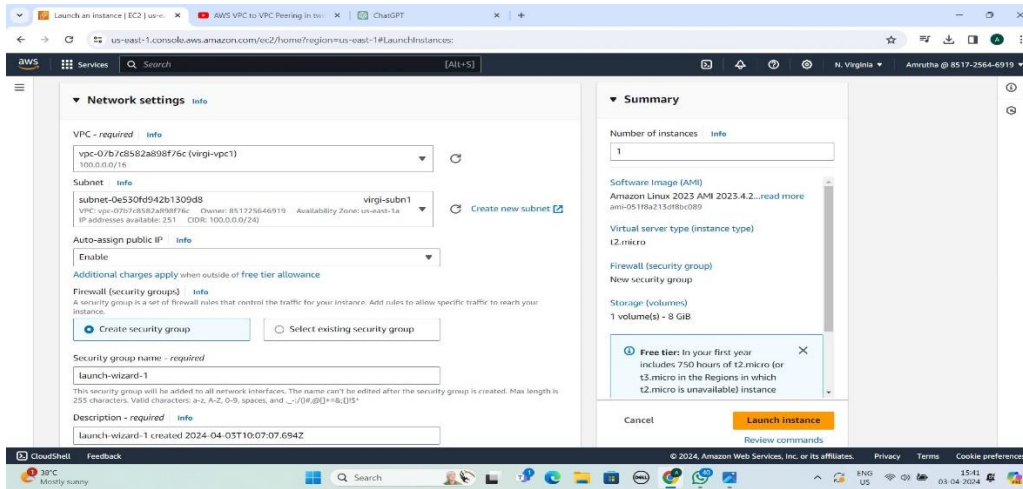
Step5: now create a EC2 instance here we created 'virgi-insta' using Amazon Linux



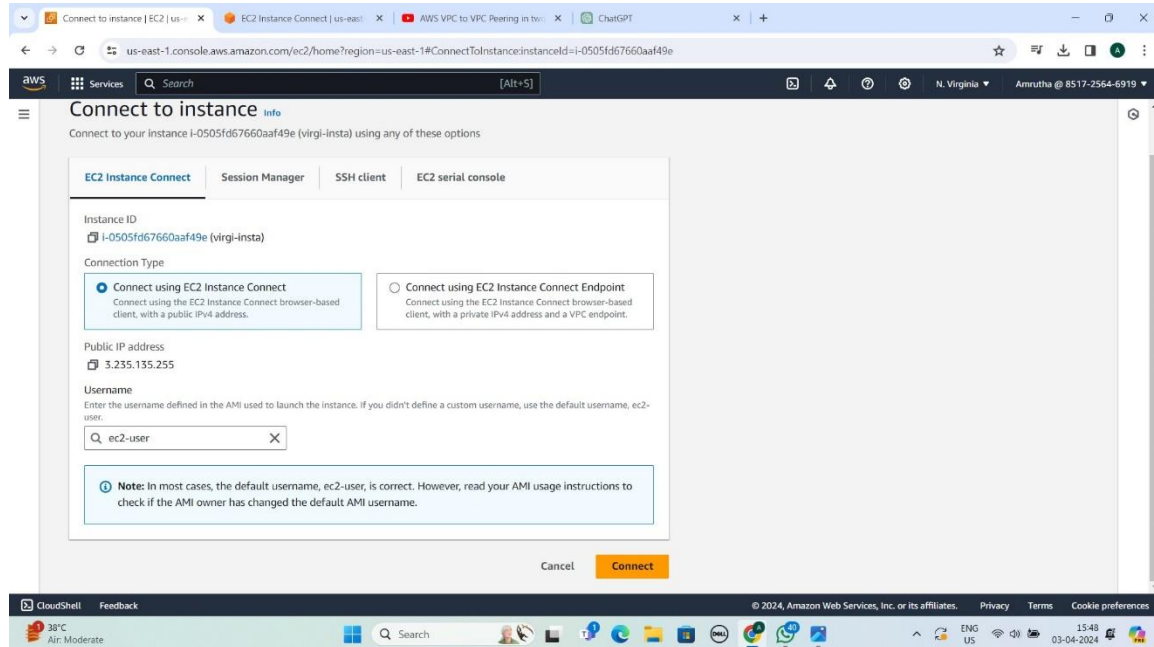
Step6: created 'virgi' keypair in .pem file formate



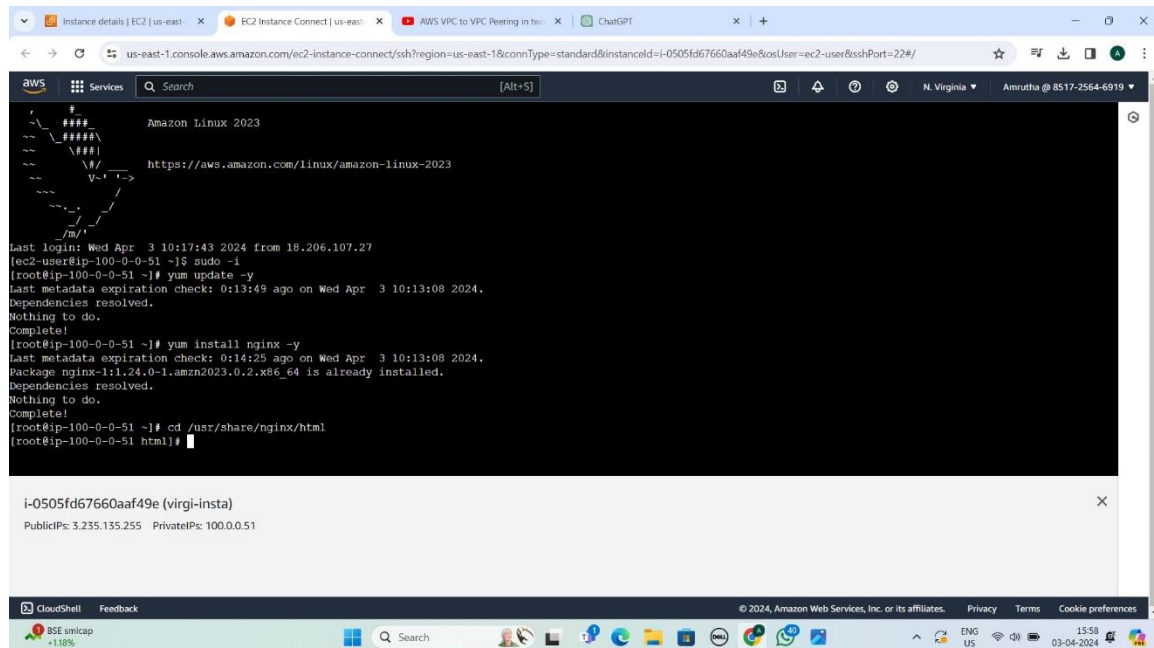
Step7:then edit network settings by selecting appropriate vpc and subnet then allow auto assign public ip then launch instance

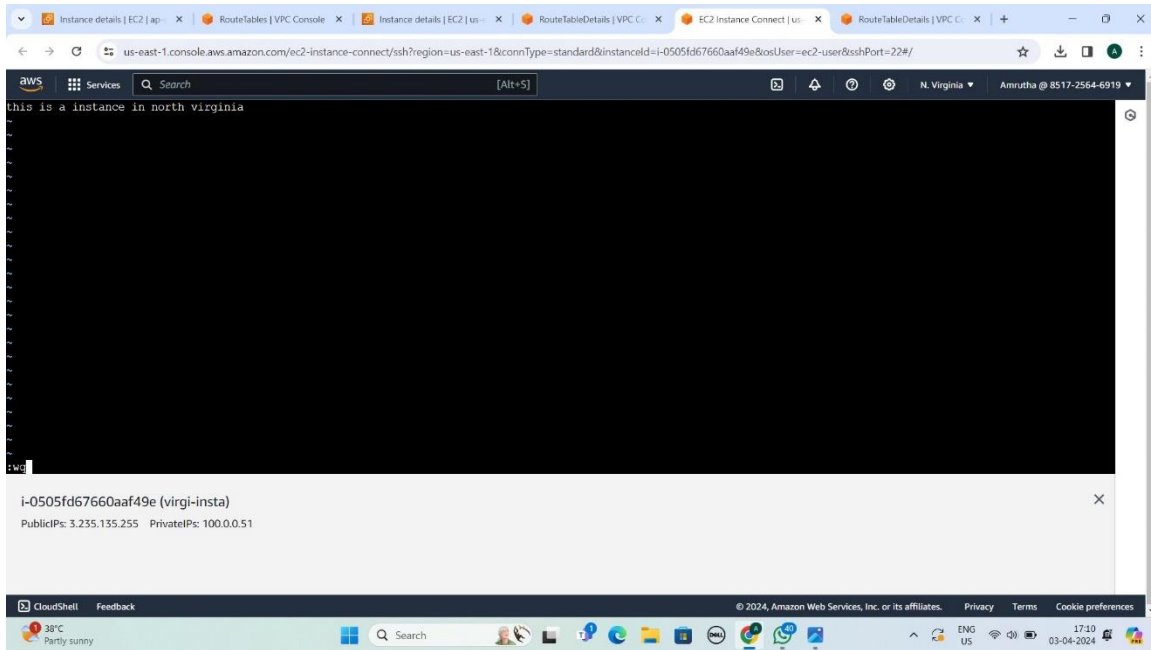


Step8:connect to the EC2 instance

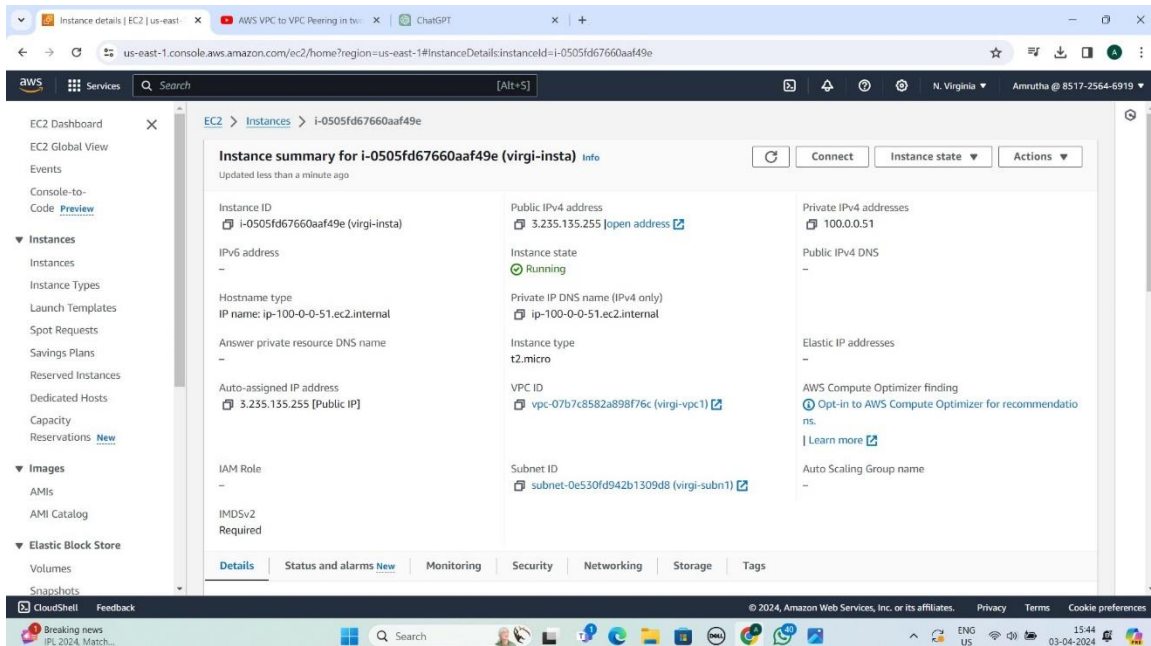


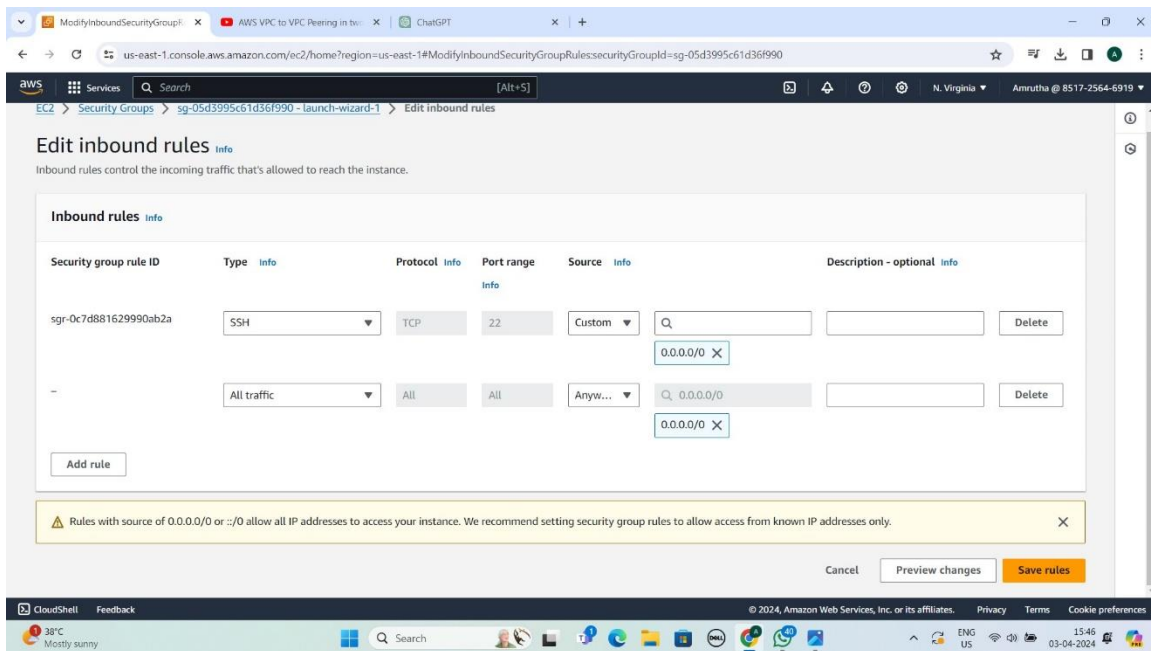
Step9: then update and install nginx and create a file and insert the content and save it



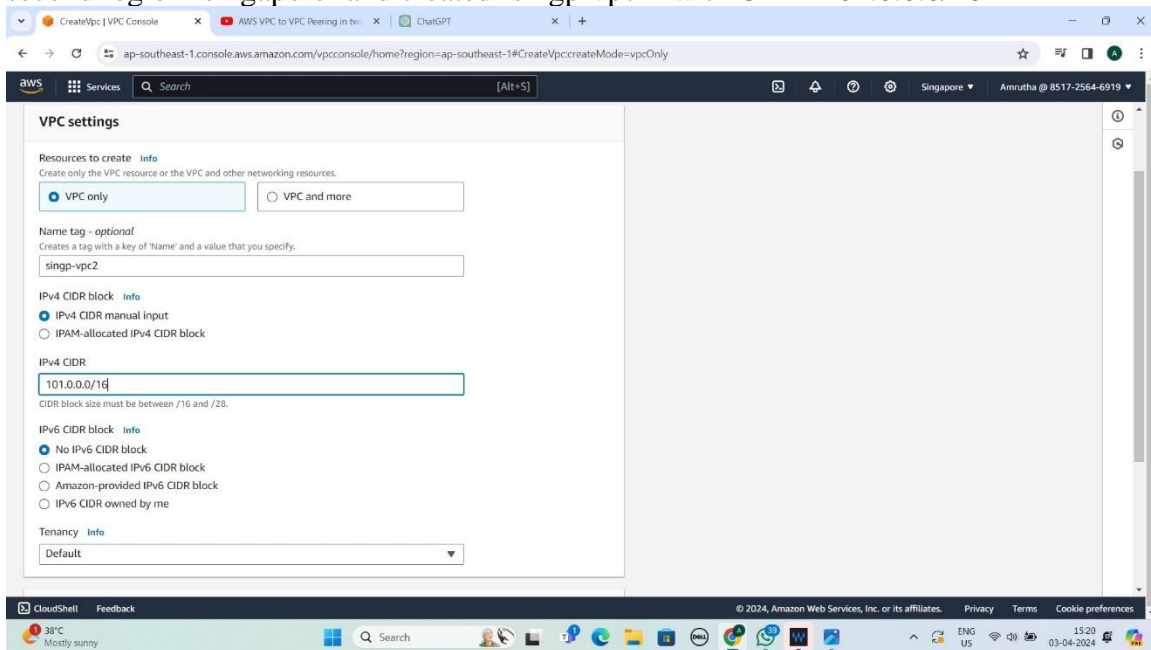


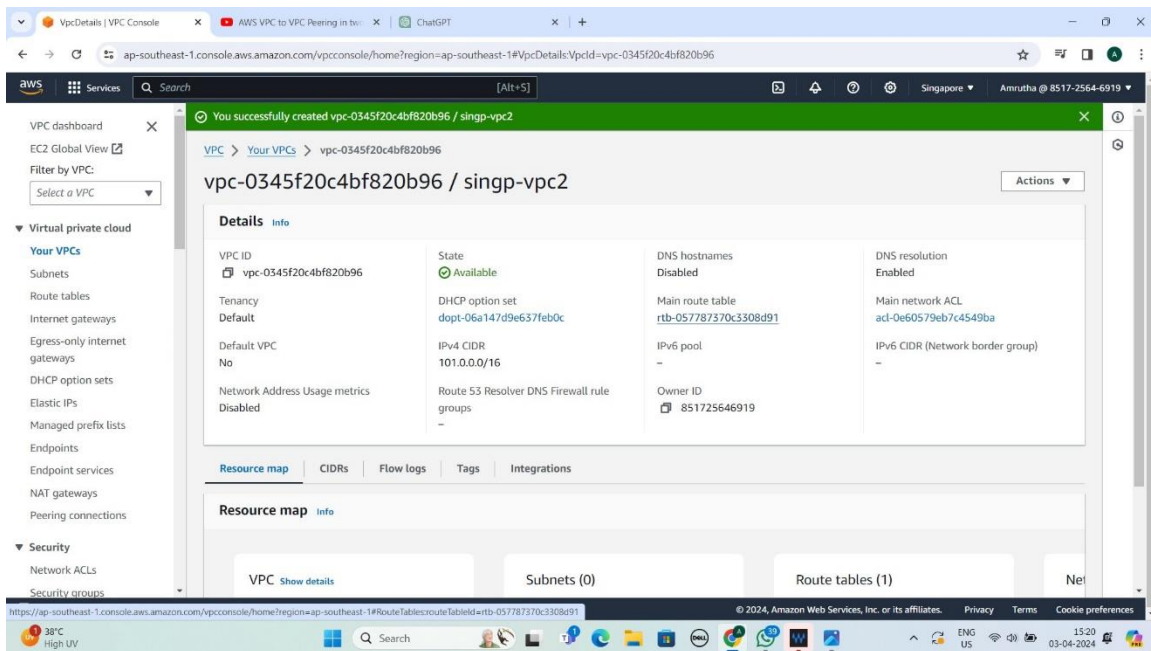
Step10: Then in the Security section click on the security groups and edit the in-bound rules by adding the all traffic type and save rules



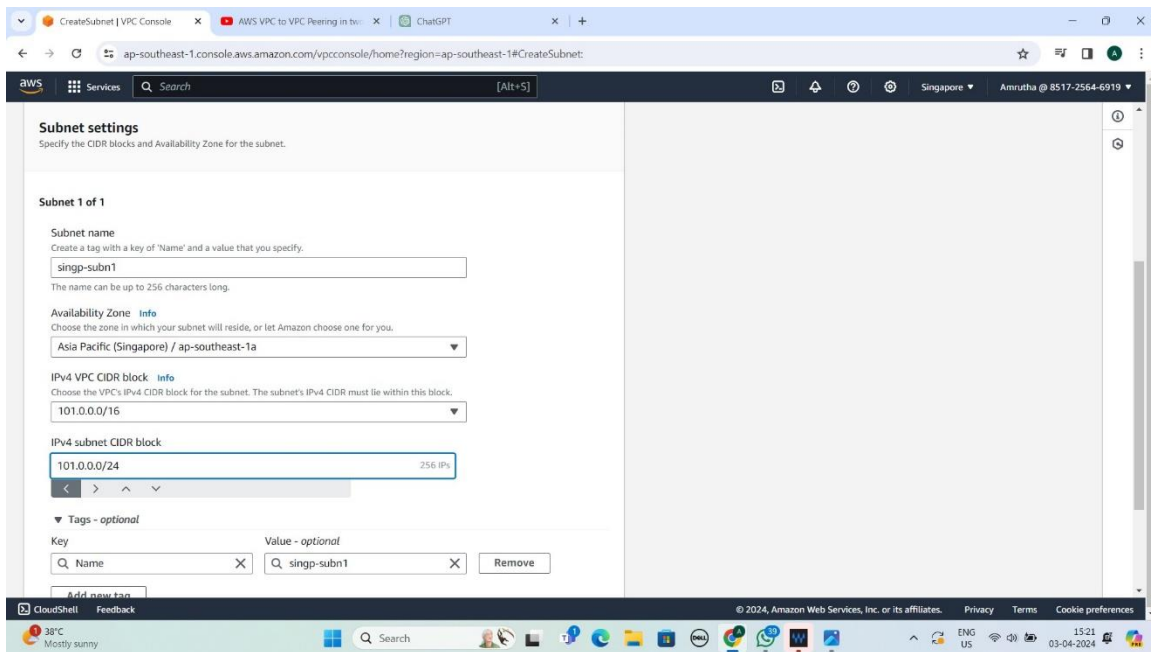


Step11: In the similar way we create another vpc in another region here we take our second region 'singapore' and created 'singp-vpc2' with CIDR 101.0.0.0/16

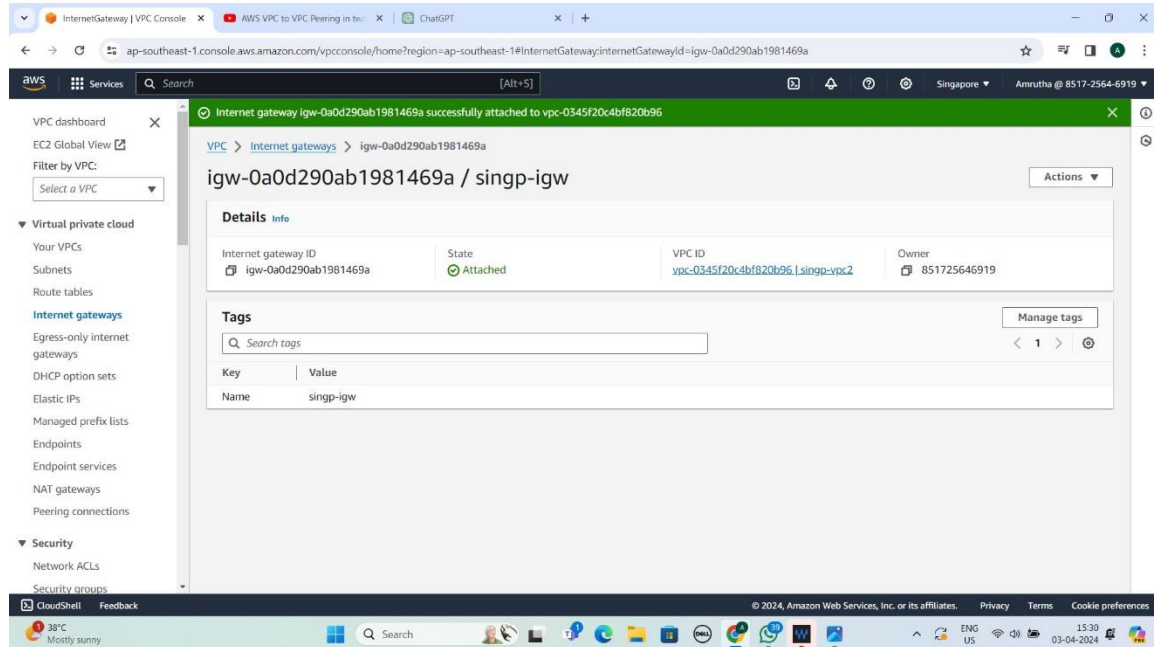




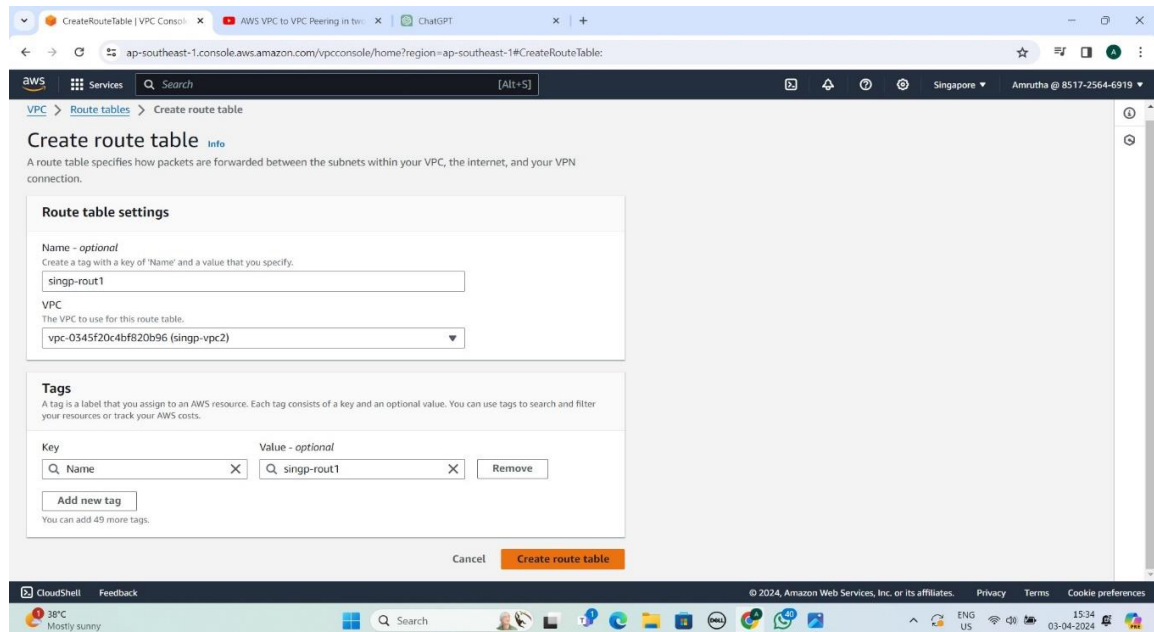
Step12: created 'singp-subn1' with CIDR 101.0.0.0/24 in 'ap-southeast-1a'

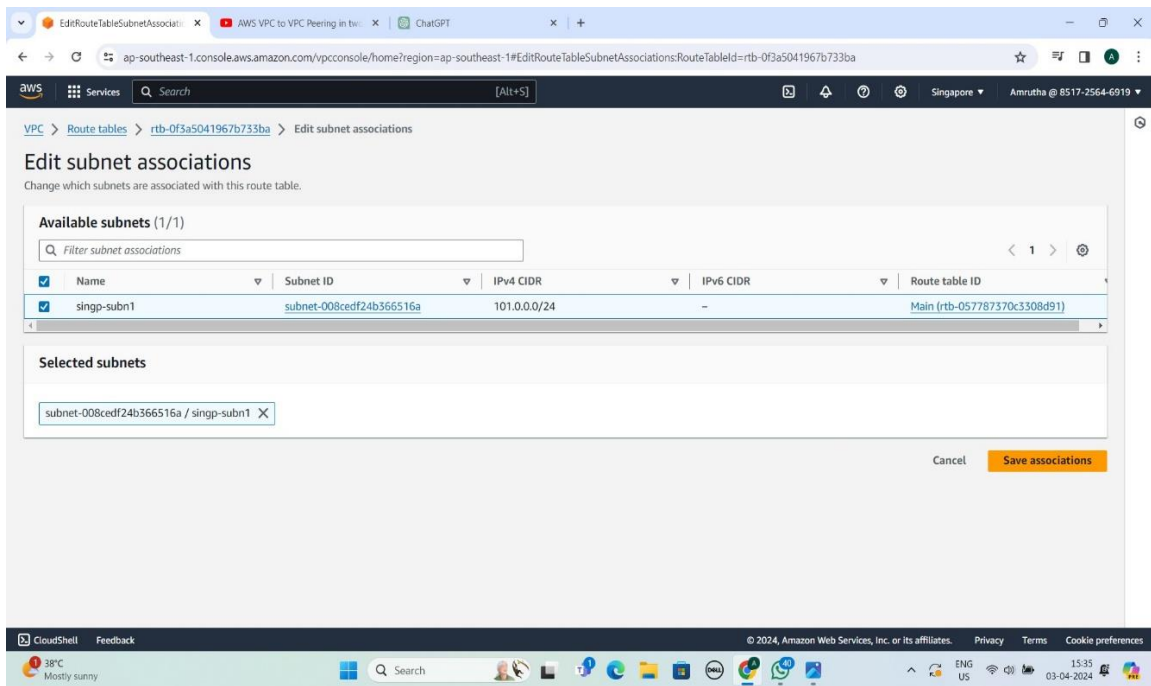


Step13:created 'singp-igw' internet gateway and attached the vpc

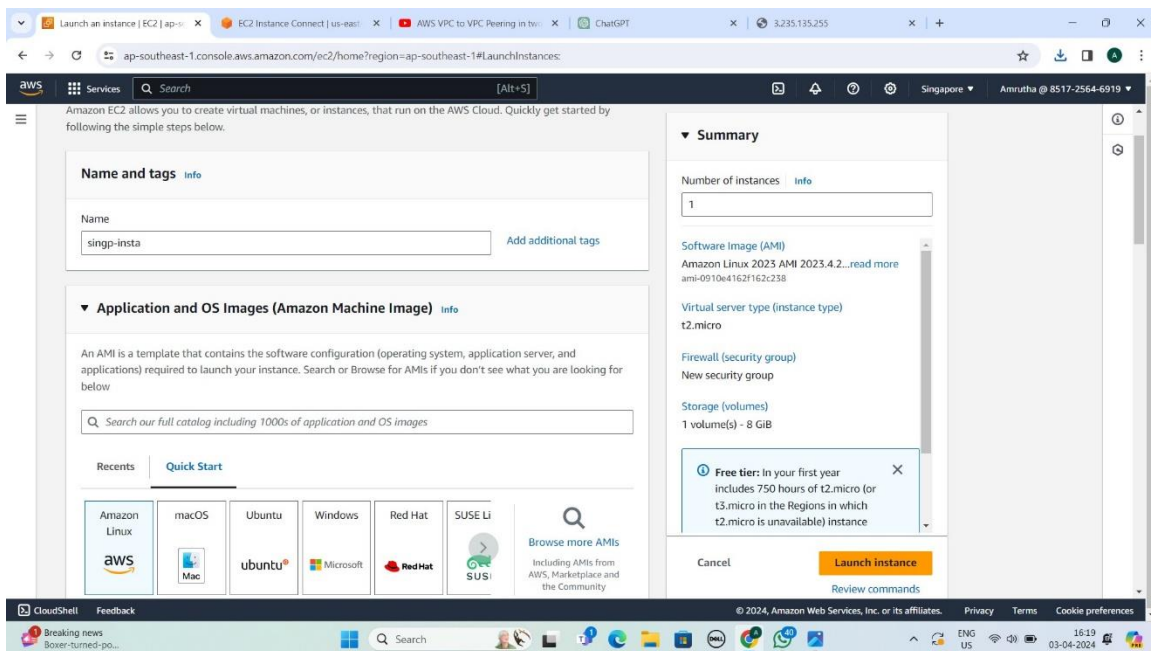


Step14:created 'singp-route1' and attached the internet gateway and associated the subnet





Step15: now create a EC2 instance here we created 'singp-insta' using Amazon Linux



The screenshot displays the AWS Management Console's 'Launch instances' wizard. The 'Network settings' tab is active, showing the following configurations:

- VPC:** vpc-0345f20c4bf820b96 (singp-vpc2) with CIDR 101.0.0.0/16.
- Subnet:** subnet-008cedf24b366516a (singp-subn1) with CIDR 101.0.0.0/24.
- Auto-assign public IP:** Enabled.
- Firewall (security groups):** A new security group is being created.

The 'Summary' panel on the right provides a overview of the instance configuration:

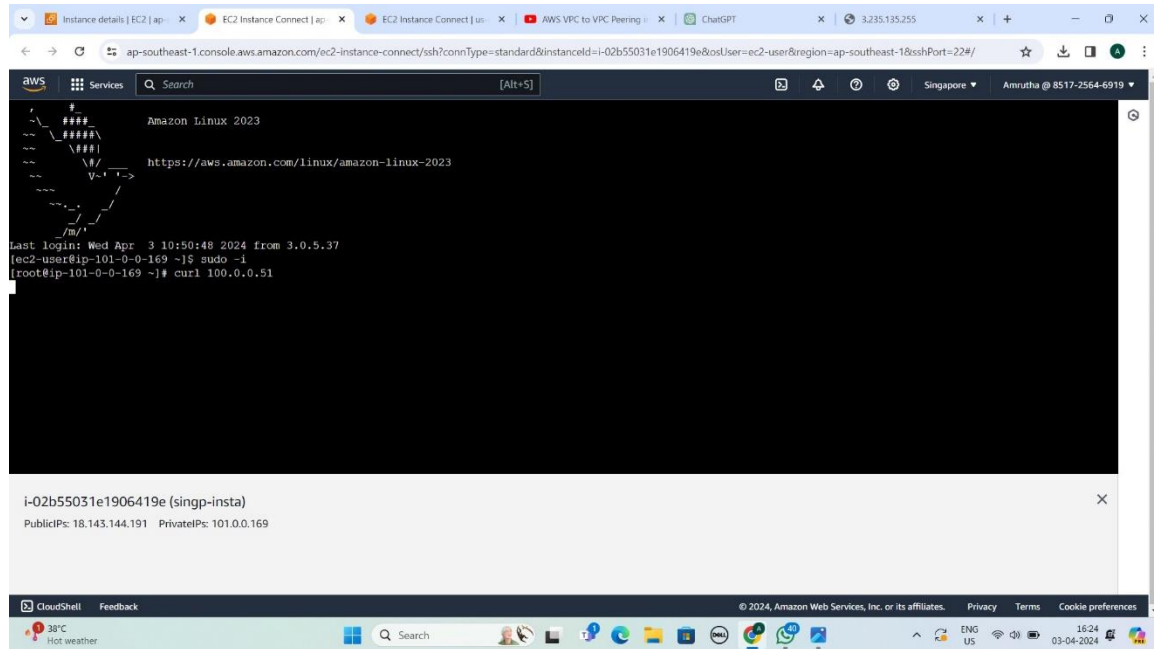
- Number of instances:** 1
- Software Image (AMI):** Amazon Linux 2023 AMI 2023.4.2...
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** New security group
- Storage (volumes):** 1 volume(s) - 8 GiB

A 'Free tier' notification is shown, stating: 'Free tier: In your first year includes 750 hours of t2.micro for t3.micro in the Regions in which t2.micro is unavailable instance'.

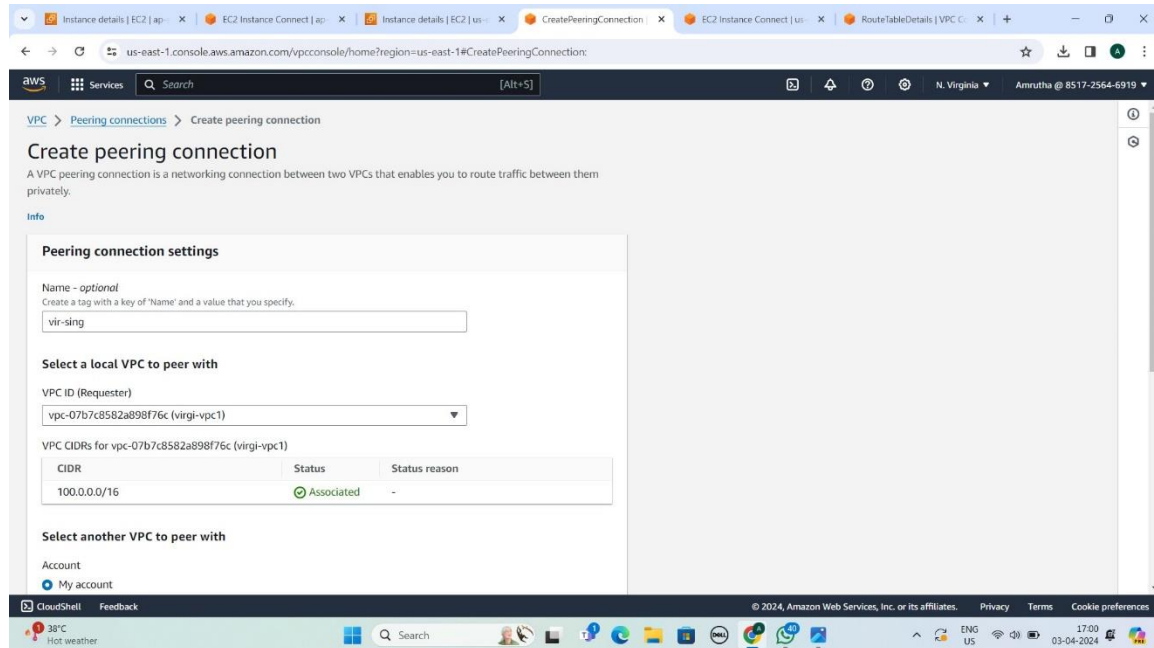
The 'Launch instance' button is prominently displayed in orange at the bottom right of the summary panel.

The screenshot displays the AWS Management Console interface for an EC2 instance. The top navigation bar includes the AWS logo, a search bar, and a navigation menu with options like CloudShell, Feedback, and various AWS services. The main content area shows the 'Instance summary' for the instance 'i-02b55031e1906419e' (singp-insta). The instance is in the 'Running' state. The summary includes details such as the Public IP4 address (18.143.144.191), Private IP4 address (101.0.0.169), Instance state (Running), Private IP DNS name (ip-101-0-0-169.ap-southeast-1.compute.internal), Instance type (t2.micro), VPC ID (vpc-0345f20c4bf820b96), Subnet ID (subnet-008cedf24b366516a), and IAM Role (Required). The console also shows a left-hand navigation menu with options like EC2 Dashboard, Instances, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, and Lifecycle Manager. The top of the console shows the AWS logo, search bar, and navigation tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags.

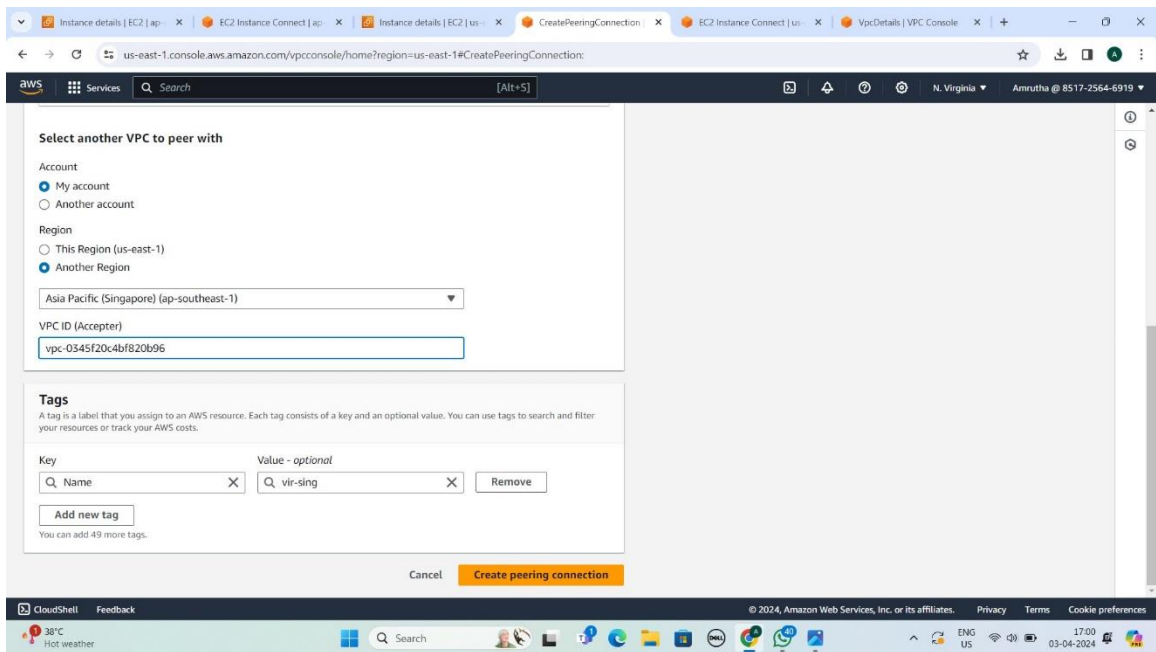
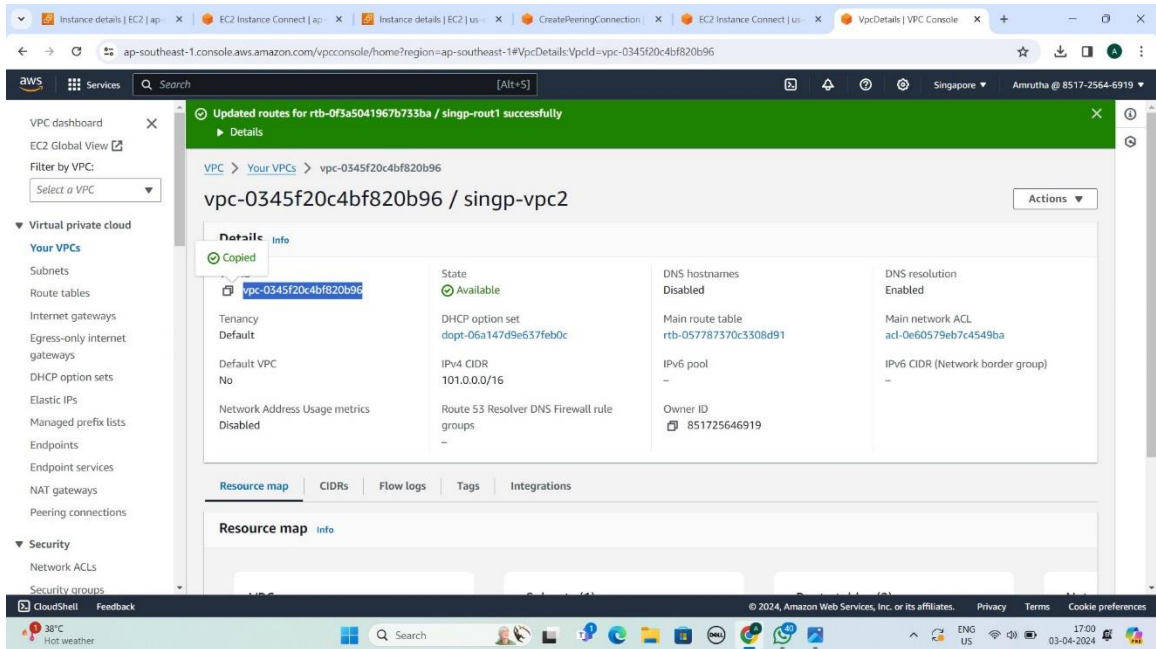
Step17:connect to the instance and try to access the private ip in singapore region it gives a failure as there is no connection between them



Step18:create a peering connection in virgina here we created ‘vir-sing’ and attach local vpc



Step19: now copy the vpc id of the acceptor [another region vpc] and past it in the virgina peering creation field and create peering



Step20: accept the peering request in the Singapore region and by clicking the edit route we link the virgina to singapore

Peering connections (1/6) info

Name	Peering connection ID	Status	Requester VPC	Accepter VPC
vir-sing	pcx-07dd2c962139106f7	Pending acceptance	vpc-07b7c8582a898f76c	vpc-0345f20c4bf820b96 / sing...
-	pcx-073929b92c1806028	Active	vpc-0f2cc799df031e0e2	vpc-002282e7c7c4062f0 / my...
-	pcx-0442979429f4f8e1e	Deleted	vpc-07b7c8582a898f76c	vpc-0345f20c4bf820b96 / sing...
singp-virgi	pcx-0b4eead3abcbe5d2	Deleted	vpc-0345f20c4bf820b96 / sing...	vpc-07b7c8582a898f76c
peering-01	pcx-0061db93f83b0c169	Deleted	vpc-002282e7c7c4062f0 / my...	vpc-0f2cc799df031e0e2
vir-singa	pcx-0c32ba67f4995c8bc	Deleted	vpc-07b7c8582a898f76c	vpc-0345f20c4bf820b96 / sing...

pcx-07dd2c962139106f7

Pending acceptance
You can accept or reject this peering connection request using the 'Actions' menu. You have until Wednesday, April 10, 2024 at 17:01:03 GMT+5:30 to accept or reject the request, otherwise it expires.

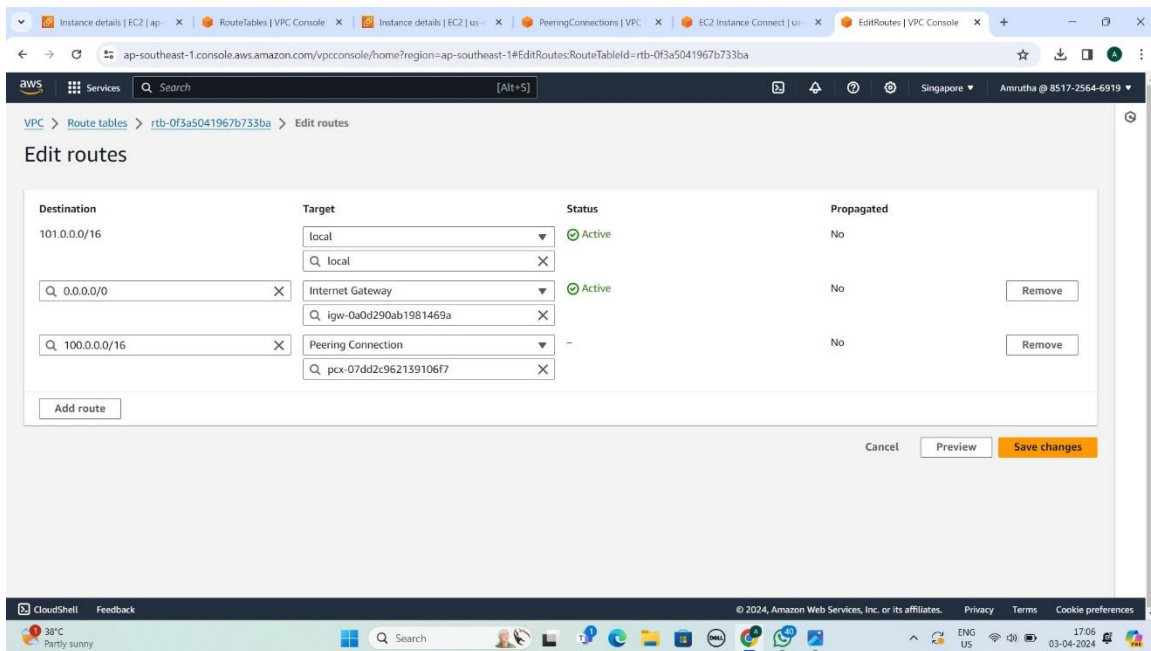
Details

Requester owner ID:
Accepter owner ID:
VPC Peering connection ARN:

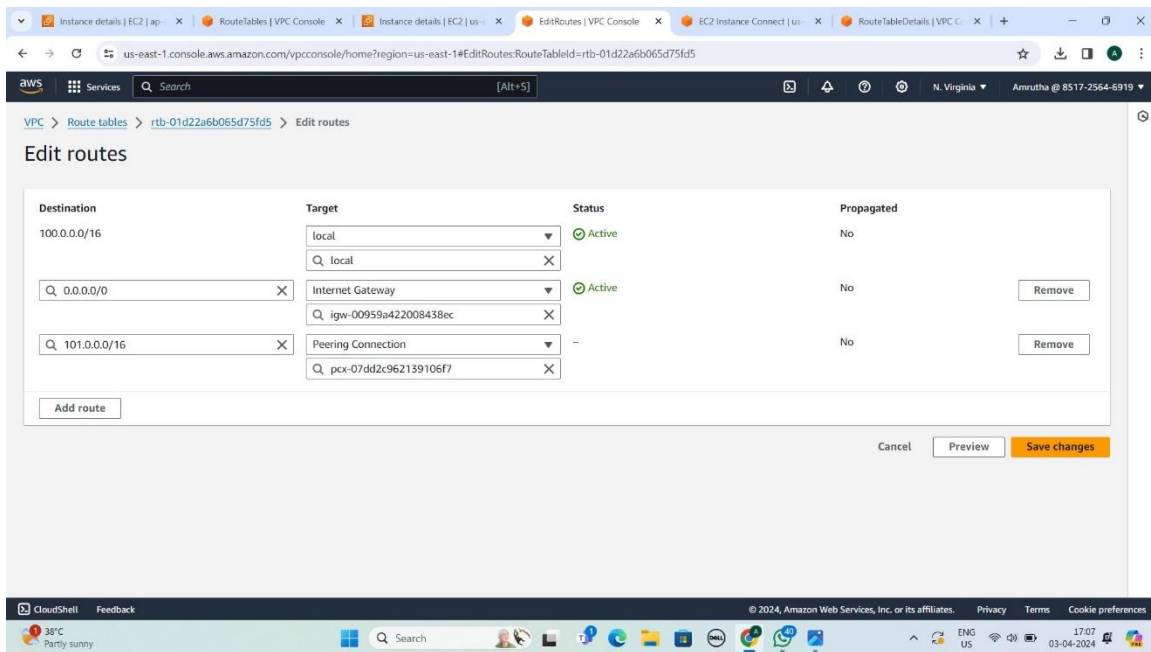
Route tables (5) info

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
singp-rout1	rtb-0f3a5041967b733ba	subnet-008cedf24b3665...	-	No	vpc-0345f20c4bf820b96
-	rtb-057787370c3308d91	-	-	Yes	vpc-0345f20c4bf820b96
my-rt2	rtb-0036a996b811bf241	subnet-0e2142e10d91ef...	-	No	vpc-002282e7c7c4062f0
-	rtb-041b697027b6eac81	-	-	Yes	vpc-0e9e93220c0218ba4
-	rtb-09f9b8f3e48f3837e	-	-	Yes	vpc-002282e7c7c4062f0

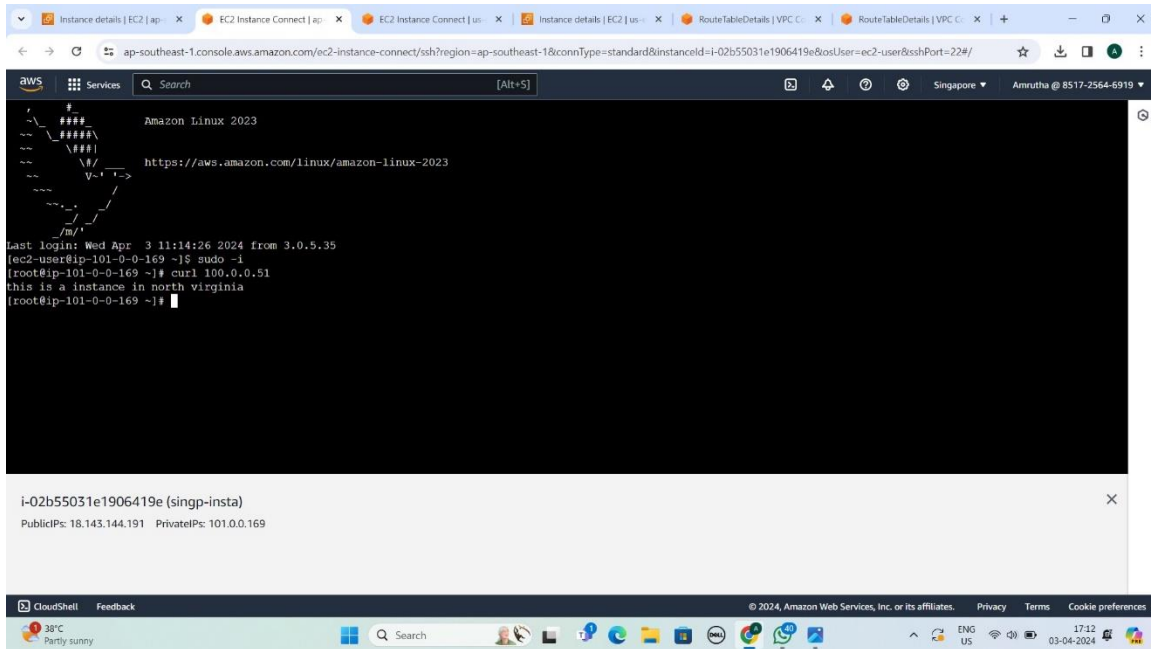
Select a route table



Step21: in the similar way edit the route in virgina and connect Singapore



Step22: them connect to the EC2 instance in Singapore and access virgina vpc through it by using the private id of virgina vpc



Conclusion: Here we successfully connect two vpc's in different region through vpc peering and access the content in them.