

# HACKATHON-2K24



**Team  
CONQUERORS**



**Project Title :**

**Multi Region Bucket Accessing Using  
Peering Connections**

**Submitted by :**

- A. Kalyan - 21W91A0402
- I. Rachana – 21W91A0441
- A. Ashok Reddy – 21W91A0406
- D. Sai Venkata Sai – 21W91A0426
- E. John Mosses – 22W95A0405

**Under the Esteemed Guidance of**

**Mr. AASHU DEV**

- AWS Academy Accreditation
  - AWS Academy Educator
  - AWS Solution Architect 2X Certified
  - AWS Re-Start Accreditation
- TSS – Tech Sign Solutions**

---

## ***ABSTRACT***

---

This project presents a robust and efficient method for accessing S3 buckets across multiple regions using Amazon Web Services (AWS). The architecture leverages key AWS services such as Multi-Region Access Points, AWS PrivateLink, VPC Peering, and S3 Cross-Region Replication (CRR). The Multi-Region Access Points route Amazon S3 data request traffic from multiple sources without the need for complex networking configurations. AWS PrivateLink provides a private connection for routing S3 requests into AWS or across multiple AWS Regions. VPC Peering is used to connect separate VPCs in different regions. S3 CRR is used to synchronize data among buckets in those Regions. The proposed architecture allows for the building of multi-region applications with the same architecture that's used in a single Region, and then run those applications anywhere in the world. This setup provides built-in network resilience with acceleration of internet-based requests to Amazon S3, thereby improving data availability and redundancy.

---

## INTRODUCTION

---

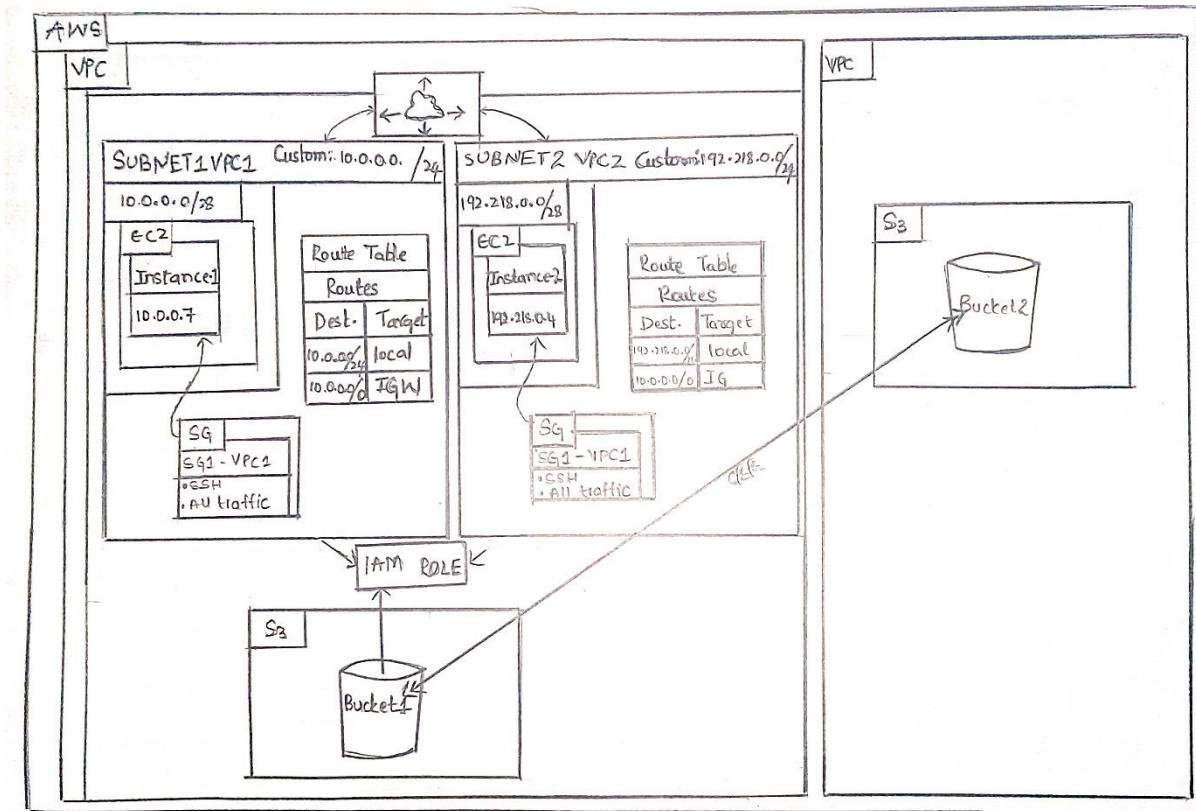
In the era of cloud computing, data accessibility and redundancy are of paramount importance. This project, titled “Multi-Region Bucket Accessing using Peering Connections in AWS”, aims to address these needs by leveraging the power and flexibility of Amazon Web Services (AWS).

The project focuses on the implementation of a robust system that allows for efficient access to S3 buckets across multiple regions. This is achieved through the use of several key AWS services, including Multi-Region Access Points, AWS PrivateLink, VPC Peering, and S3 Cross-Region Replication (CRR).

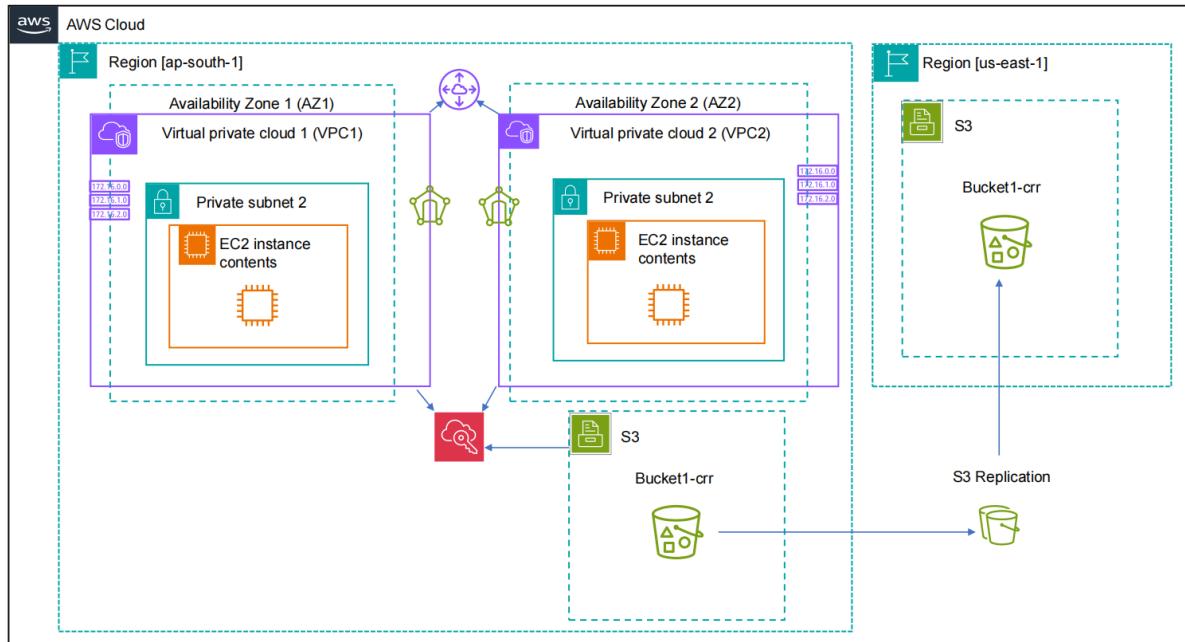
The architecture designed in this project not only ensures high availability and redundancy of data but also provides a resilient network with accelerated internet-based requests to Amazon S3. This system allows for the development and deployment of multi-region applications with the same architecture used in a single region, thereby offering a scalable and efficient solution for global data accessibility.

Stay tuned as we delve deeper into the intricacies of this project, exploring each component in detail and understanding how they all come together to form a comprehensive solution for multi-region data access in AWS.

# Rough Architecture of Aws

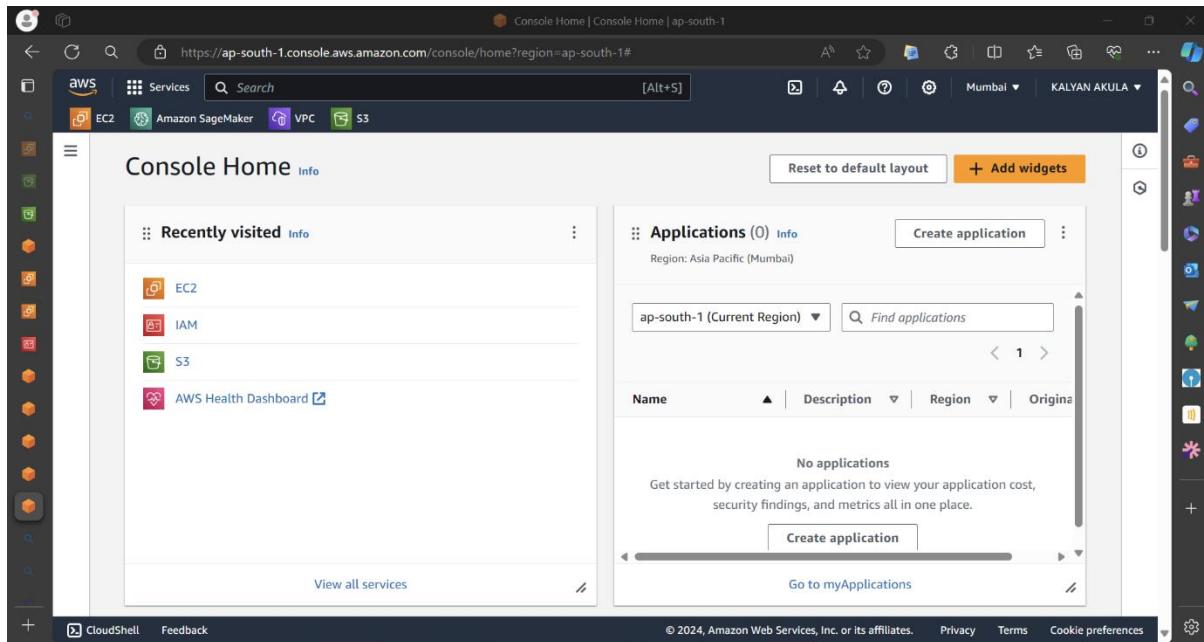


# Real Time Architecture of AWS

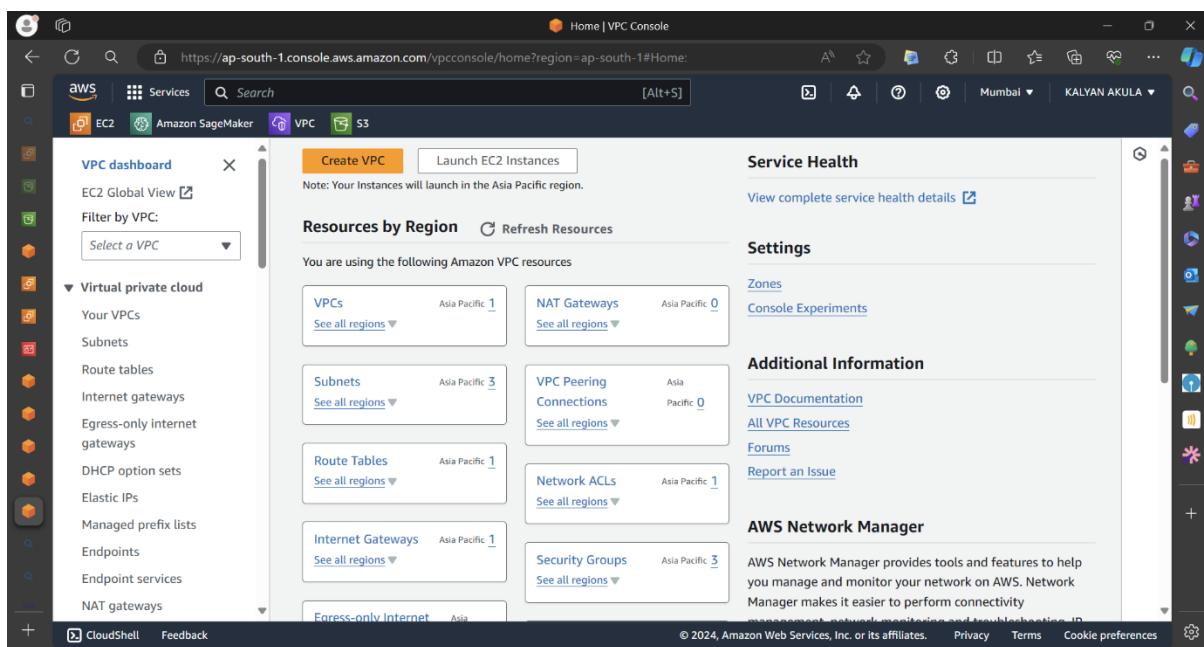


# Creating 2 VPC's in Mumbai region

Step1 : open AWS



Step2 : Open VPC



Step3 : Click on VPC & 'Create VPC'

The screenshot shows the AWS VPC Console interface. In the top navigation bar, the URL is https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#vpcs:. The left sidebar has a 'Virtual private cloud' section with various sub-links like 'Your VPCs', 'Subnets', 'Route tables', etc. The main area is titled 'Your VPCs (1) Info' and lists one VPC entry:

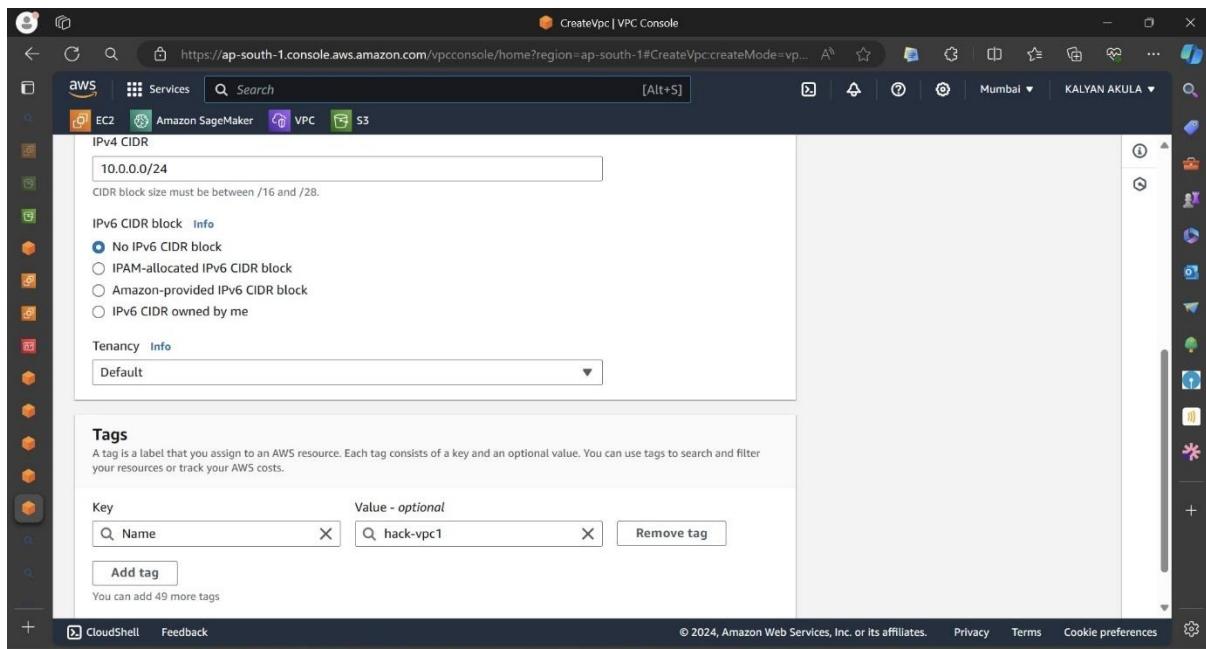
Name	VPC ID	State	IPv4 CIDR
-	vpc-08721c9ea937ffdd6	Available	172.31.0.0/16

Below the table, there's a note 'Select a VPC above' with three small icons: a magnifying glass, a pencil, and a trash can.

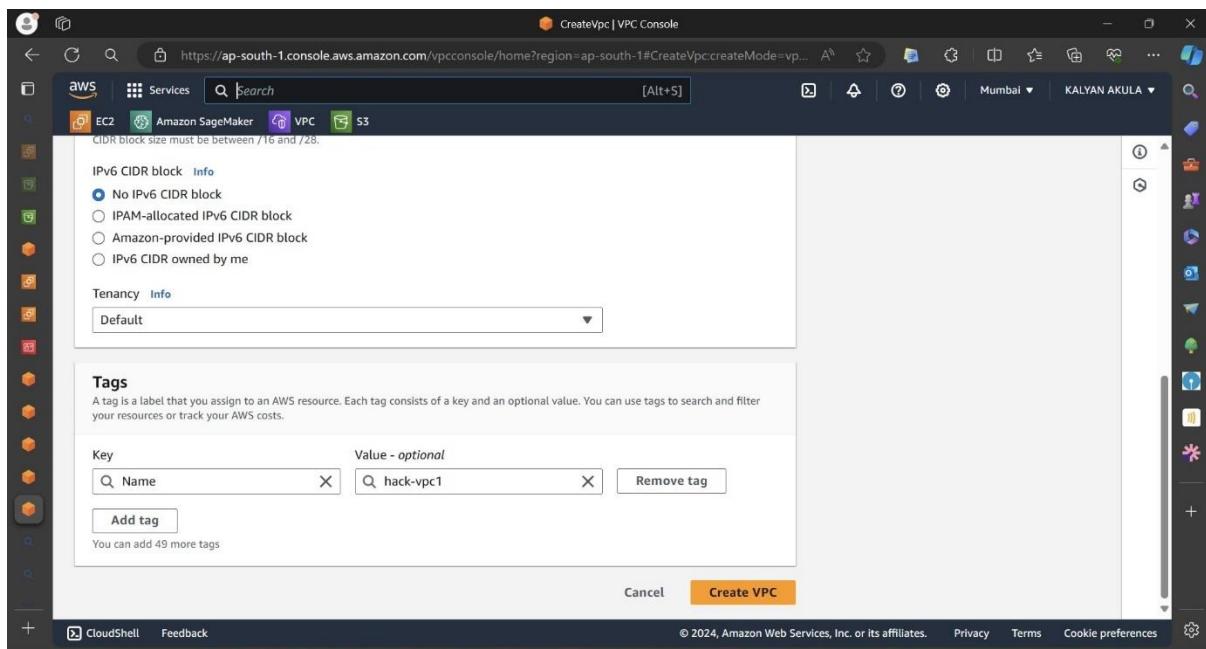
#### Step 4 : Enter the details as follows

The screenshot shows the 'Create VPC' configuration page in the AWS VPC Console. The URL is https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateVpc:createMode=vpc... . The left sidebar shows the path 'VPC > Your VPCs > Create VPC'. The main form is titled 'Create VPC' and contains the following fields:

- VPC settings**
  - Resources to create**: A dropdown menu with two options: 'VPC only' (selected) and 'VPC and more'.
  - Name tag - optional**: An input field containing 'hack-vpc1'.
  - IPv4 CIDR block**:
    - A radio button for 'IPv4 CIDR manual input' (selected).
    - A radio button for 'IPAM-allocated IPv4 CIDR block'.
  - IPv4 CIDR**: An input field containing '10.0.0.0/24'.



### Step 5 : Click on 'Create VPC'



### Step 6 : Click on 'Route tables' at left side to the window

The screenshot shows the AWS VPC console with a success message: "You successfully created **vpc-0f8b5fc7429d301d4 / hack-vpc1**". The main pane displays the details of the VPC, including its ID, state, and various network configurations. The sidebar on the left shows the "Virtual private cloud" section with options like "Your VPCs", "Subnets", "Route tables", etc.

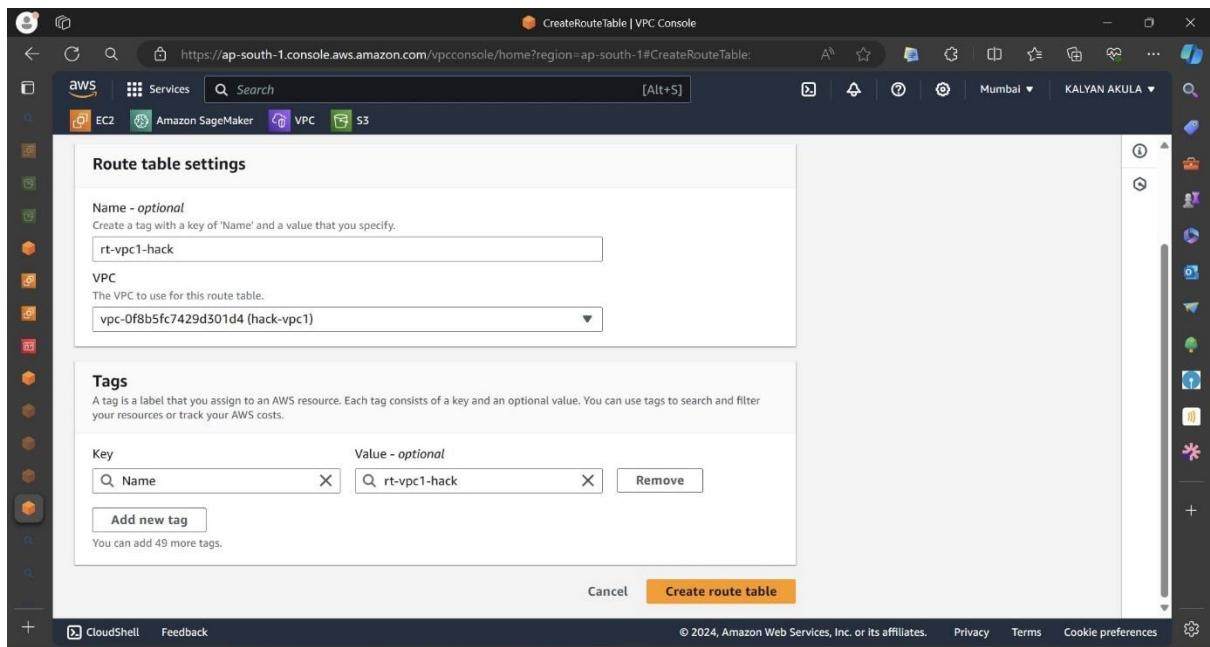
VPC ID	State	DNS hostnames	DNS resolution
vpc-0f8b5fc7429d301d4	Available	Disabled	Enabled
Tenancy	Default	DHCP option set	Main route table
Default		dopt-0db853259ab6a86eb	rtb-0e1928f5accbe2457
Default VPC	No	IPv4 CIDR	IPv6 pool
		10.0.0.0/24	-
Network Address Usage metrics	Disabled	Route 53 Resolver DNS	Main network ACL
		Firewall rule groups	acl-06f5b0d2d58a9d9a2
		-	IPv6 CIDR (Network border group)
			-
		Owner ID	471112701347

### Step 7: Click on 'Create Route Table'

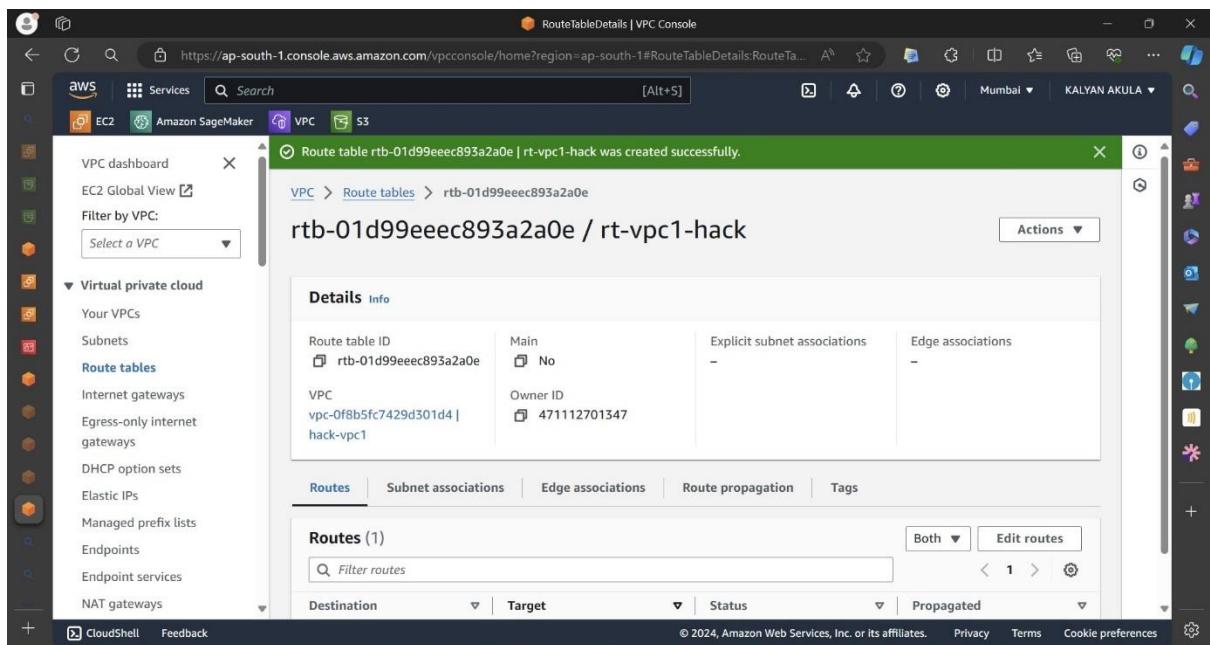
The screenshot shows the AWS VPC console with the "Route tables" section selected. It lists two existing route tables: "rtb-0e1928f5accbe2457" and "rtb-02f52e80f3941f8cd". A "Create route table" button is visible at the top right of the table header.

Name	Route table ID	Explicit subnet associations	Edge associations
-	rtb-0e1928f5accbe2457	-	-
-	rtb-02f52e80f3941f8cd	-	-

### Step 8 : Fill the details as follows & click on 'Create Table'



### Step 9: Click on Subnet at left of window



### Step 10 : Click on create 'Subnet'

The screenshot shows the AWS VPC Subnets console. On the left, there's a navigation sidebar with options like VPC dashboard, EC2 Global View, Filter by VPC, Virtual private cloud, Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, and NAT gateways. The main area is titled "Subnets (3) Info" and displays a table with three rows of subnet information:

Name	Subnet ID	State	VPC
-	subnet-0e0085abb6e3f76	Available	vpc-08721c9ea937ffdd6
-	subnet-030142aa921d02603	Available	vpc-08721c9ea937ffdd6
-	subnet-0d8681833e2ca07f8	Available	vpc-08721c9ea937ffdd6

At the bottom, there's a section titled "Select a subnet" with three small icons.

### Step 11 : Fill the following Details

The screenshot shows the AWS Create Subnet console. The top navigation bar includes "CreateSubnet | VPC Console", "Services", "Search", "Mumbai", "KALYAN AKULA", and various AWS service icons. The main content area has a breadcrumb trail: "VPC > Subnets > Create subnet". The page is titled "Create subnet" and contains two main sections: "VPC" and "Subnet settings".

**VPC** section:

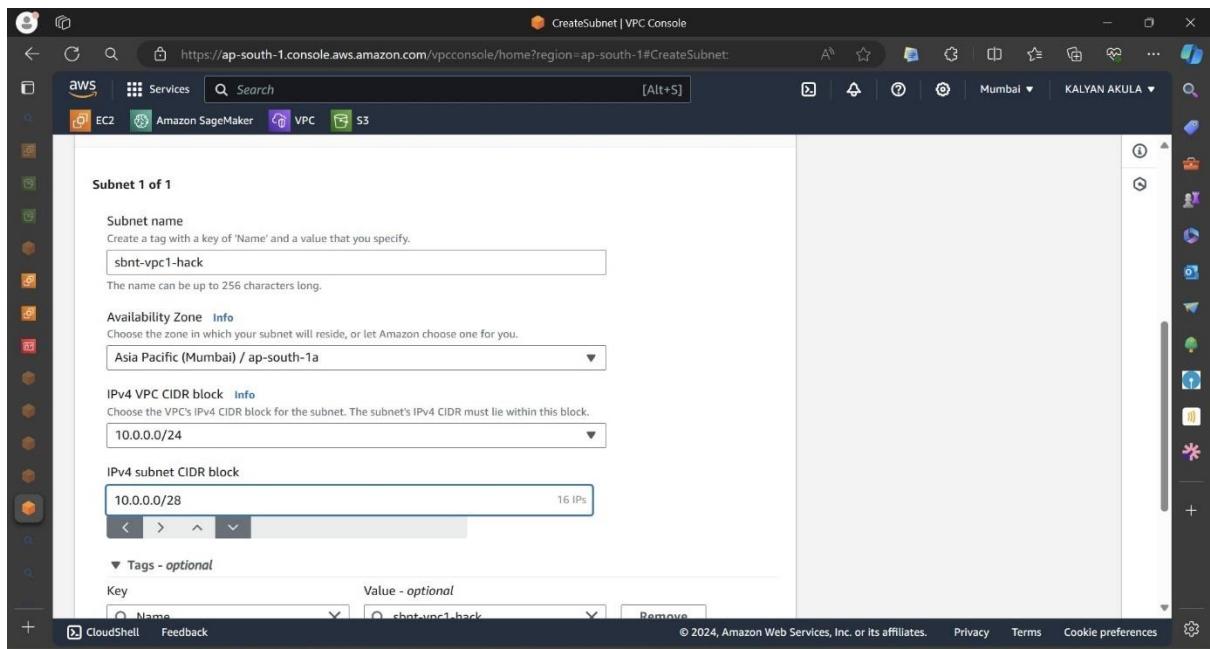
- VPC ID: A dropdown menu showing "vpc-0f8b5fc7429d301d4 (hack-vpc1)".
- Associated VPC CIDRs:
  - IPv4 CIDR: "10.0.0.0/24"

**Subnet settings** section:

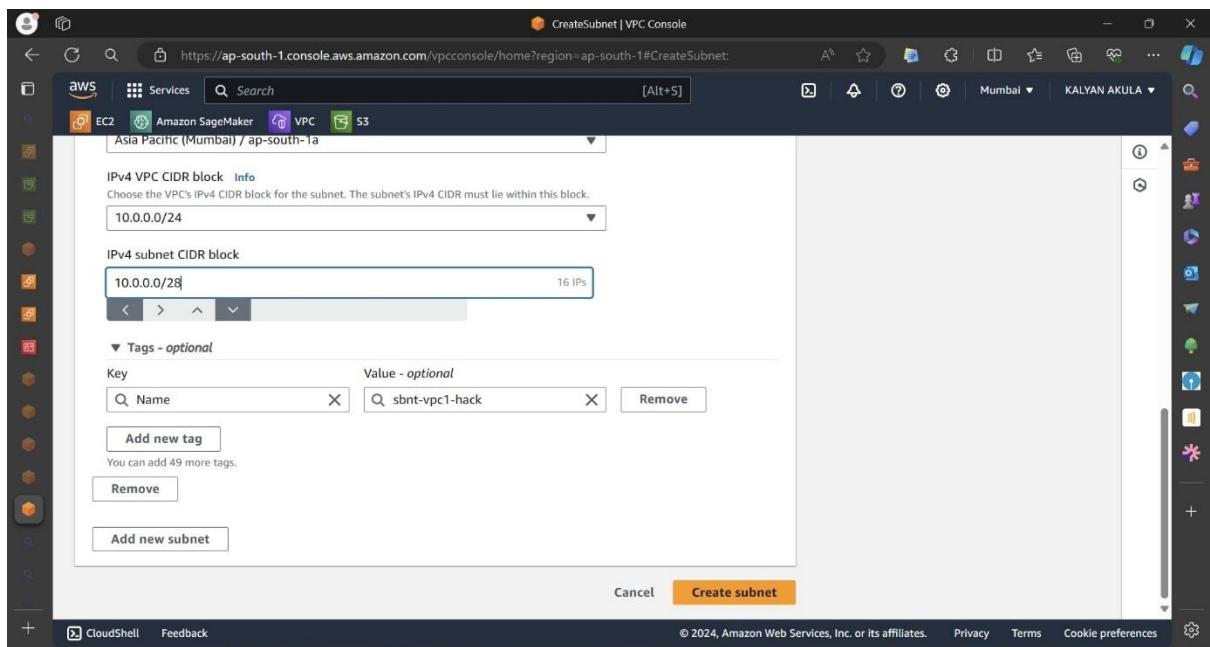
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1** (highlighted in blue)

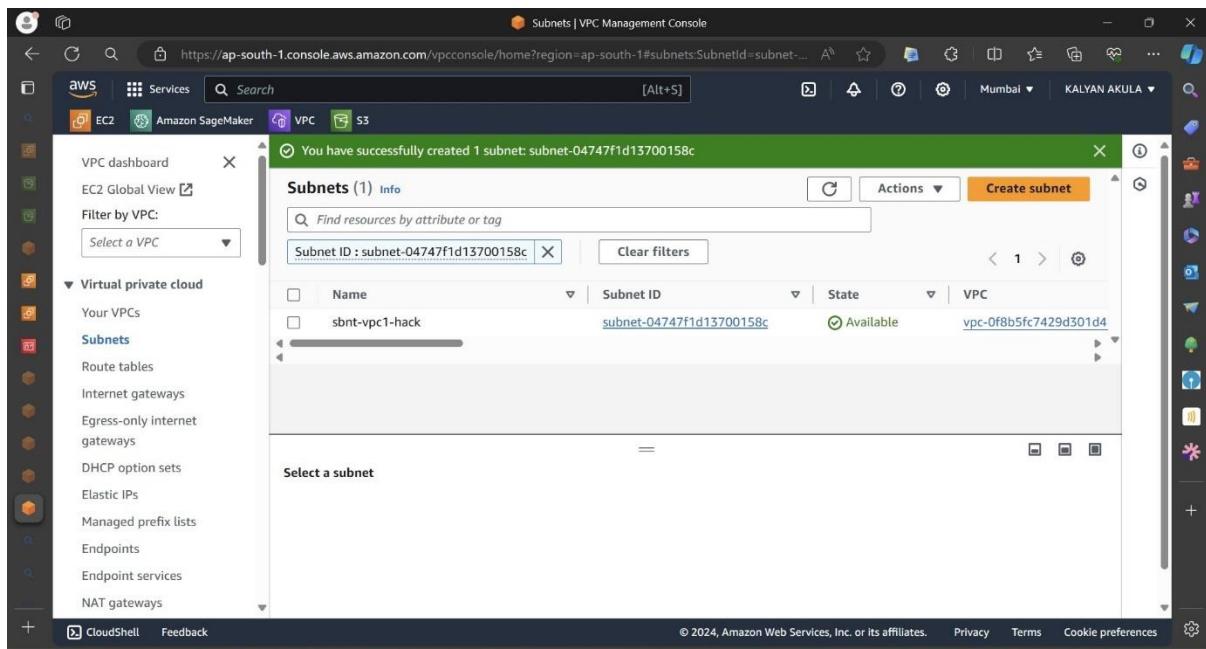
At the bottom, there are "CloudShell" and "Feedback" links, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.



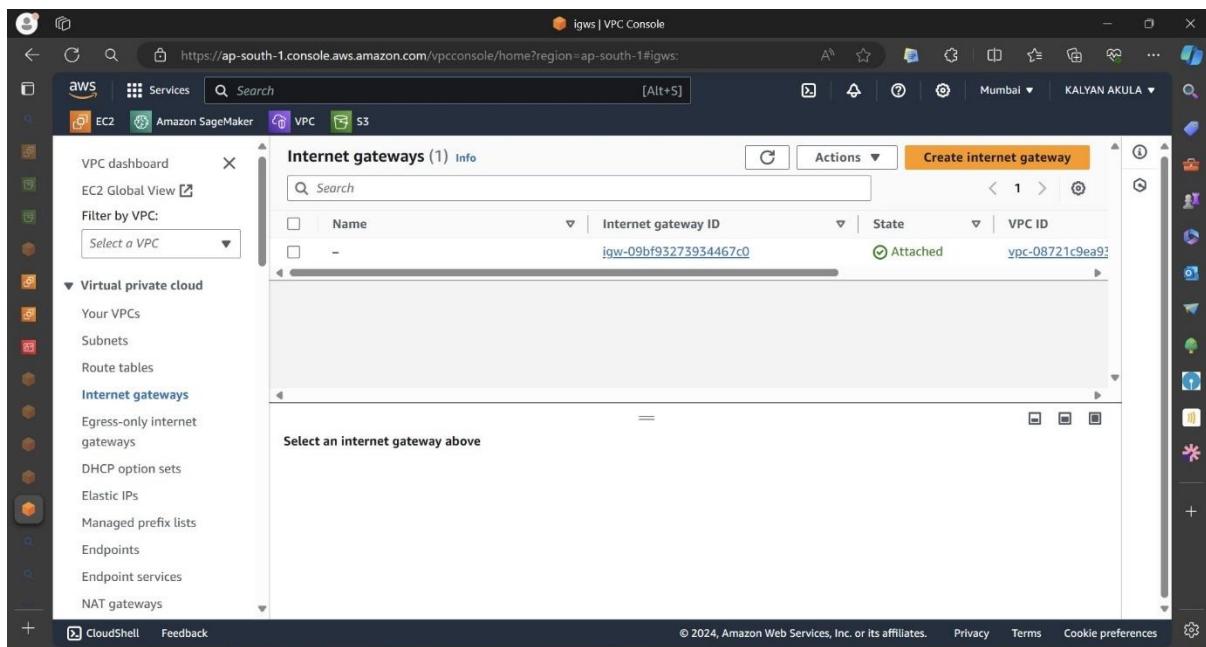
### Step 12 : Click on 'Create Subnet'



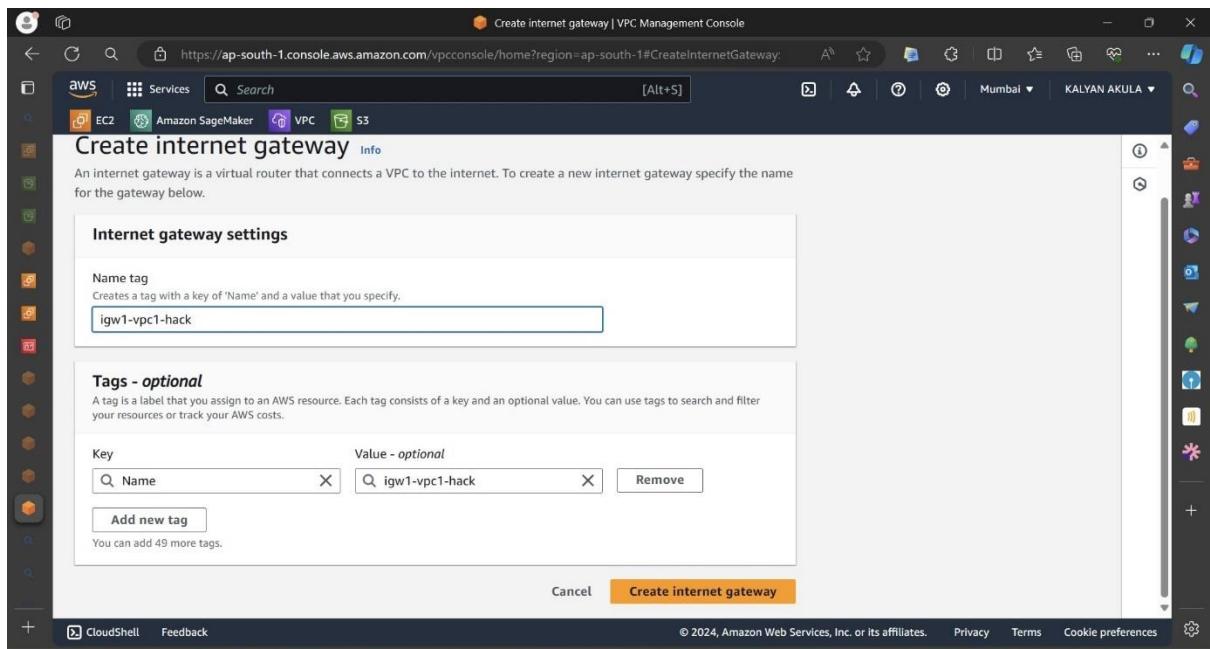
### Step 13 : click on 'Internet Gateway'



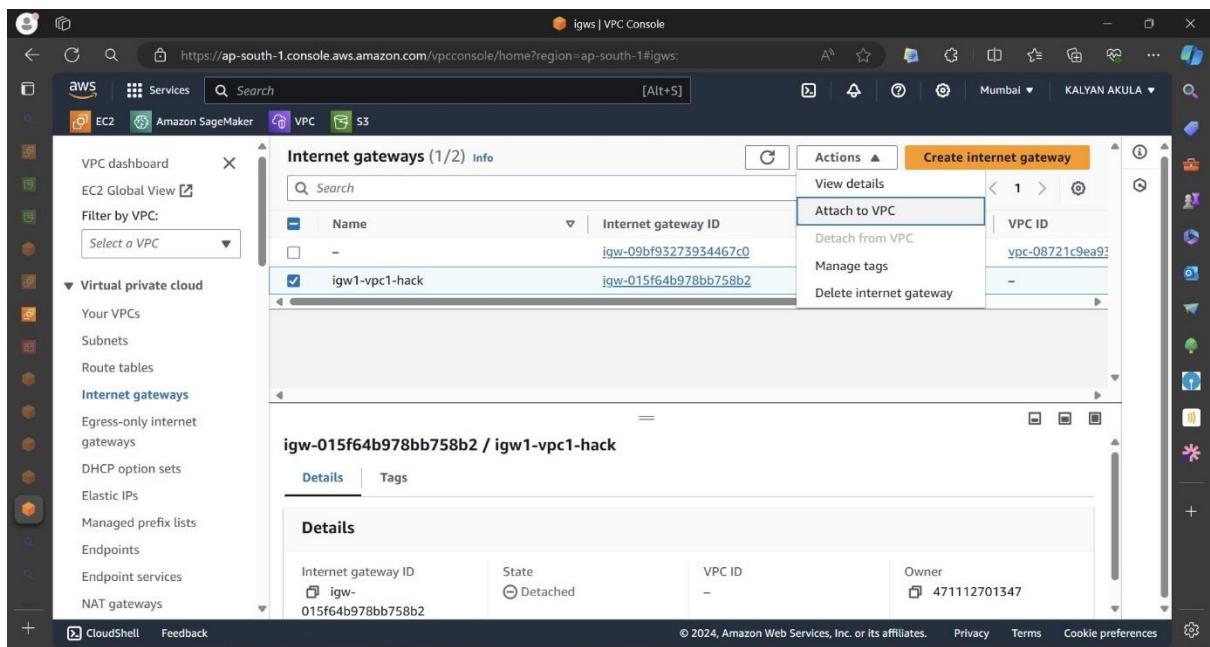
**Step 15 :** Click on 'Create internet gateway'



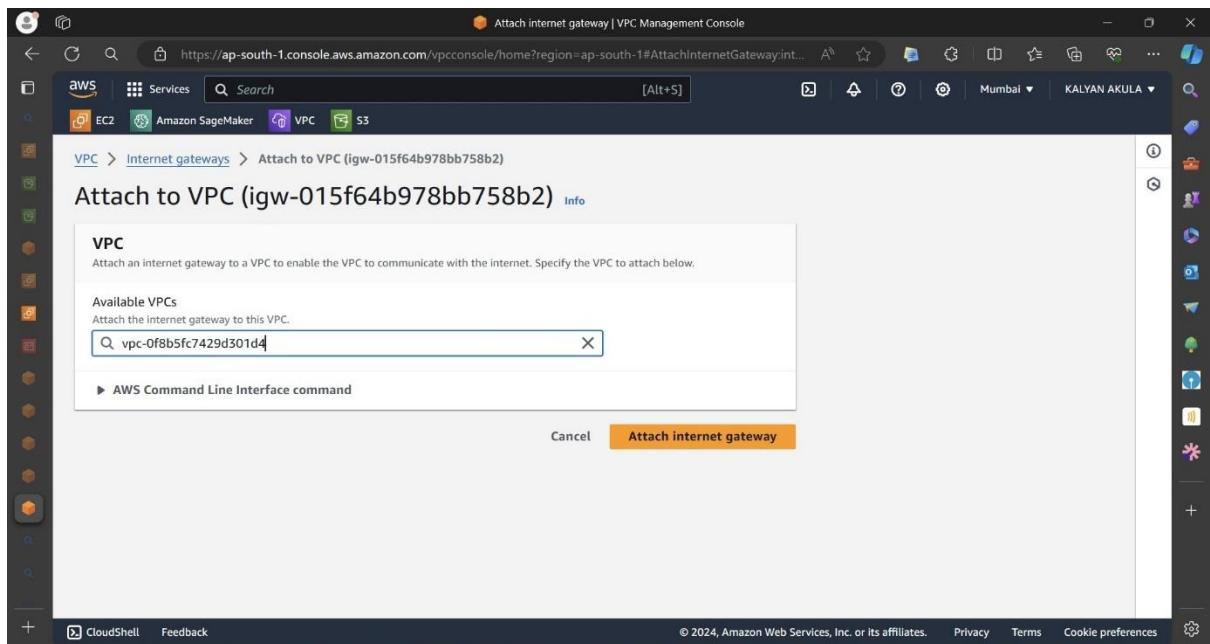
**Step 16 :** fill the following details & click on 'Create internet gateway'



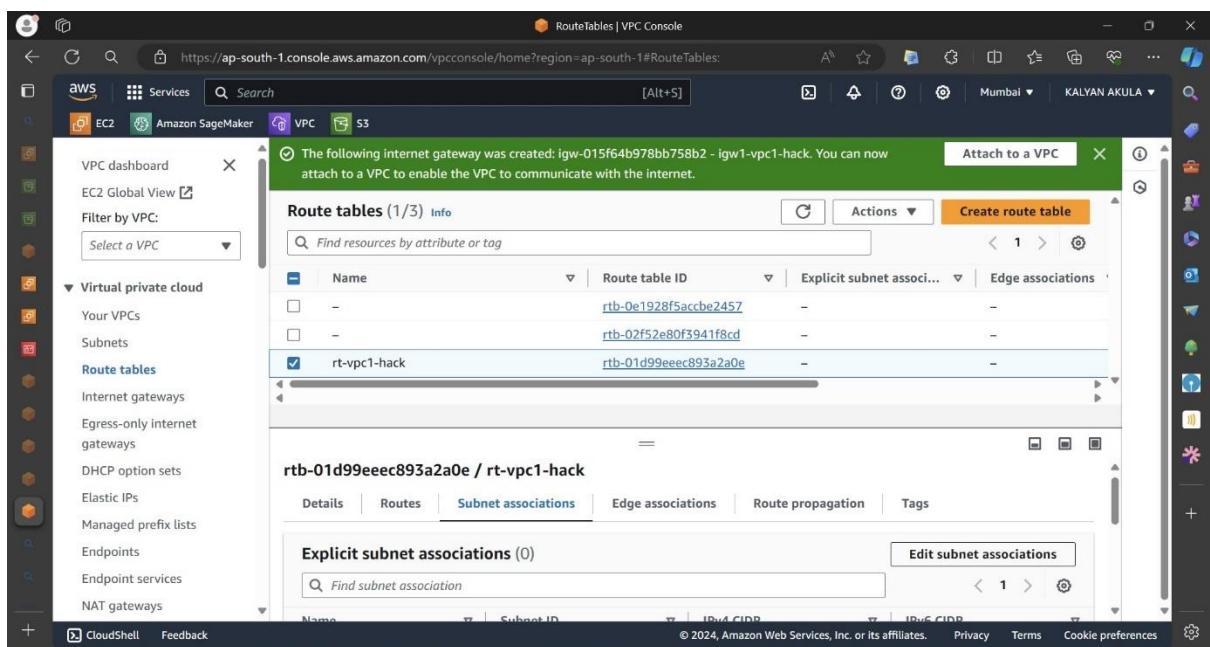
**Step 17 :** Select the Internet gateway , Click on ‘Actions’ & ‘Attach to VPC’



**Step 18 :** Fill the details & Click ‘Attach internet gateway’



**Step19 :** Click on ‘Route table’ , ‘Subnet associations’ & ‘edit subnet associations’



**Step 20 :** Select the subnet & click on Save Submissions

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (1/1)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
sbnt-vpc1-hack	subnet-04747f1d13700158c	10.0.0.0/28	-	Main (rtb-0e1928f5accbe2457)

**Selected subnets**

subnet-04747f1d13700158c / sbnt-vpc1-hack X
---

Cancel **Save associations**

**Step 21 :** Now click on ‘Route tables’ , select the route , click on ‘Routes’ & Edit routes’

**Route tables (1/3) Info**

You have successfully updated subnet associations for rtb-01d99eeec893a2a0e / rt-vpc1-hack.

Name	Route table ID	Explicit subnet associations	Edge associations
-	rtb-0e1928f5accbe2457	-	-
-	rtb-02f52e80f3941f8cd	-	-
<b>rt-vpc1-hack</b>	<b>rtb-01d99eeec893a2a0e</b>	<b>subnet-04747f1d13700158c</b>	-

**rtb-01d99eeec893a2a0e / rt-vpc1-hack**

Details | **Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (1)**

**Step 22 :** Fill the details in the following by adding a route & save the route

Route 1

Destination	Target	Status
10.0.0.0/24	local	Active

Propagated: No

Add route

Cancel Preview Save changes

Propagated: No

Route 2

Destination	Target	Status
0.0.0.0/0	Internet Gateway igw-015f64b978bb758b2	In Progress

Propagated: No

Add route

Remove

Cancel Preview Save changes

**Step 23 :** Now we need to add another route [ destination & source route]

Propagated

No

**Route 3**

Destination	Target	Status
192.218.0.0/24	Peering Connection pcx-039e8093159e4bb28	-

Propagated

No

Add route Remove

#### Step 24 : Now open EC2 in a new window

EC2 Dashboard

Instances

Images

Resources

Instances (running)	0	Auto Scaling Groups	0	Dedicated Hosts	0
Elastic IPs	0	Instances	0	Key pairs	2
Load balancers	0	Placement groups	0	Security groups	5
Snapshots	0	Volumes	0		

Launch instance

Migrate a server

AWS Health Dashboard

Region: Asia Pacific (Mumbai)

Status

#### Step 25 : Now Create a Instance with the Following Details

[Launch an instance | EC2 | ap-south-1](https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances)

aws Services Search [Alt+S] Mumbai KALYAN AKULA

EC2 Amazon SageMaker VPC S3

EC2 Instances Launch an instance

### Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags Info**

Name  Add additional tags

**Application and OS Images (Amazon Machine Image) Info**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

[Launch an instance | EC2 | ap-south-1](https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances)

aws Services Search [Alt+S] Mumbai KALYAN AKULA

EC2 Amazon SageMaker VPC S3

### Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux  [Browse more AMIs](#) Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

**Amazon Linux 2023 AMI** ami-09298640a92b2d12c (64-bit (x86), uefi-preferred) / ami-0d0bed6857ceda4b (64-bit (Arm), uefi)  
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible

**Description**  
Amazon Linux 2023 AMI 2023.4.20240401.1 x86\_64 HVM kernel-6.1

**Architecture** 64-bit (x86) **Boot mode** uefi-preferred **AMI ID** ami-09298640a92b2d12c Verified provider

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Launch an instance | EC2 | ap-south-1

https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Services Search [Alt+S] Mumbai KALYAN AKULA

EC2 Amazon SageMaker VPC S3

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

Click refresh to view backup information  
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems Edit

Advanced details Info

Summary

Number of instances Info  
1

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Launch an instance | EC2 | ap-south-1

https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Services Search [Alt+S] Mumbai KALYAN AKULA

EC2 Amazon SageMaker VPC S3

Source type Info Source Info Description - optional Info

Anywhere Add CIDR, prefix list or security group e.g. SSH for admin desktop 0.0.0.0/0 X

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Add security group rule Advanced network configuration

Configure storage Info Advanced

1x 8 GiB gp3 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

[Launch an instance | EC2 | ap-south-1](https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances)

**vpc1-sg**

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_.-:/()#@[]+=&.;|\$^

**Description - required** Info  
launch-wizard-1 created 2024-04-03T22:16:56.307Z

**Inbound Security Group Rules**

**Security group rule 1 (TCP, 22, 0.0.0.0/0)**

Type Info Protocol Info Port range Info  
ssh TCP 22

Source type Info Source Info Description - optional Info  
Anywhere Add CIDR, prefix list or security group e.g. SSH for admin desktop  
0.0.0.0/0 X

**Security group rule 2 (All, All, 0.0.0.0/0)**

Type Info Protocol Info Port range Info  
All traffic All All

**CloudShell Feedback**

[Launch an instance | EC2 | ap-south-1](https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances)

**Network settings** Info

**VPC - required** Info  
vpc-0f8b5fc7429d301d4 (hack-vpc)  
10.0.0.0/24

**Subnet** Info  
subnet-04747f1d13700158c sbnt-vpc1-hack  
VPC: vpc-0f8b5fc7429d301d4 Owner: 471112701347 Availability Zone: ap-south-1a IP addresses available: 10 CIDR: 10.0.0.0/28

**Create new subnet**

**Auto-assign public IP** Info  
Enable Additional charges apply when outside of free tier allowance

**Firewall (security groups)** Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group  Select existing security group

**Security group name - required**  
vpc1-sg

<https://ap-south-1.console.aws.amazon.com/vpc/home?region=ap-south-1#CreateSubnet>

The screenshot shows two consecutive steps of the AWS EC2 Launch Instance wizard.

**Step 2: Instance type**

Selected instance type: t2.micro (Free tier eligible)

Key pair (login): kalyan

**Step 3: Summary**

Number of instances: 1

Software Image (AMI): Amazon Linux 2023.4.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Buttons: Cancel, Launch instance, Review commands

**Step 26 :** Now follow from Step 2 – Step 26 , for creating another VPC

## Peering connection between VPC1&VPC2

**Step 1 :** Go to VPC Dashboard & open Peering Connections

The screenshot shows the AWS VPC Console Home page. On the left, a sidebar lists various VPC-related resources: Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections, Security (Network ACLs, Security groups), DNS firewall (Rule groups), and Peering connections. The main content area displays 'Resources by Region' with the following data:

Type	Region	Count
VPCs	Asia Pacific	3
NAT Gateways	Asia Pacific	0
Subnets	Asia Pacific	5
VPC Peering Connections	Asia Pacific	0
Route Tables	Asia Pacific	5
Network ACLs	Asia Pacific	3
Internet Gateways	Asia Pacific	3
Security Groups	Asia Pacific	7
Egress-only Internet	Asia	0

On the right, there are sections for Service Health (View complete service health details), Settings (Zones, Console Experiments), Additional Information (VPC Documentation, All VPC Resources, Forums, Report an Issue), and AWS Network Manager (Provides tools and features to help manage and monitor your network on AWS). The bottom of the page includes links for CloudShell, Feedback, and copyright information (© 2024, Amazon Web Services, Inc. or its affiliates.).

## Step 2 : Click on 'Create Peering Connection'

The screenshot shows the AWS PeeringConnections | VPC Console page. The left sidebar is identical to the VPC Home page, showing the same list of resources. The main content area is titled 'Peering connections' and contains a table with one row: 'No peering connection found'. Below the table, a message says 'Select a peering connection above'. The top right of the page has a prominent orange 'Create peering connection' button. The bottom of the page includes links for CloudShell, Feedback, and copyright information.

## Step 3: Fill the following details & click on 'Create peering connection'

CreatePeeringConnection | VPC Console

https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreatePeeringConnection: [Alt+S]

Mumbai | KALYAN AKULA

aws Services Search [Alt+S]

EC2 Amazon SageMaker VPC S3

VPC > Peering connections > Create peering connection

## Create peering connection

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately.

Info

**Peering connection settings**

Name - optional  
Create a tag with a key of 'Name' and a value that you specify.  
vpc1-vpc2

Select a local VPC to peer with

VPC ID (Requester)  
vpc-0f8b5fc7429d301d4 (hack-vpc1)

VPC CIDRs for vpc-0f8b5fc7429d301d4 (hack-vpc1)

CIDR	Status	Status reason
10.0.0.0/24	Associated	-

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CreatePeeringConnection | VPC Console

https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreatePeeringConnection: [Alt+S]

Mumbai | KALYAN AKULA

aws Services Search [Alt+S]

EC2 Amazon SageMaker VPC S3

CIDR Status Status reason

10.0.0.0/24 Associated -

Select another VPC to peer with

Account  
 My account  
 Another account

Region  
 This Region (ap-south-1)  
 Another Region

VPC ID (Acceptor)  
vpc-068b8032f3cd333c6 (hack-vpc2)

VPC CIDRs for vpc-068b8032f3cd333c6 (hack-vpc2)

CIDR	Status	Status reason
192.218.0.0/24	Associated	-

Tags  
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 across different AWS accounts.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CreatePeeringConnection | VPC Console

This Region (ap-south-1)

VPC ID (Acceptor)

vpc-068b8032f3cd333c6 (hack-vpc2)

VPC CIDRs for vpc-068b8032f3cd333c6 (hack-vpc2)

CIDR	Status	Status reason
192.218.0.0/24	Associated	-

**Tags**

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	Value - optional
Name	vpc1-vpc2

Add new tag

You can add 49 more tags.

Cancel   **Create peering connection**

**Step 4 :** Now click on 'Actions' & 'Accept request'

PeeringConnectionDetails | VPC Console

A VPC peering connection pcx-039e8093159e4bb28 / vpc1-vpc2 has been requested.

VPC > Peering connections > pcx-039e8093159e4bb28

**pcx-039e8093159e4bb28 / vpc1-vpc2**

**Pending acceptance**

You can accept or reject this peering connection request using the 'Actions' menu. You have until 04:11:36 GMT+5:30 to accept or reject the request, otherwise it expires.

**Actions**

- Accept request
- Reject request
- Edit DNS settings
- Manage tags
- Delete peering connection

**Details**

Requester owner ID	Acceptor owner ID	VPC Peering connection ARN
471112701347	471112701347	arn:aws:ec2:ap-south-1:471112701347:vpc-peering-connection/pcx-039e8093159e4bb28
Peering connection ID	Requester VPC	Acceptor VPC
pcx-039e8093159e4bb28	vpc-0f8b5fc7429d301d4 / hack-vpc1	vpc-068b8032f3cd333c6 / hack-vpc2
Status	Requester CIDRs	Acceptor CIDRs
Pending Acceptance by 471112701347	10.0.0.0/24	-
Expiration time	Requester Region	Acceptor Region
Thursday, April 11, 2024 at 04:11:36	Mumbai (ap-south-1)	-

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Step 5 :** Go to EC2 Dashboard

**Step 6 :** Select Inst1-vpc1 & click on connect

The screenshot shows the AWS EC2 Instances page. The left sidebar has sections for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. The main area shows 'Instances (1/3) Info' with a table. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, and Alarm status. There are three rows: 1. inst1-vpc1 (i-0f07eceeба626b5a) - Running, t2.micro, 2/2 checks passed, View alarms. 2. inst2-vpc2 (i-0500c548c32ea288) - Running, t2.micro, 2/2 checks passed, View alarms. 3. inst1-vpc1 (i-068fc4e595be629f0) - Terminated, t2.micro, - (empty), View alarms. A search bar at the top says 'Find Instance by attribute or tag (case-sensitive)' and dropdowns for 'All states' and 'Actions'. An orange 'Launch instances' button is at the top right. Below the table is a section for 'Instance: i-0f07eceeба626b5a (inst1-vpc1)' with tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is selected. It shows 'Instance summary' with fields: Instance ID (i-0f07eceeба626b5a (inst1-vpc1)), Public IPv4 address (43.205.236.227), Private IPv4 addresses (10.0.0.7), IPv6 address (-), Instance state (Running), and Public IPv4 DNS (-).

### Step 7 : Click on connect

The screenshot shows the 'Connect to instance' dialog box. It has a title 'Connect to instance | EC2 | ap-south-1'. The left sidebar is the same as the previous screenshot. The main area has 'Instance ID' set to 'i-0f07eceeба626b5a (inst1-vpc1)'. Under 'Connection Type', there are two options: 'Connect using EC2 Instance Connect' (selected) and 'Connect using EC2 Instance Connect Endpoint'. Both options have a sub-note: 'Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.' Below this, 'Public IP address' is listed as '43.205.236.227'. Under 'Username', it says 'Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.' A text input field contains 'ec2-user'. At the bottom is a note: 'Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.' At the bottom right are 'Cancel' and 'Connect' buttons.

- Linux interface Opens

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Wed Apr  3 23:07:39 2024 from 13.233.177.3
[ec2-user@ip-10-0-0-7 ~]$
```

i-0f07eceebaa626b5a (inst1-vpc1)  
PublicIPs: 43.205.236.227 PrivateIPs: 10.0.0.7

**Step 8 :** In the same way we need to connect inst2-vpc2 & open Linux interface

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Wed Apr  3 23:00:55 2024 from 13.233.177.5
[ec2-user@ip-192-218-0-4 ~]$
```

i-0500c5485c32ea288 (inst2-vpc2)  
PublicIPs: 15.206.195.42 PrivateIPs: 192.218.0.4

**Step 9 :** In the linux interface , we need to enter command

- pin [ip of another instance]
  - if it runs successfully , the peering connection is established.

```
Last login: Wed Apr 3 23:28:53 2024 from 13.233.177.4  
[ec2-user@ip-192-218-0-4 ~]$ ping 10.0.0.7  
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data.  
64 bytes from 10.0.0.7: icmp_seq=1 ttl=127 time=1.07 ms  
64 bytes from 10.0.0.7: icmp_seq=2 ttl=127 time=0.845 ms  
64 bytes from 10.0.0.7: icmp_seq=3 ttl=127 time=0.913 ms  
64 bytes from 10.0.0.7: icmp_seq=4 ttl=127 time=0.774 ms  
64 bytes from 10.0.0.7: icmp_seq=5 ttl=127 time=0.870 ms  
64 bytes from 10.0.0.7: icmp_seq=6 ttl=127 time=0.924 ms
```

i-0500c5485c32ea288 (inst2-vpc2)  
Public IPs: 15.206.195.42 Private IPs: 192.218.0.4

#### Step 10 : Check same for inst2-vpc2

```
Last login: Wed Apr 3 23:12:24 2024 from 13.233.177.3  
[ec2-user@ip-10-0-0-7 ~]$ ping 192.218.0.4  
PING 192.218.0.4 (192.218.0.4) 56(84) bytes of data.  
64 bytes from 192.218.0.4: icmp_seq=1 ttl=127 time=0.854 ms  
64 bytes from 192.218.0.4: icmp_seq=2 ttl=127 time=0.829 ms
```

i-0f07eceebaa626b5a (inst1-vpc1)  
Public IPs: 43.205.236.227 Private IPs: 10.0.0.7

## Creating S3 Buckets in 2 different regions & establishing a Cross Region Replication

#### Step 1 : open S3 & click on Buckets

The screenshot shows the AWS S3 homepage. On the left, there's a sidebar with options like Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. Below that is a section for 'Block Public Access settings for this account'. Under 'Storage Lens', there are links for Dashboards, Storage Lens groups, and AWS Organizations settings. At the bottom of the sidebar are CloudShell, Feedback, and Language links. The main content area features a large heading 'Amazon S3' and the sub-headline 'Store and retrieve any amount of data from anywhere'. A paragraph explains that Amazon S3 offers scalability, data availability, security, and performance. A prominent orange 'Create bucket' button is centered in a call-to-action box. At the very bottom of the page, there's a footer with links for © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

## Step 2 : Create a Bucket

The screenshot shows the 'S3 buckets' page. The top navigation bar includes the AWS logo, Services, a search bar, and links for EC2, Amazon SageMaker, VPC, and S3. The left sidebar shows 'Amazon S3 > Buckets' and has a 'General purpose buckets' tab selected. There's also a 'View Storage Lens dashboard' button. The main content area displays a table of 'General purpose buckets' with two entries: 'bucket1-crr' and 'bucket2-crr'. The table columns are Name, AWS Region, IAM Access Analyzer, and Creation date. Each row has a circular icon, a 'Copy ARN' button, an 'Empty' button, a 'Delete' button, and an orange 'Create bucket' button. At the bottom of the table, there's a search bar labeled 'Find buckets by name' and a pagination indicator showing page 1 of 1. The footer contains links for CloudShell, Feedback, Language, and standard AWS footer links.

## Step 3 : Fill the details as follows

The screenshot shows two consecutive steps in the AWS S3 'Create bucket' wizard.

**Step 1: General configuration**

- AWS Region:** Asia Pacific (Mumbai) ap-south-1
- Bucket name:** bucket1-crr
- Copy settings from existing bucket - optional:** Choose bucket

**Step 2: Object Ownership**

- Object Ownership Info:** Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.
- ACLs disabled (recommended):** All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
- ACLs enabled:** Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

**Block Public Access settings for this bucket:**

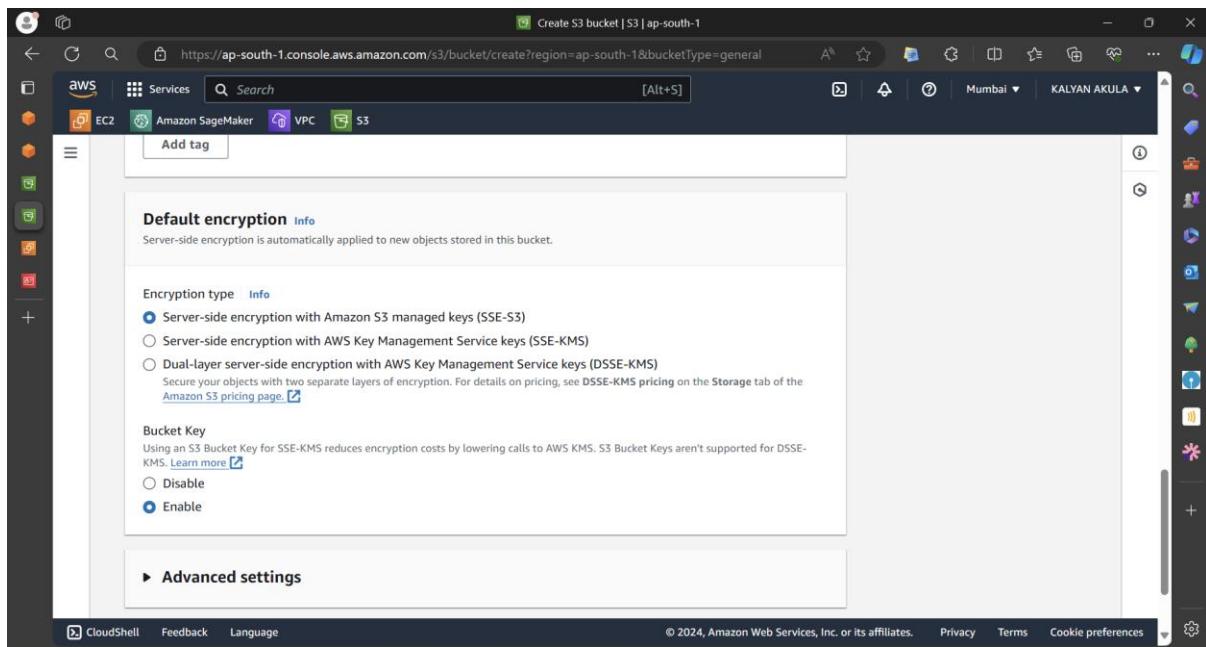
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Note:** Block Public Access settings for this account are currently turned on

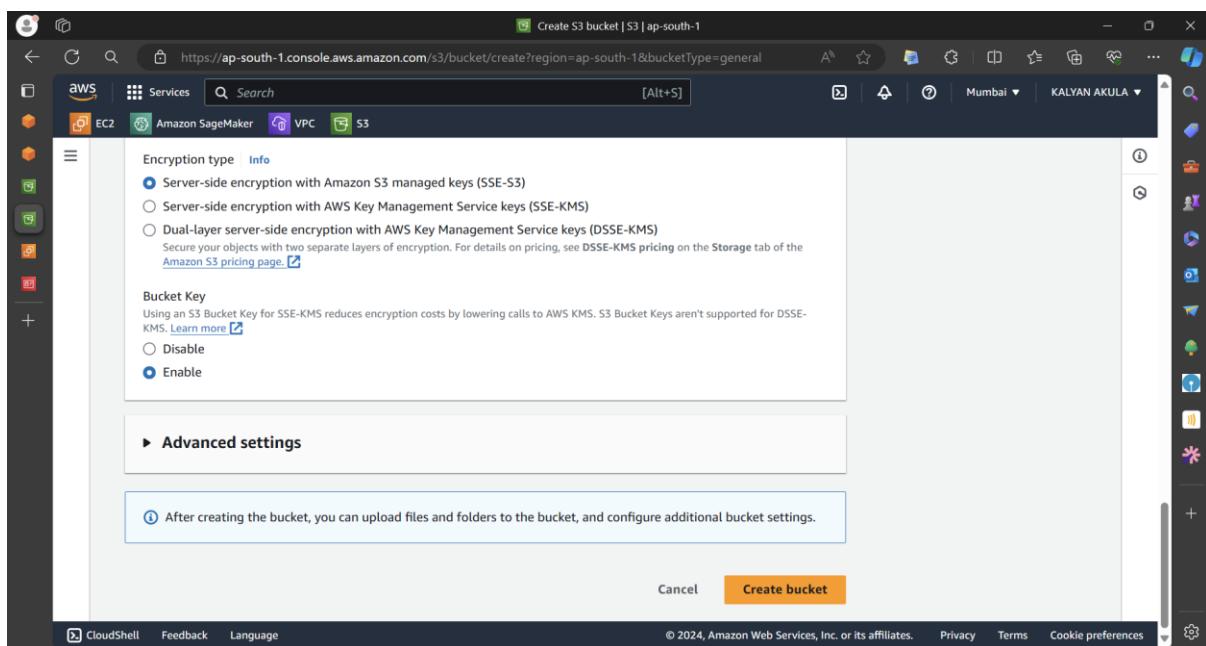
The screenshot shows the AWS S3 Bucket Creation wizard. The first step, "Block Public Access settings for this account are currently turned on", is displayed. It includes a note about existing enabled settings and four checkboxes under "Block all public access":

- Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

The second step, "Bucket Versioning", is shown next. It includes a note about versioning, a "Bucket Versioning" section with a radio button for "Enable" (which is selected), and a "Tags - optional (0)" section with an "Add tag" button.



#### Step 4 : Click on Create Bucket



#### Step 5 : Again create a Bucket with the following details

Create S3 bucket | S3 | ap-south-1

https://ap-south-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general [Alt+S]

Mumbai | KALYAN AKULA

Amazon S3 > Buckets > Create bucket

### Create bucket Info

Buckets are containers for data stored in S3.

#### General configuration

AWS Region: US East (N. Virginia) us-east-1

Bucket type: Info

General purpose  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

Directory - New  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name: Info  
bucket2-crr

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming [?]

CloudShell Feedback Language © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create S3 bucket | S3 | ap-south-1

https://ap-south-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general [Alt+S]

Mumbai | KALYAN AKULA

bucket2-crr

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming [?]

Copy settings from existing bucket - optional  
Only the bucket settings in the following configuration are copied.  
 Choose bucket  
Format: s3://bucket/prefix

#### Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)  
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled  
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

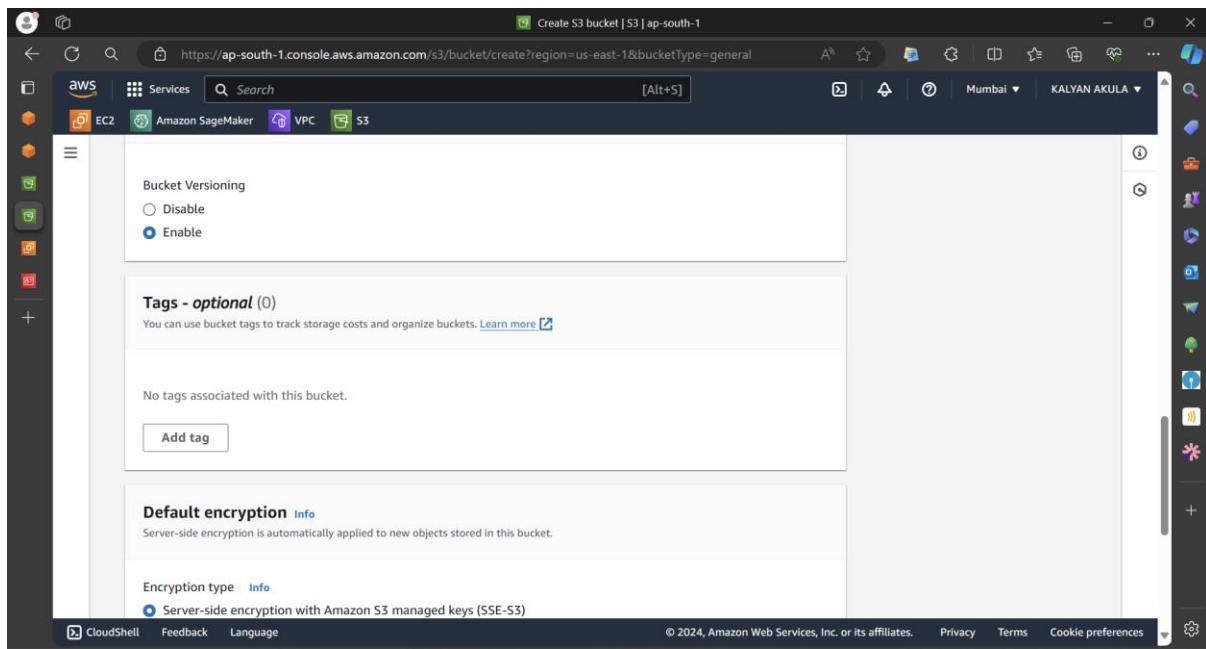
Object Ownership  
Bucket owner enforced

#### Block Public Access settings for this bucket

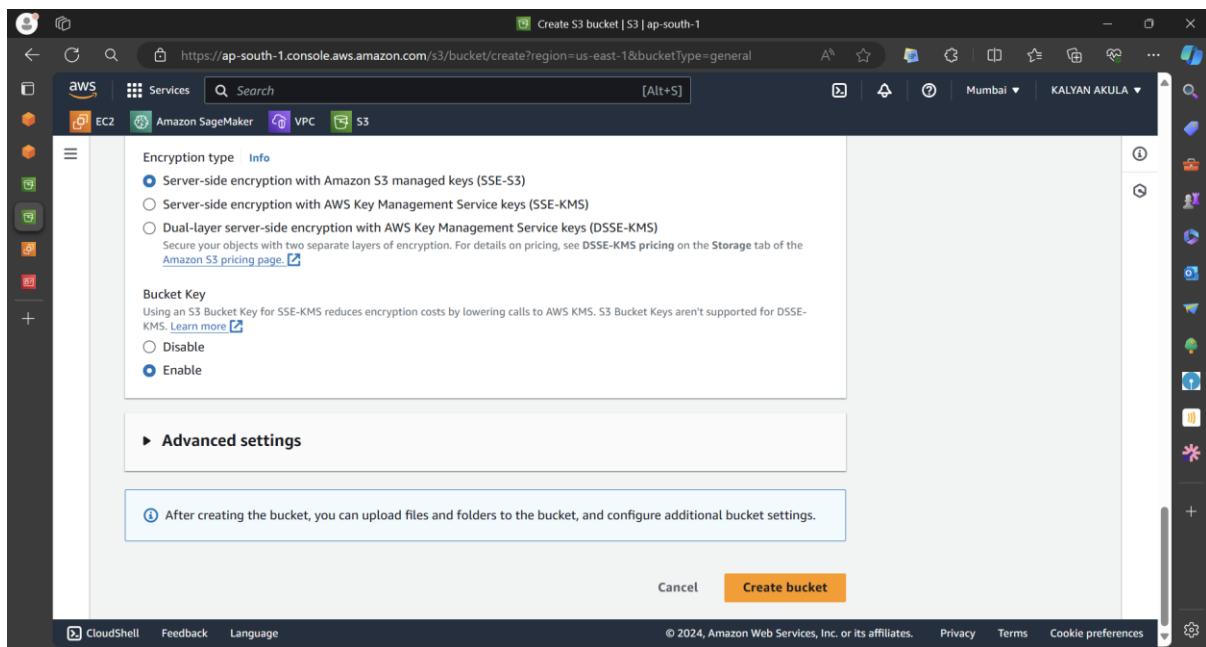
CloudShell Feedback Language © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS S3 Bucket Creation wizard, Step 3: Block Public Access settings for this bucket. The page title is "Create S3 bucket | S3 | ap-south-1". The main content area displays a message: "Block Public Access settings for this account are currently turned on" and "Block Public Access settings for this account [?] that are enabled apply even if they are disabled for this bucket." Below this, there is a section titled "Block all public access" with a checked checkbox. A note states: "Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another." Under this note, there are three additional checkboxes: "Block public access to buckets and objects granted through new access control lists (ACLs)", "Block public access to buckets and objects granted through any access control lists (ACLs)", and "Block public access to buckets and objects granted through new public bucket or access point policies".

This screenshot is identical to the one above, showing the AWS S3 Bucket Creation wizard, Step 3: Block Public Access settings for this bucket. The page title is "Create S3 bucket | S3 | ap-south-1". The main content area displays a message: "Block Public Access settings for this account are currently turned on" and "Block Public Access settings for this account [?] that are enabled apply even if they are disabled for this bucket." Below this, there is a section titled "Block all public access" with a checked checkbox. A note states: "Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another." Under this note, there are three additional checkboxes: "Block public access to buckets and objects granted through new access control lists (ACLs)", "Block public access to buckets and objects granted through any access control lists (ACLs)", and "Block public access to buckets and objects granted through new public bucket or access point policies".



## Step 6 : Click on Create Bucket



## Step 7 : Click on Bucket1-crr

bucket1-crr - S3 bucket | S3 | ap-south-1

Amazon S3 > Buckets > bucket1-crr

bucket1-crr

Objects (3) Info

Actions ▾ Create folder Upload

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Find objects by prefix

Show versions

Name	Type	Last modified	Size	Storage class
April 1, 2024				

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Step 8 :** Click on Management , in Replication rules Click on Create replication rule

bucket1-crr - S3 bucket | S3 | ap-south-1

Amazon S3 > Buckets > bucket1-crr

Management

Lifecycle rules View details Edit Delete Actions ▾ Create lifecycle rule

Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. Learn more

No lifecycle rules

There are no lifecycle rules for this bucket.

Create lifecycle rule

Replication rules (1)

View details Edit rule Delete Actions ▾ Create replication rule

Use replication rules to define options you want Amazon S3 to apply during replication such as server-side encryption, replica ownership, transitioning replicas to another storage class, and more. Learn more

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Step 9 :** Fill the data as follows

Create replication rule - S3 bucket bucket1-crr | S3 | ap-south-1

https://ap-south-1.console.aws.amazon.com/s3/management/bucket1-crr/replication/create?region=ap-south-1

Mumbai KALYAN AKULA

Services Search [Alt+S]

EC2 Amazon SageMaker VPC S3

Amazon S3 > Buckets > bucket1-crr > Replication rules > Create replication rule

## Create replication rule Info

**Replication rule configuration**

Replication rule name: replicationrule1  
Up to 255 characters. In order to be able to use CloudWatch metrics to monitor the progress of your replication rule, the replication rule name must only contain English characters.

Status: Enabled

Priority: 1

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback Language

Create replication rule - S3 bucket bucket1-crr | S3 | ap-south-1

https://ap-south-1.console.aws.amazon.com/s3/management/bucket1-crr/replication/create?region=ap-south-1

Mumbai KALYAN AKULA

Services Search [Alt+S]

EC2 Amazon SageMaker VPC S3

1

**Source bucket**

Source bucket name: bucket1-crr

Source Region: Asia Pacific (Mumbai) ap-south-1

Choose a rule scope: Apply to all objects in the bucket

**Destination**

Destination: You can replicate objects across buckets in different AWS Regions (Cross-Region Replication) or you can replicate objects across buckets in the same AWS Region (Same-Region Replication). You can also specify a different bucket for each rule in the configuration. [Learn more](#) or see [Amazon S3 pricing](#)

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback Language

Create replication rule - S3 bucket bucket1-crr | S3 | ap-south-1

aws Services Search [Alt+S] Mumbai KALYAN AKULA

EC2 Amazon SageMaker VPC S3

**Destination**

You can replicate objects across buckets in different AWS Regions (Cross-Region Replication) or you can replicate objects across buckets in the same AWS Region (Same-Region Replication). You can also specify a different bucket for each rule in the configuration. [Learn more](#) or see [Amazon S3 pricing](#).

Choose a bucket in this account  
 Specify a bucket in another account

**Bucket name**  
Choose the bucket that will receive replicated objects.  
bucket2-crr [Browse S3](#)

**Destination Region**  
US East (N. Virginia) us-east-1

**IAM role**

The selected IAM role applies to all rules in the bucket. To change the IAM role for all rules in the bucket, edit the [Replication configuration settings](#).

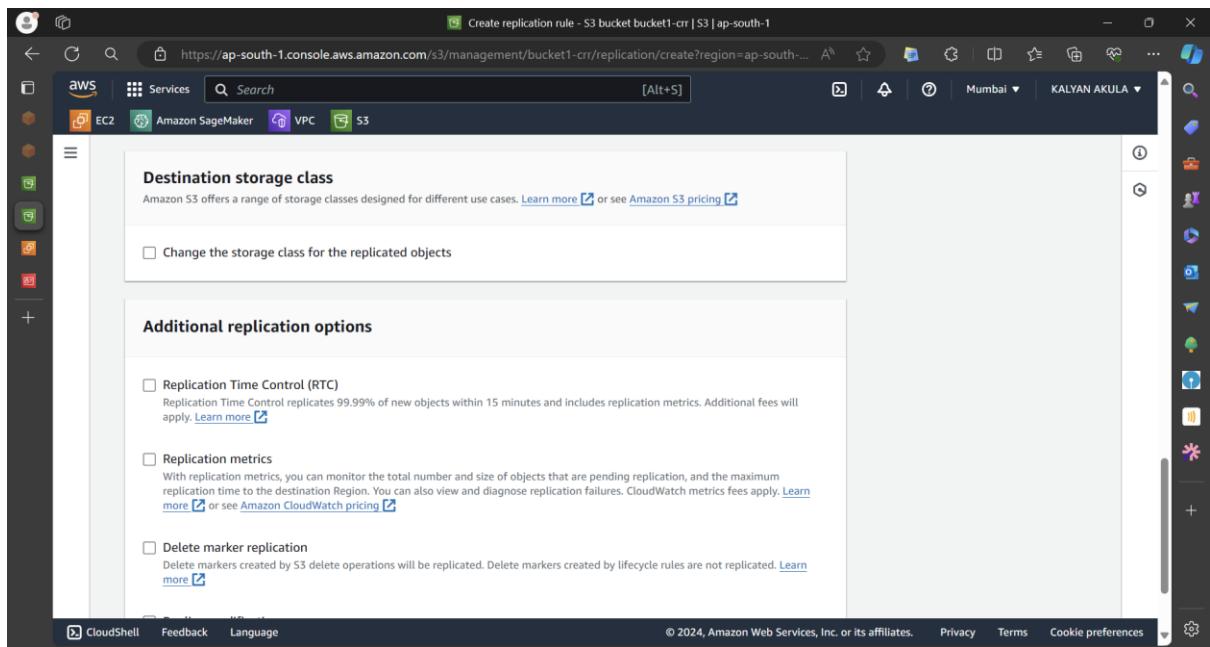
**IAM role**  
[S3crr\\_role\\_for\\_bucket1-crr](#)

**Encryption**  
Server-side encryption protects data at rest.

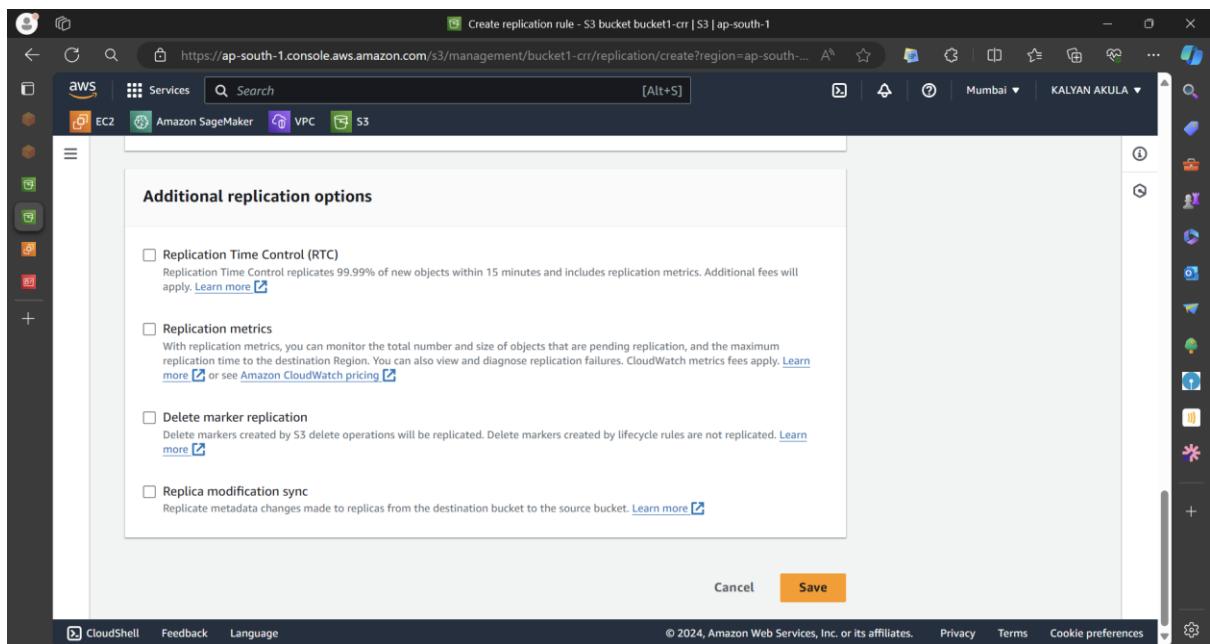
Replicate objects encrypted with AWS Key Management Service (AWS KMS)  
Replicate SSE-KMS and DSSE-KMS encrypted objects.

**Destination storage class**

CloudShell Feedback Language © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



### Step 9 : Click on Save



- To check whether the CRR is established or not

### Step 10 : open bucket1-crr

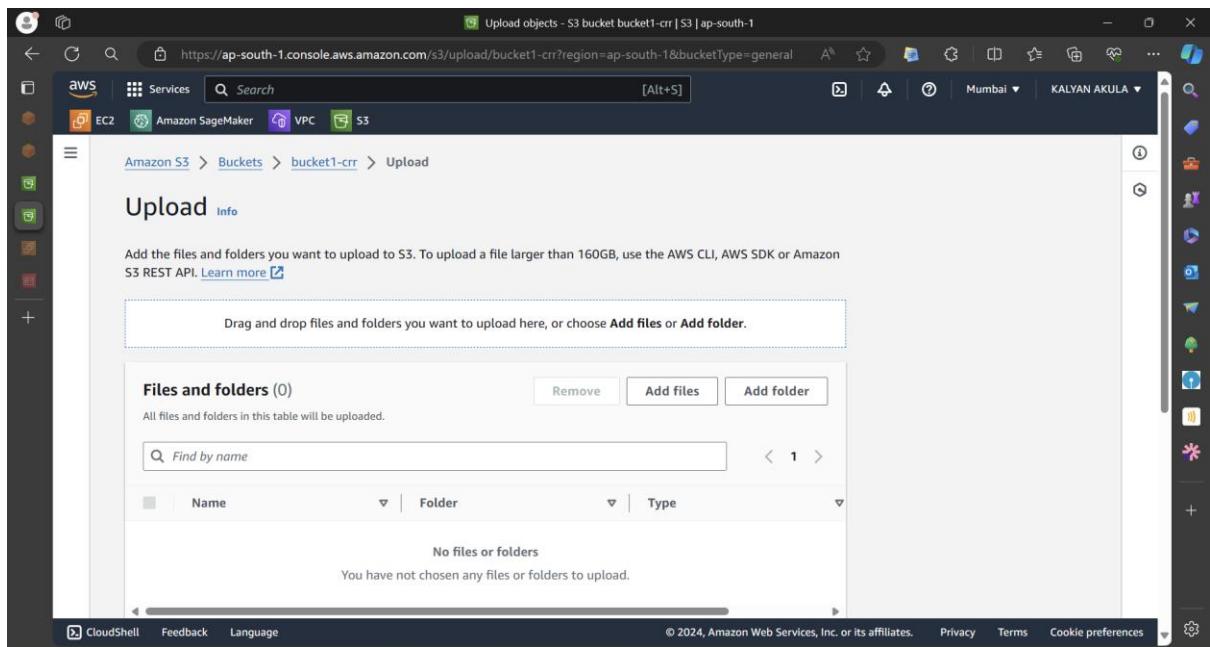
The screenshot shows the AWS S3 console interface. The left sidebar is titled 'Amazon S3' and includes sections for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings for this account, Storage Lens (Dashboards, Storage Lens groups, AWS Organizations settings), CloudShell, Feedback, and Language. The main content area is titled 'bucket1-crr' and shows 'Objects (1)'. A single object named 'info' is listed. The object has a size of 1.00 KB and was last modified on April 1, 2024. The 'Upload' button is highlighted in orange at the top of the object list.

**Step 11:** Now we need to upload a file in Bucket1-crr & it has to be reflect in Bucket2-crr

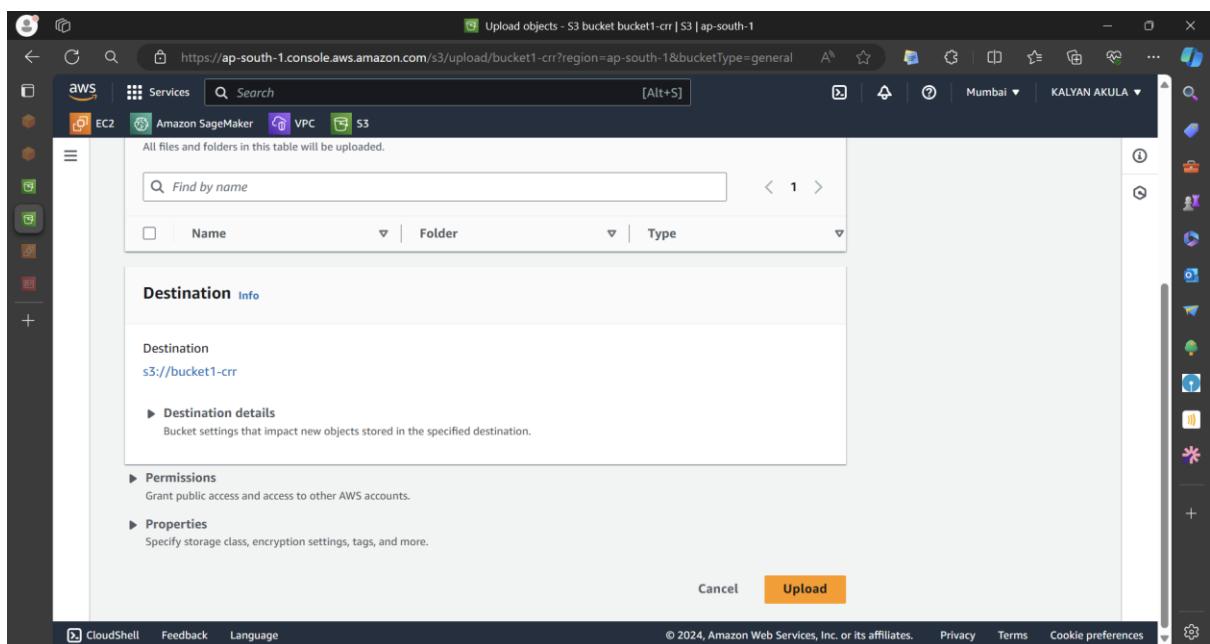
- Click on upload

This screenshot is identical to the one above, showing the AWS S3 console with the same interface and the 'info' object in Bucket1-crr. The 'Upload' button is again highlighted in orange.

- Click on Add Files



- Add a file & Scroll down , then click on upload



- Now it has to reflect in Bucket2-crr

The screenshot shows the AWS S3 console interface. On the left, the sidebar has 'Amazon S3' selected. The main area displays 'Objects (2) Info' with two items: '2435.jpg' (Type: jpg, Last modified: April 4, 2024, 05:37:14 (UTC+05:30), Size: 73.9 KB, Storage class: Standard) and 'rachna.com' (Type: com, Last modified: April 4, 2024, 04:20:31 (UTC+05:30), Size: 12.0 B, Storage class: Standard). At the top right, there are buttons for Actions, Create folder, and Upload, with 'Upload' being orange. A search bar and a 'Find objects by prefix' input field are also present.

- Here the file gets uploaded . So, the CRR is established

## Establishing Connection between VPC's & S3:Bucket1 in the same region

### Step 1: Go to EC2 Instance dashboard

The screenshot shows the AWS EC2 Instances dashboard. The sidebar has 'EC2 Dashboard' selected. The main area displays 'Resources' with the following counts: Instances (running) 2, Auto Scaling Groups 0, Dedicated Hosts 0, Elastic IPs 0, Instances 3, Key pairs 2, Load balancers 0, Placement groups 0, Security groups 7, Snapshots 0, and Volumes 2. Below this, there's a 'Launch instance' section with a 'Launch instance' button and a note: 'Note: Your instances will launch in the Asia Pacific (Mumbai) Region'. To the right, there's a 'Service health' section with a 'AWS Health Dashboard' button and a note: 'Region Asia Pacific (Mumbai) Status'. The URL at the bottom is https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#S3.

### Step 2 : Go to Instances

The screenshot shows the AWS EC2 Instances page. The left sidebar is expanded, showing the 'Instances' section with options like 'Instances', 'Instance Types', 'Launch Templates', etc. The main area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
inst2-vpc2	i-0500c5485c32ea288	Running	t2.micro	2/2 checks passed	View alarms
inst1-vpc1	i-0f07eceeба626b5a	Running	t2.micro	2/2 checks passed	View alarms

A modal window titled 'Select an instance' is open, listing the same two instances. The 'Actions' dropdown menu at the top right of the main table has 'Security' selected.

### Step 3: Select an instance & Select Actions

The screenshot shows the AWS EC2 Instance details page for 'inst1-vpc1'. The left sidebar is expanded, showing the 'Instances' section. The main area shows the instance details:

**Instance: i-0f07eceeба626b5a (inst1-vpc1)**

**Actions** ▾ **Launch instances** ▾

- Connect
- View details
- Manage instance state
- Instance settings
- Networking
- Security** ▾
- Image and templates
- Monitor and troubleshoot

The 'Security' option is highlighted in the dropdown. The 'Details' tab is selected in the bottom navigation bar.

### Step 4 : Click on Security & Modify IAM rule

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. Below that is the Images section with AMIs and AMI Catalog. The main area displays two instances: 'inst2-vpc2' and 'inst1-vpc1'. The instance 'inst1-vpc1' is selected. The Actions menu is open, and the 'Modify IAM role' option is highlighted. The instance details show it has a Public IPv4 address of 43.205.236.227 and is currently running.

### Step 5 : Click on Create new IAM role

The screenshot shows the 'Modify IAM role' dialog box. It starts with the instruction 'Attach an IAM role to your instance.' Below that is a section for 'Instance ID' with the value 'i-0f07eceeба626b5a (inst1-vpc1)'. The next section is 'IAM role', which contains a dropdown menu labeled 'Choose IAM role' and a button 'Create new IAM role'. A warning message in a box states: '⚠ If you choose No IAM Role, any IAM role that is currently attached to the instance will be removed. Are you sure you want to remove from the selected instance?' At the bottom are 'Cancel' and 'Update IAM role' buttons, with 'Update IAM role' being highlighted.

### Step 6 : Click on Create Role

The screenshot shows the AWS IAM Roles page. The left sidebar has 'Identity and Access Management (IAM)' selected under 'Access management'. The main area shows a table of roles:

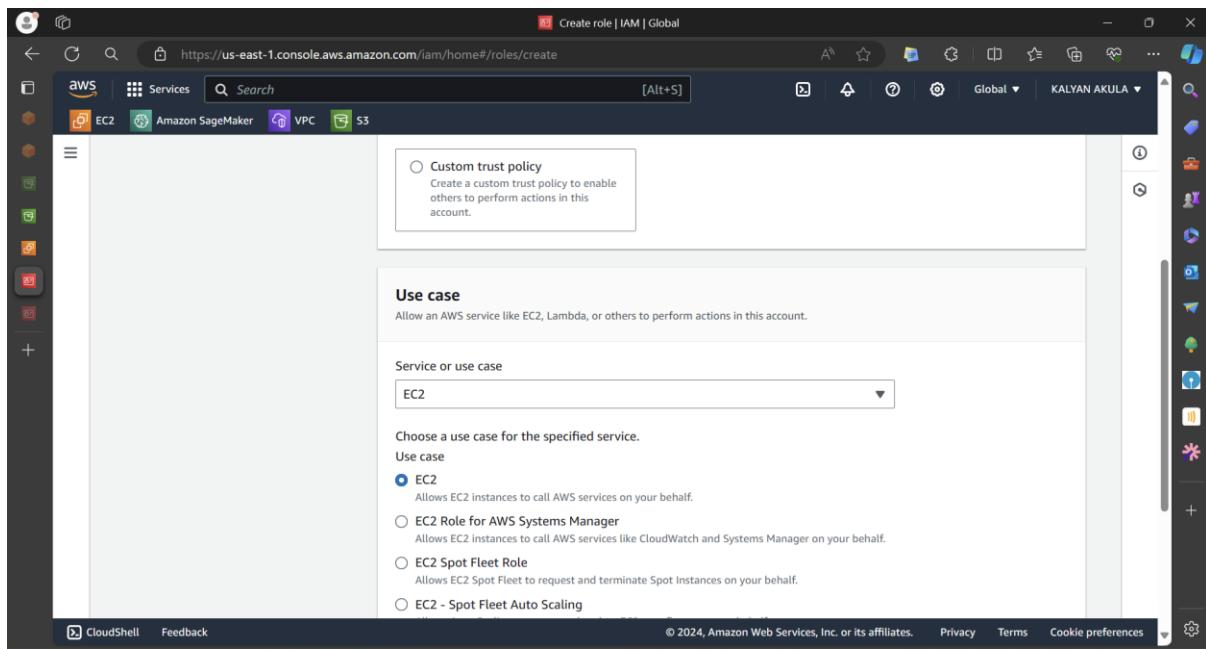
Role name	Trusted entities
AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)
ec2-role	AWS Service: ec2
s3crr_role_for_bucket1-crr	AWS Service: s3

Below the table is a section titled 'Roles Anywhere' with a 'Manage' button.

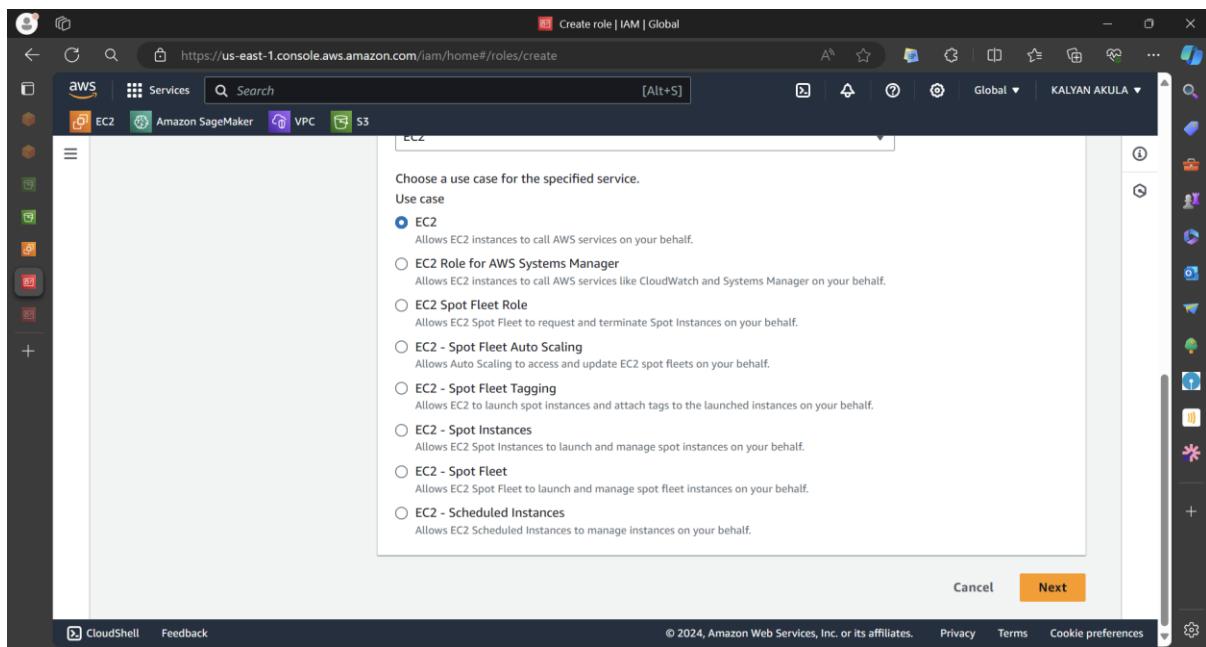
### Step 7 : Fill details as follows

The screenshot shows the 'Create role' wizard at Step 1: 'Select trusted entity'. The left sidebar shows steps 1 through 3. The main area is titled 'Select trusted entity' and shows the 'Trusted entity type' section with five options:

- AWS service: Allows AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account: Allows entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity: Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation: Allows users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy: Create a custom trust policy to enable others to perform actions in this account.



### Step 8 : Click on next



### Step 9 : Select Amazons3fullaccess

Add permissions Info

Permissions policies (1/922) Info

Choose one or more policies to attach to your new role.

Filter by Type

Policy name	Type	Description
AmazonDMSRedshiftFullAccess	AWS managed	Provides access to manage S3 settings ...
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	Provides full access to all buckets via t...
AmazonS3ObjectLambdaFullAccess	AWS managed	Provides AWS Lambda functions permis...
AmazonS3OutpostFullAccess	AWS managed	Provides full access to Amazon S3 on ...
AmazonS3OutpostReadAccess	AWS managed	Provides read only access to Amazon S...

### Step 10 : Click on Next

Create role | IAM | Global

Permissions policies

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	Provides full access to all buckets via t...
<input type="checkbox"/> AmazonS3ObjectLambdaFullAccess	AWS managed	Provides AWS Lambda functions permis...
<input type="checkbox"/> AmazonS3OutpostFullAccess	AWS managed	Provides full access to Amazon S3 on ...
<input type="checkbox"/> AmazonS3OutpostReadAccess	AWS managed	Provides read only access to Amazon S...
<input type="checkbox"/> AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all bucket...
<input type="checkbox"/> AWSBackupServiceFullAccess	AWS managed	Policy containing permissions necessar...
<input type="checkbox"/> AWSBackupServiceObjectLambdaAccess	AWS managed	Policy containing permissions necessar...
<input type="checkbox"/> QuickSightAccessForAWSLambda	AWS managed	Policy used by QuickSight team to acc...
<input type="checkbox"/> s3crr_for_bucket1-crr...	Customer managed	-

▶ Set permissions boundary - *optional*

Cancel Previous Next

### Step 11 : fill details as follows

The screenshot shows the 'Create role' wizard in the AWS IAM console. The current step is 'Step 1: Select trusted entities'. The 'Role details' section is visible, showing a 'Role name' field containing 'ec2-role' and a 'Description' field containing 'Allows EC2 instances to call AWS services on your behalf.' The sidebar on the left shows navigation links for IAM, Roles, and Create role.

The screenshot shows the 'Create role' wizard in the AWS IAM console, now at 'Step 2: Add permissions'. The 'Trust policy' section displays a JSON policy document:

```
1+ [
2+     "Version": "2012-10-17",
3+     "Statement": [
4+         {
5+             "Effect": "Allow",
6+             "Action": [
7+                 "sts:AssumeRole"
8+             ],
9+             "Principal": {
10+                 "Service": [
11+                     "ec2.amazonaws.com"
12+                 ]
13+             }
14+         }
15+     ]
16+ ]
```

The 'Permissions policy summary' section is also visible below the trust policy.

**Step 12 : click on create roll**

Permissions policy summary

Policy name	Type	Attached as
AmazonS3FullAccess	AWS managed	Permissions policy

**Step 3: Add tags**

Add tags - optional [Info](#)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

Cancel Previous **Create role**

**Step 13 :** Now go back to Modify IAM role tab

**Step 14 :** Select ec2-role

Modify IAM role [Info](#)

Attach an IAM role to your instance.

Instance ID  
i-0f07eceeба626b5a (inst1-vpc1)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.

ec2-role [Create new IAM role](#)

Cancel **Update IAM role**

**Step 15 :** Click on Update IAM role

**Step 16 :** Now Go to instance tab

The screenshot shows the AWS EC2 Instances page. The left sidebar has 'Instances' expanded, showing 'Instances' selected. The main table lists two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
inst2-vpc2	i-0500c5485c32ea288	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>
<input checked="" type="checkbox"/> inst1-vpc1	i-0f07eceeба626b5a	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>

The details for the selected instance (inst1-vpc1) are shown below:

IPv6 address	Instance state	Public IPv4 DNS
-	Running	-
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-10-0-0-7.ap-south-1.compute.internal	ip-10-0-0-7.ap-south-1.compute.internal	-
Answer private resource DNS name	Instance type	
-	t2.micro	-

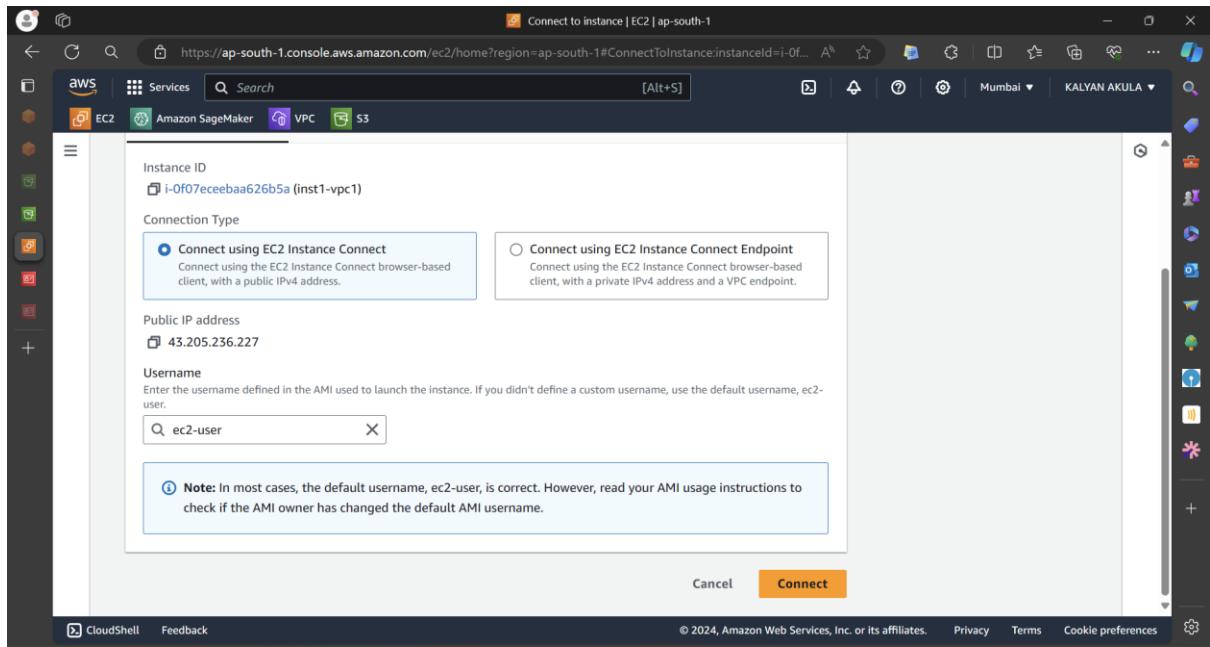
### Step 17 : Select inst1-vpc1 & Click on Connect

The screenshot shows the AWS EC2 Instance Details page for the selected instance (inst1-vpc1). The top navigation bar shows 'Instances | EC2 | ap-south-1'. The left sidebar has 'Instances' expanded, with 'Details' selected. The main content area shows the instance summary:

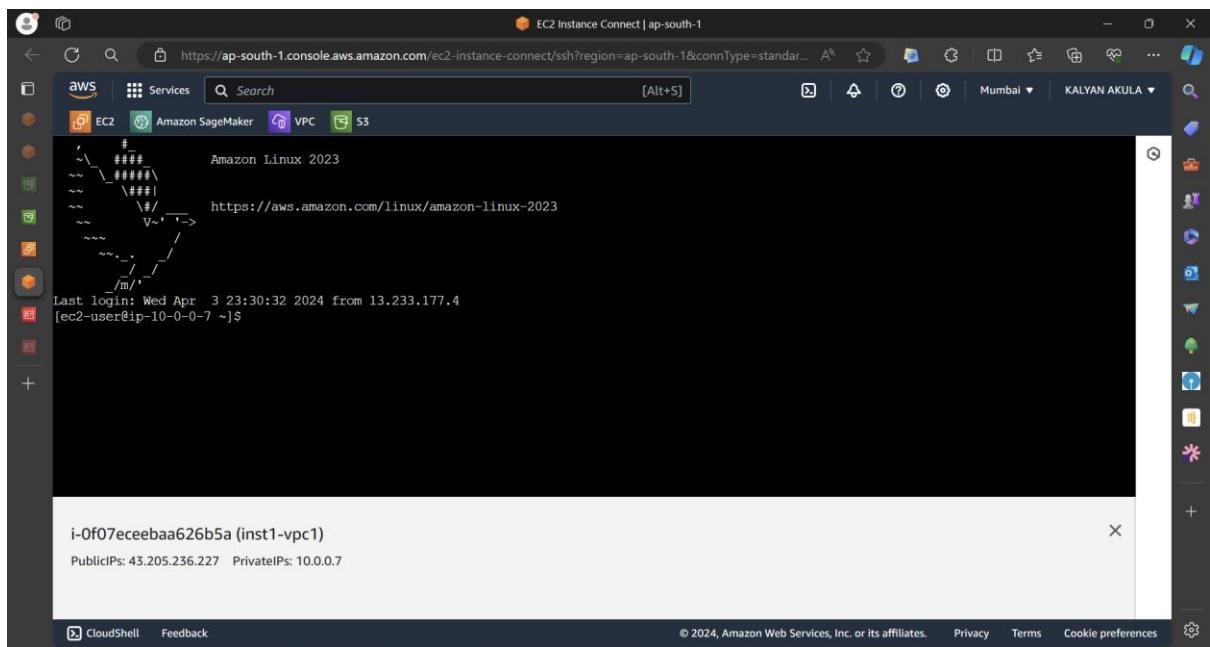
Instance ID	Public IPv4 address	Private IPv4 addresses
i-0f07eceeба626b5a (inst1-vpc1)	43.205.236.227 <a href="#">open address</a>	10.0.0.7
IPv6 address	Instance state	Public IPv4 DNS
-	Running	-

The 'Connect' button is located at the top right of the instance summary section.

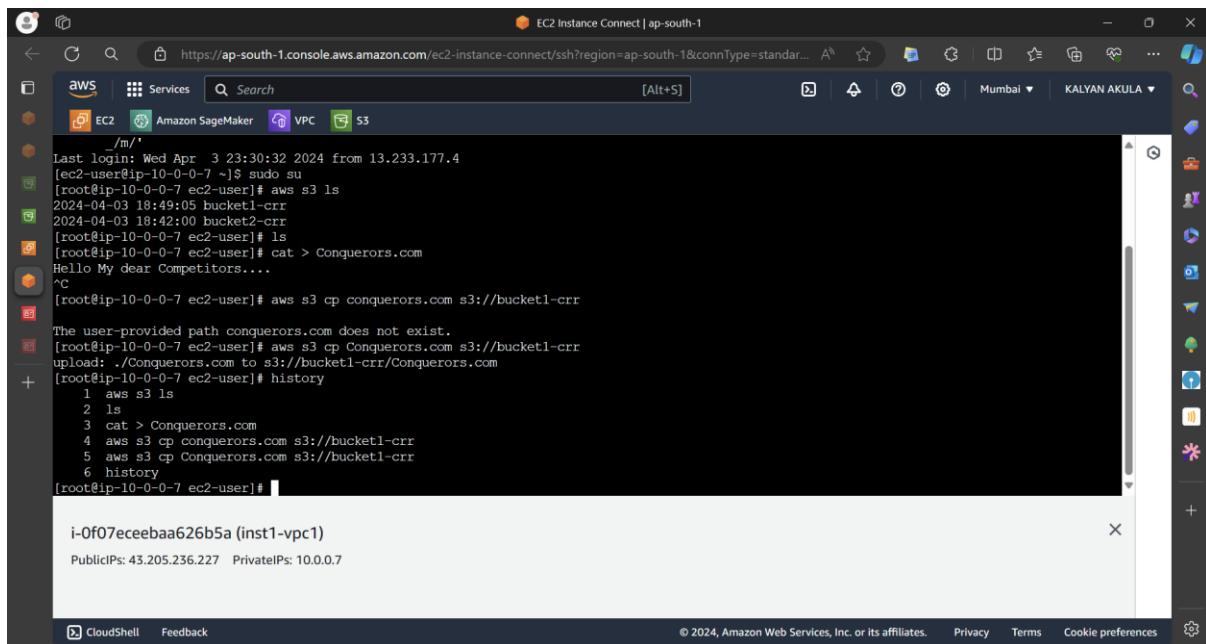
### Step 18 : Click on Connect



- A linux interface will open



**Step 19 :** By entering the following commands , we can upload files in S3:Bucket1-crr



The screenshot shows a terminal session on an EC2 instance. The user has run several commands to upload files from the instance to an S3 bucket named 'Bucket1-crr'. The session history is as follows:

```
Last login: Wed Apr  3 23:30:32 2024 from 13.233.177.4
[root@ip-10-0-0-7 ec2-user]# aws s3 ls
2024-04-03 18:49:05 bucket1-crr
2024-04-03 18:42:00 bucket2-crr
[root@ip-10-0-0-7 ec2-user]# ls
[root@ip-10-0-0-7 ec2-user]# cat > Conquerors.com
Hello My dear Competitors...
^C
[root@ip-10-0-0-7 ec2-user]# aws s3 cp conquerors.com s3://bucket1-crr
The user-provided path conquerors.com does not exist.
[root@ip-10-0-0-7 ec2-user]# aws s3 cp Conquerors.com s3://bucket1-crr
upload: ./Conquerors.com to s3://bucket1-crr/Conquerors.com
[root@ip-10-0-0-7 ec2-user]# history
 1 aws s3 ls
 2 ls
 3 cat > Conquerors.com
 4 aws s3 cp conquerors.com s3://bucket1-crr
 5 aws s3 cp Conquerors.com s3://bucket1-crr
 6 history
[root@ip-10-0-0-7 ec2-user]#
```

Below the terminal window, there is a message box containing:

i-0f07eceebaa626b5a (inst1-vpc1)  
Public IPs: 43.205.236.227 Private IPs: 10.0.0.7

```
[root@ip-10-0-0-7 ec2-user]# history
1 aws s3 ls
2 ls
3 cat > Conquerors.com
4 aws s3 cp conquerors.com s3://bucket1-crr
5 aws s3 cp Conquerors.com s3://bucket1-crr
6 history
+0 ip-10-0-0-7 ec2-user]#
```

## End Results

- By using this commands , We can upload data files from EC2 instance to the Bucket1-crr in the same region.
- We already established a CRR between Bucket1-crr & bucket2-crr , which are from different regions'
- So , if we upload files in Bucket1-crr through instance , it will also reflects in bucket2-crr

## BUCKET1-CRR

The screenshot shows the AWS S3 console interface for the 'bucket1-crr' bucket. The left sidebar has 'Amazon S3' selected under 'Buckets'. The main area displays three objects:

Name	Type	Last modified	Size	Storage class
(UTC+05:30)				
Conquerors.com	com	April 4, 2024, 06:13:59 (UTC+05:30)	30.0 B	Standard
rachna.com	com	April 4, 2024, 04:20:31 (UTC+05:30)	12.0 B	Standard

Actions buttons include Copy S3 URI, Copy URL, Download, Open, Delete, Create folder, and Upload.

## BUCKET2-CRR

The screenshot shows the AWS S3 console interface for the bucket2-crr bucket. The left sidebar includes links for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. The main content area displays the 'Objects' tab with three items listed:

Name	Type	Last modified	Size	Storage class
(UTC+05:30)				
Conquerors.com	com	April 4, 2024, 06:13:59 (UTC+05:30)	30.0 B	Standard
rachna.com	com	April 4, 2024, 04:20:31 (UTC+05:30)	12.0 B	Standard

Below the table, there is a note: "Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more".