

**DAY 16**

**DATE:17/05/2025**

**NAME: ANNIE JOHN**

**USER ID:27739**

**Batch: 25VID0885\_DC\_Batch4**

## **TITLE: AMAZON VPC (VIRTUAL PRIVATE CLOUD) AND CREATION OF VPC, INSTANCE IN EC2 AND S3 BUCKET**

### **➤ What is Amazon VPC?**

Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a logically isolated virtual network that you define. It closely mirrors a traditional on-premises network setup, while benefiting from the scalable infrastructure of AWS.

### **➤ Amazon VPC Concepts**

Amazon VPC serves as the **networking layer** for Amazon EC2 and other AWS services. Below are the core concepts and components associated with VPC:

#### **1. Virtual Private Cloud (VPC)**

A logically isolated section of the AWS cloud dedicated to your account where you can define your own virtual network topology, including IP address ranges, subnets, route tables, and network gateways.

#### **2. Subnet**

A subset of your VPC's IP address range where you can place groups of isolated resources. Subnets can be **public** (accessible from the internet) or **private** (not accessible directly from the internet).

#### **3. CIDR Block**

CIDR (Classless Inter-Domain Routing) block defines the IP address range for the VPC or subnet. For example, 10.0.0.0/16 provides 65,536 private IP addresses.

#### **4. Route Table**

A set of rules (routes) used to determine the path of outbound traffic from the subnets to other networks, such as the internet or other VPCs.

#### **5. DHCP Options Set**

Configuration options such as domain name, DNS servers, and NTP servers that are passed to instances launched in the VPC.

#### **6. Internet Gateway**

A horizontally scaled, redundant, and highly available gateway that enables communication between instances in your VPC and the internet.

#### **7. Egress-Only Internet Gateway**

Used with IPv6 traffic, this gateway allows outbound communication to the internet but blocks inbound traffic, enhancing security for IPv6-enabled instances.

#### **8. VPC Endpoint**

Enables private connectivity between your VPC and supported AWS services, without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect.

## **9. NAT Gateway**

A managed service that allows instances in a private subnet to initiate outbound connections to the internet while preventing inbound connections initiated by the internet.

## **10. NAT Instance**

An EC2 instance configured to provide the same function as a NAT Gateway. This is a more manual and less scalable alternative to the managed NAT Gateway.

## **11. Carrier Gateway**

Used in Wavelength Zones, it enables connectivity between your VPC and a telecommunications carrier network, supporting both inbound and outbound traffic.

## **12. Prefix Lists**

A set of one or more CIDR blocks that you can use as a reference in security group and route table configurations. These can also be shared across AWS accounts via Resource Access Manager (RAM).

## **13. Security Groups**

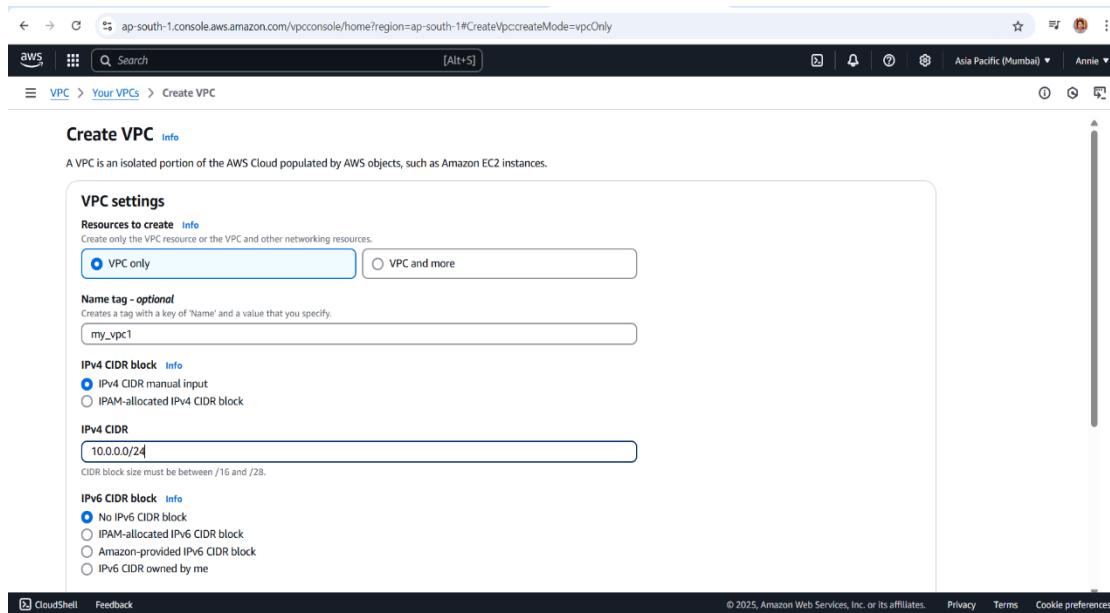
Virtual firewalls attached to EC2 instances that control inbound and outbound traffic. Security groups are stateful and apply at the instance level.

## **14. Network ACLs (Access Control Lists)**

An optional layer of stateless security at the subnet level that controls inbound and outbound traffic. Unlike security groups, NACLs are not stateful and can be used to explicitly allow or deny traffic.

## ➤ **Create your VPC**

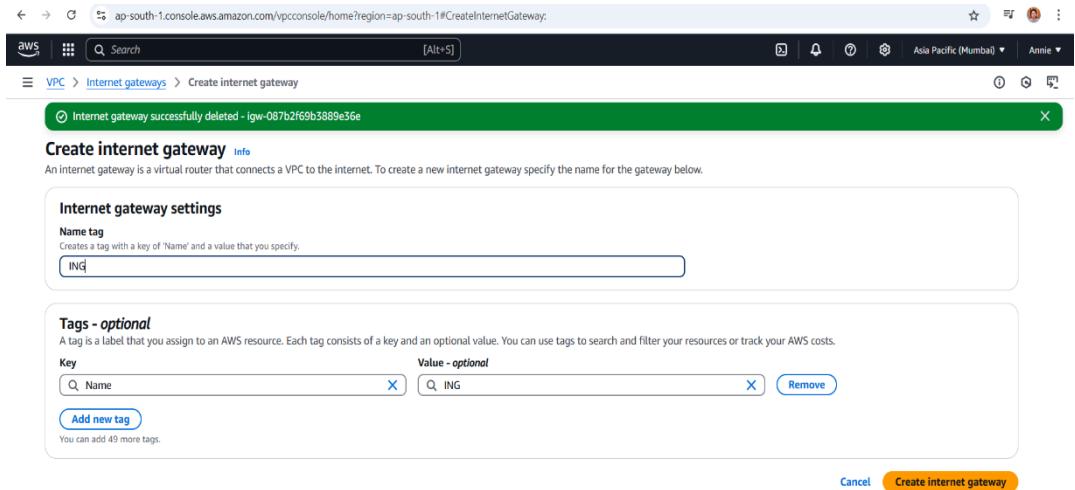
1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>
2. On the **VPC Dashboard**, choose **Create VPC**.  
Under **VPC Settings** select VPC only and give your VPC name and **IP CIDR block 10.0.0.0/24(IPV4)**.



### 3. Choose **Create VPC**.

### 4. Create and attach an **internet gateway** to a VPC

- An **Internet Gateway** is a horizontally scaled, redundant, and highly available VPC component that enables communication between instances in your VPC and the internet.
- Key Functions:
  - i. Route Target: Acts as a target in your route tables for traffic destined for the internet.
  - ii. NAT Functionality: Performs network address translation (NAT) for instances with public IPv4 addresses, allowing them to communicate with the internet.
  - iii. IPv6 Support: Fully supports both IPv4 and IPv6 traffic.
  - iv. No Extra Charges: There is no cost to create or use an internet gateway.
  - v. No Bottlenecks: It does not limit bandwidth or create availability risks.
- **Steps to Create and Attach an Internet Gateway**
  1. **Navigate to Internet Gateways->select Create Internet Gateway->Name the Internet Gateway (Optional)->Create the Internet Gateway->Click Create Internet Gateway.**



- **Attach the Internet Gateway to a VPC**

- Select the internet gateway you just created.
- Click **Actions → Attach to VPC**.

Name	Internet gateway ID	State	VPC ID
ING	igw-0184d47fc69e01792	Detached	-

- Choose your VPC from the dropdown list.
- Click **Attach Internet Gateway**.

The screenshot shows the AWS VPC console with the URL [ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#AttachInternetGateway:internetGatewayId=igw-0184d47fc69e01792](https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#AttachInternetGateway:internetGatewayId=igw-0184d47fc69e01792). The page title is "Attach to VPC (igw-0184d47fc69e01792)". A green banner at the top states: "The following internet gateway was created: igw-0184d47fc69e01792 - ING. You can now attach to a VPC to enable the VPC to communicate with the internet." Below this is a "Notifications" section with 2 notifications. A large button labeled "Attach to a VPC" is prominently displayed. The main content area is titled "VPC" and contains instructions: "Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below." It shows a search bar with the query "vpc-04eb04c18522796c" and a "Cancel" button. At the bottom right are "Cancel" and "Attach internet gateway" buttons.

## 5. Create Subnets

- **Steps to Create a Public Subnet**

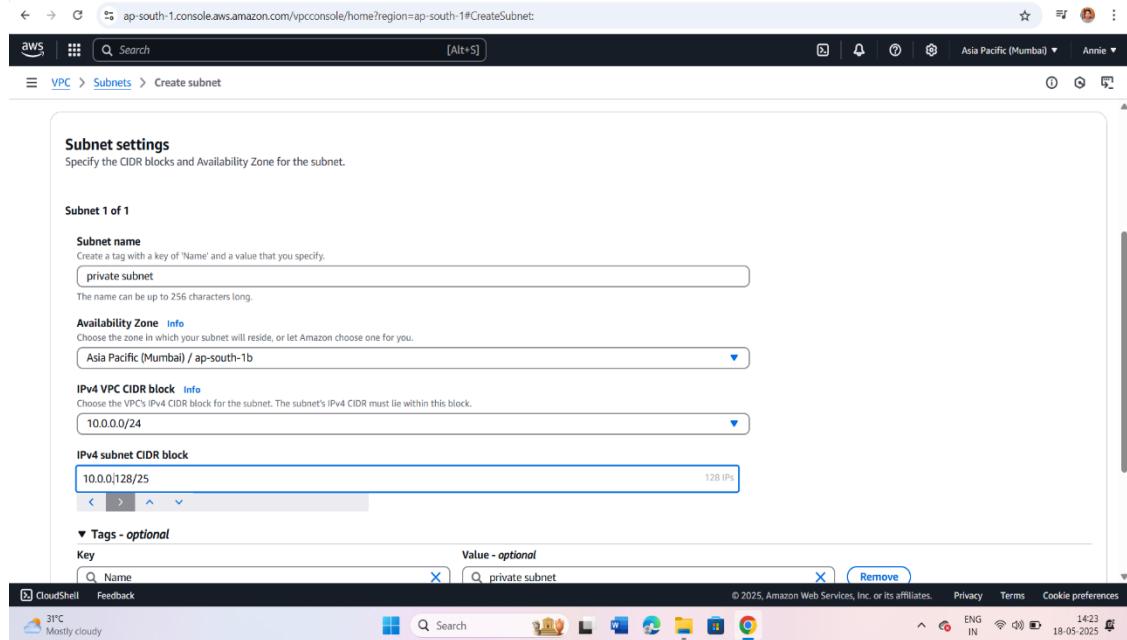
Go to Subnets ->Click Create Subnet->Enter Subnet Name->Choose VPC Select the same VPC you created earlier->Choose a Different Availability Zone)->Enter CIDR Block (eg:10.0.0.0/25 -dividing 256 ip to 128)->Click Create Subnet.

The screenshot shows the AWS VPC console with the URL [ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateSubnet](https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateSubnet). The page title is "Create subnet". The main form is titled "Subnet 1 of 1". It has fields for "Subnet name" (with placeholder "Create a tag with a key of 'Name' and a value that you specify." and input "public subnet"), "Availability Zone" (set to "Asia Pacific (Mumbai) / ap-south-1"), and "IPv4 VPC CIDR block" (set to "10.0.0.0/25"). Below these are sections for "Tags - optional" (with a key "Name" and value "public subnet") and "CloudShell" (with a link to "Feedback"). The status bar at the bottom shows "CloudShell Feedback" and various system icons.

- **Steps to Create a Private Subnet**

Go to Subnets ->Click Create Subnet->Enter Subnet Name->Choose VPC Select the same VPC you created earlier->Choose a Different Availability

Zone)->Enter CIDR Block (eg:10.0.0.128/25 -dividing 256 ip to 128)->Click Create Subnet.



## 6. Route tables:

A *route table* contains a set of rules, called *routes*, that are used to determine where network traffic from your subnet or gateway is directed.

The following are the key concepts for route tables.

- **Main route table**—The route table that automatically comes with your VPC. It controls the routing for all subnets that are not explicitly associated with any other route table.
- **Custom route table**—A route table that you create for your VPC.

### ✓ Create a custom route table

- a. Go to Route Tables → Click Create Route Table  
→ Enter a Name (e.g., Private Route Table) → Choose the same VPC → Click Create

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateRouteTable:

**Create route table** Info

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

**Route table settings**

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

**VPC**  
The VPC to use for this route table.

**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
<input type="text" value="Name"/>	<input type="text" value="private"/> <span>X</span>

**Add new tag**  
You can add 49 more tags.

Cancel Create route table



b. Here there will be one default route table name it as public.

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTables:

**Route tables (2) Info**

Last updated less than a minute ago

**Actions** Create route table

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
public	rtb-0fc0572be1481ac20	-	-	Yes	<a href="#">vpc-04eb04c18522796c</a>
private	rtb-01f6a5e5745a5a0ef	-	-	No	<a href="#">vpc-04eb04c18522796c</a>

**Select a route table**

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 31°C Mostly cloudy ENG IN 14:43 18-05-2025

c. Select the Route Table you just created(I choose public) → Go to Routes tab → Click Edit Routes → Click Add Route → Enter Destination: 0.0.0.0/0 → Target: Select Internet Gateway → Click Save Changes

The screenshot shows the AWS VPC console with the URL <https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRoutesRouteTableId=rtb-0fc0572be1481ac20>. The page title is "Edit routes". The main area shows a table with one row:

Destination	Target	Status	Propagated
10.0.0.0/24	local Internet Gateway	Active	No

Buttons at the bottom include "Add route", "Cancel", "Preview", and "Save changes". The status bar at the bottom right shows "CloudShell Feedback", "31°C Mostly cloudy", and the date "18-05-2025".

d. Go to Subnet Associations tab → Click Edit Subnet Associations

The screenshot shows the AWS VPC console with the URL <https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTables>. The left sidebar shows "Virtual private cloud" selected, with "Route tables" expanded. The main area shows the "Route tables (1/2) info" table:

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC
private	rtb-01f6a5e5745a5a0ef	-	-	No	vpc-04eab04c18522796c [m]
<input checked="" type="checkbox"/> public	rtb-0fc0572be1481ac20	-	-	Yes	vpc-04eab04c18522796c [m]

The "public" route table is selected. Below it, the "rtb-0fc0572be1481ac20 / public" details page is shown, with the "Subnet associations" tab selected. The "Explicit subnet associations (0)" section shows a table with columns: Name, Subnet ID, IPv4 CIDR, and IPv6 CIDR. A button "Edit subnet associations" is visible.

e. Select the Public Subnet → Click Save Associations

The screenshot shows the AWS VPC console with the URL [ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRouteTableSubnetAssociations:RouteTableId=rtb-0fc0572be1481ac20](https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRouteTableSubnetAssociations:RouteTableId=rtb-0fc0572be1481ac20). The page title is "Edit subnet associations". The "Available subnets (1/2)" section lists one public subnet: "public subnet" with Subnet ID "subnet-066e9145b286e9033" and IPv4 CIDR "10.0.0.0/25". The "Selected subnets" section also lists the same subnet. At the bottom right are "Cancel" and "Save associations" buttons.

f. Since we are keeping private route table as private, we only do subnet association for private net we will not edit route table.

The screenshot shows the AWS VPC dashboard with the URL [ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTables:sort=tagName](https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTables:sort=tagName). The left sidebar shows sections like VPC dashboard, EC2 Global View, Virtual private cloud, Route tables, Security, and PrivateLink and Lattice. The "Route tables" section shows two entries: "private" (selected) and "public". The "private" route table has Subnet ID "rtb-01f6a5e5745a5a0ef" and Main VPC "vpc-04eb04c18522796c". The "public" route table has Subnet ID "rtb-0fc0572be1481ac20" and Main VPC "vpc-04eb04c18522796c". Below the table, the "rtb-01f6a5e5745a5a0ef / private" details page is shown, with the "Subnet associations" tab selected. It shows "Explicit subnet associations (0)". At the bottom right of the screenshot, there is a status bar indicating "Rain coming in about 1.5 hours".

The screenshot shows the AWS VPC console with the URL [ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRouteTableSubnetAssociations:RouteTableId=rtb-01f6a5e5745a5a0ef](https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRouteTableSubnetAssociations:RouteTableId=rtb-01f6a5e5745a5a0ef). The page title is "Edit subnet associations". It displays a table of available subnets and a list of selected subnets. The selected subnet is "private subnet" (subnet-04eda5a3dabb11146).

Available subnets (1/2)					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
public subnet	subnet-066e9145b286e9033	10.0.0.0/25	-	rtb-0fc0572be1481ac20 / public	
<input checked="" type="checkbox"/> private subnet	subnet-04eda5a3dabb11146	10.0.0.128/25	-	Main (rtb-0fc0572be1481ac20 / public)	

**Selected subnets**

- subnet-04eda5a3dabb11146 / private subnet X

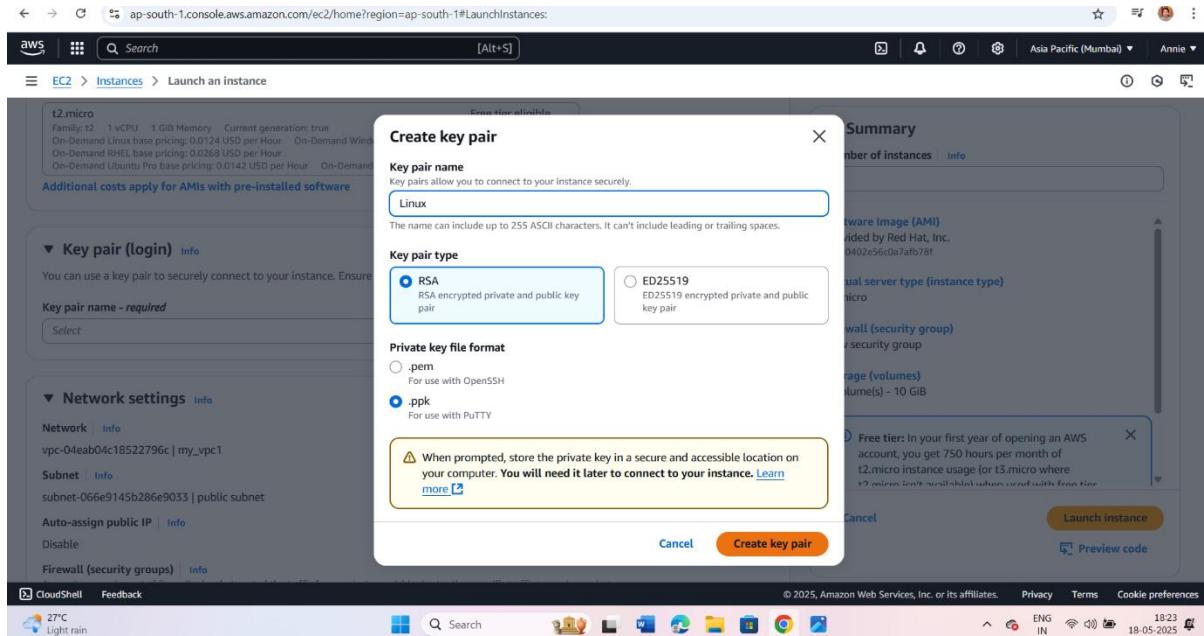
Buttons: Cancel, Save associations.

- **EC2 Instance (Amazon Elastic Compute Cloud):**  
An EC2 instance is a virtual server in the AWS cloud used to run applications, host websites, store data, or perform computations — just like a physical computer, but scalable, flexible, and pay-as-you-go.
- **Instance creation in EC2**

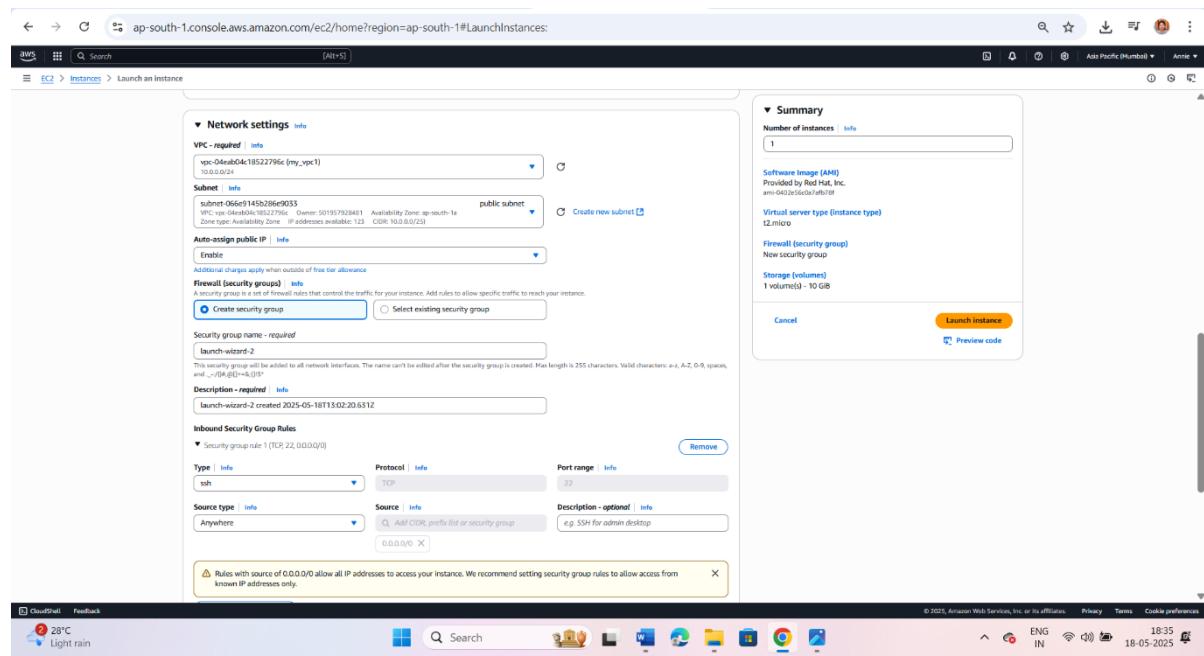
1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. Click on **Launch instance** for creating an instance->give name ->select OS image(choose Red Hat).

The screenshot shows the AWS EC2 console with the URL [ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:](https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:). The page title is "Launch an instance". It shows the "Launch an instance" wizard, step 1: "Launch an instance". Step 2: "Application and OS Images (Amazon Machine Image)". It lists various AMIs, including "Red Hat Enterprise Linux 9 (x86\_64), SSD Volume Type". Step 3: "Summary". It shows the number of instances (1), software image (Red Hat, Inc. ami-0402e56ca7a7bf7bf), virtual server type (t2.micro), and storage (1 volume(s) - 10 GiB). Buttons: "Launch instance", "Preview code".

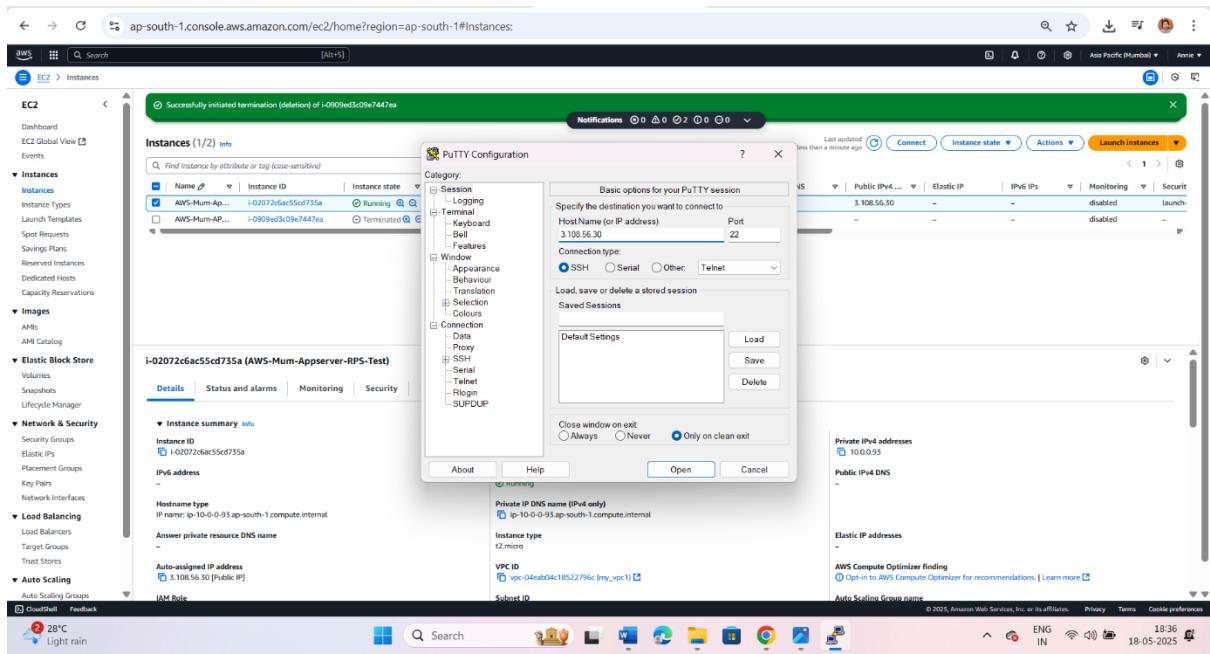
- Create a key pair choose RSA, give name for key pair( since Linux is chosen select file format ‘.ppk’).



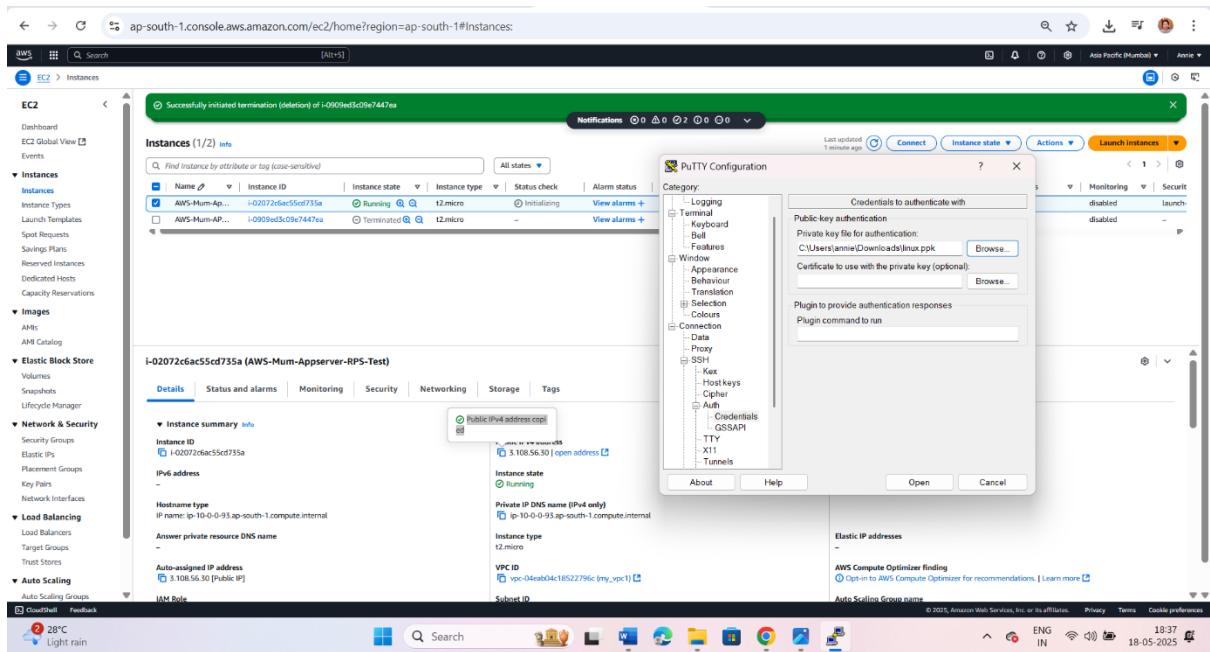
- Edit Network Settings->Enable auto-assign public IP->create security group use ssh.



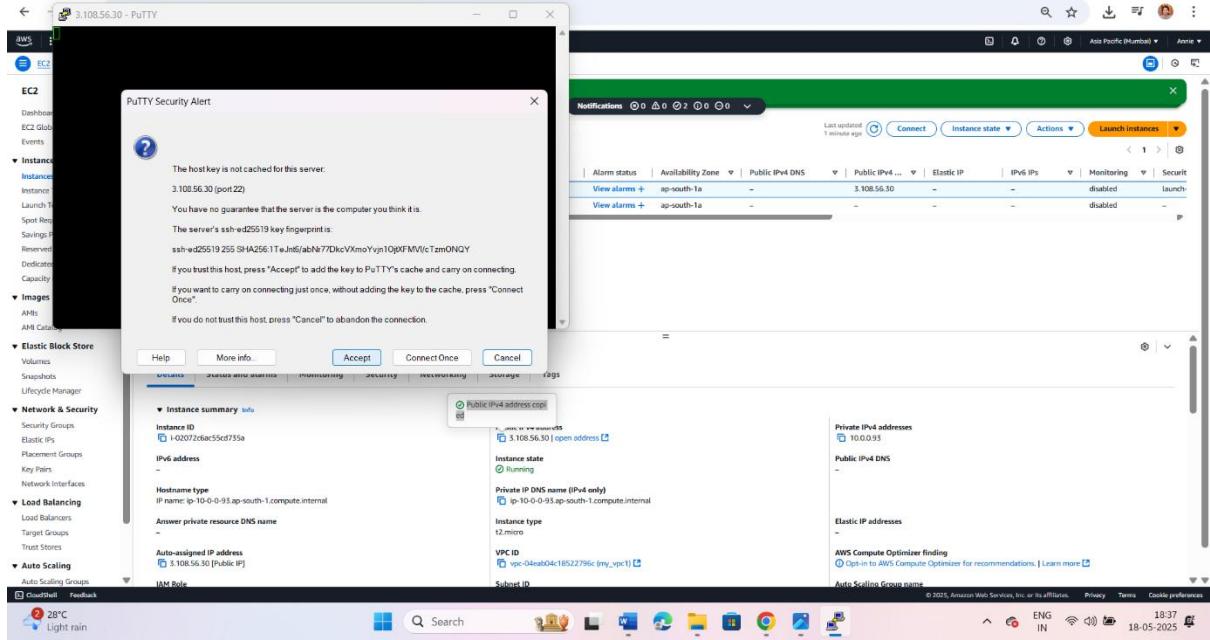
### 3. Created instance->open PuTTY->give public IP.



#### 4. In SSH->Auth->Credentials->Browse the downloaded key pair.



#### 5. Click Open->accept.



## 6. Give your username :-ec2-user

```
ec2-user@ip-10-0-0-93:~$ 
[ec2-user@ip-10-0-0-93 ~]$ login as: ec2-user
[ec2-user@ip-10-0-0-93 ~]$ Authenticating with public key "linux"
Register this system with Red Hat Insights: rhc connect
Example:
# rhc connect --activation-key <key> --organization <org>
The rhc client and Red Hat Insights will enable analytics and additional
management capabilities on your system.
View your connected systems at https://console.redhat.com/insights

You can learn more about how to register your system
using rhc at https://red.ht/registration
[ec2-user@ip-10-0-0-93 ~]$ 
```

## 7. Successfully logged into your **EC2 instance** using SSH.

**This Means:**

- You connected to the EC2 instance as the user: ec2-user
- You're using a **Linux-based EC2 instance** (Red Hat)
- The login used **public key authentication** ("linux" key pair)
- You're now at the **Linux command-line terminal** of your EC2 server.

## ➤ Amazon S3 (Simple Storage Service)

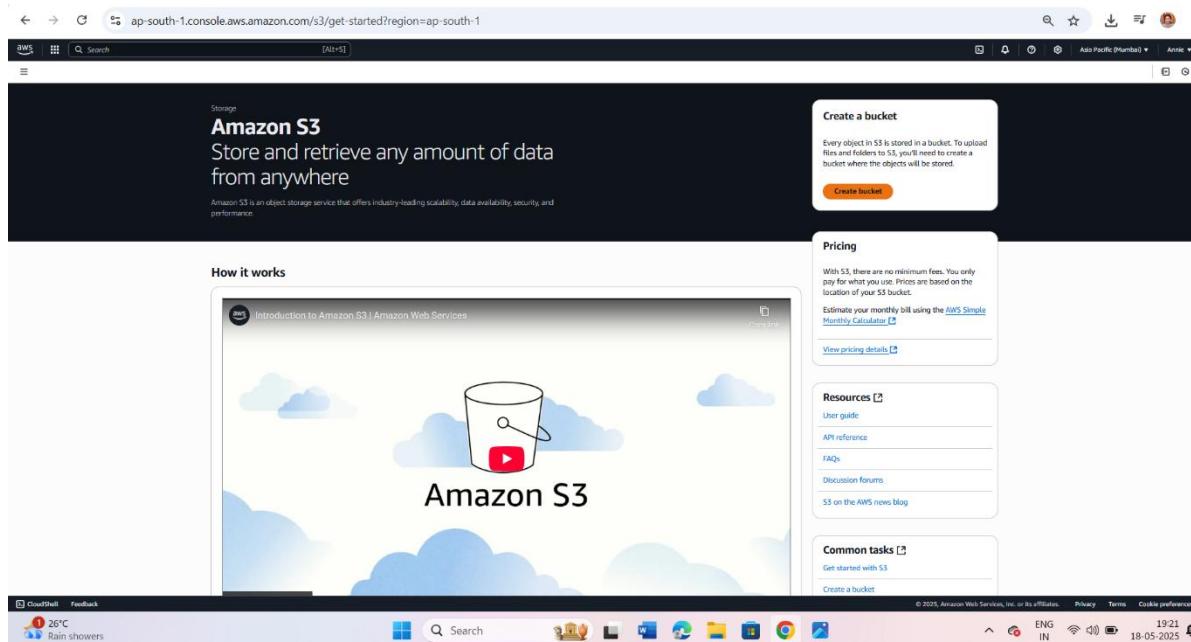
**Amazon S3** is a scalable, durable cloud storage service offered by AWS. You use it to store and retrieve any amount of data, at any time, from anywhere on the web.

### • What Can You Store in S3?

- a. Documents, images, videos
- b. Website files (HTML, CSS, JS)
- c. Backups and logs
- d. Data for analytics
- e. Application data (mobile/web apps)

## ➤ Creating S3

- Open Amazon S3 console.
- Click on create bucket.



- Give a bucket name should be global one->change object ownership to ACL enabled->uncheck the block public access ->create bucket.

**Object Ownership**

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 to new buckets and objects you create. Note that if 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](#)

**Block all public access**  
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

**Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly created 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.

- Created bucket, it was created on location Mumbai but I changed AWS region to Virginia.

Name	AWS Region	IAM Access Analyzer	Creation date
testbucketwipro	Asia Pacific (Mumbai) ap-south-1	<a href="#">View analyzer for ap-south-1</a>	May 18, 2025, 19:39:59 (UTC+05:30)

- Click on bucket name->upload some documents.

The screenshot shows the AWS S3 'Upload' interface. At the top, it says 'Upload info'. Below that, there's a note about uploading files larger than 160GB. A large dashed box allows dragging and dropping files. A table lists four files: '4.Installation and Configure IIS Web Server On Wind...' (application/pdf, 875.2 KB), '5.Configure NTFS permissions on a shared folder.pdf' (application/pdf, 1.2 MB), 'Screenshot 2025-05-07 203634.png' (image/png, 314.5 KB), and 'Screenshot 2025-05-09 105645.png' (image/png, 449.8 KB). Below the table, the 'Destination' section shows the path '/testbucketwipro'. Under 'Properties', there are sections for 'Permissions' and 'Properties'. At the bottom right are 'Cancel' and 'Upload' buttons.

- Copy the object URL of one document and open it from browser we will get the error.

This screenshot shows a browser window displaying an XML error page. The URL is 'testbucketwipro.s3.ap-south-1.amazonaws.com/4.Installation+and+Configure+IIS+Web+Server+On+Windows+Server+2019.pdf'. The page content includes:

```

<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>202F0K5ABC0MWNBTB</RequestId>
  <HostId>P8Y1l2IVcqXuYNs7C0XBFrBbUte9Y0W0WJ02UvEFlgcZ12idU11HVmrmPUyNg+gqewA1w+GeY=</HostId>
</Error>

```

## 7. Giving file permissions

- So, the files don't have permission for public we give permission **individually** by clicking any one of file ->click the permission give object permission for read and save changes.

- Again, refresh the browser, the file will be visible.

- ii. File permission using policy in Jason format(This JSON policy means: "**Allow everyone in the world to read (download) any file inside this S3 bucket.**")
- Go to permissions in bucket.

Name	Type	Last modified	Size	Storage class
Installation and Configure IIS Web...	pdf	May 18, 2025, 19:42:31 (UTC+05:30)	873.7 kB	Standard
Share On Windows Server 2019.pdf	pdf	May 18, 2025, 19:42:32 (UTC+05:30)	1.2 MB	Standard
Screenshot 2025-05-07 200804.png	png	May 18, 2025, 19:42:32 (UTC+05:30)	314.3 kB	Standard
Screenshot 2025-05-09 105645.png	png	May 18, 2025, 19:42:33 (UTC+05:30)	449.8 kB	Standard

- See the bucket policy is empty.

**Bucket policy**  
The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

No policy to display.

**Object Ownership**  
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.

**Object Ownership**  
Bucket owner preferred  
ACLs are retained and can be used to grant access to this bucket and its objects. If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

**Access control list (ACL)**  
Grant basic read/write permissions to other AWS accounts. [Learn more](#)

The console displays combined access grants for duplicate grantees  
To see the full list of ACLs, use the Amazon S3 REST API, AWS CLI, or AWS SDKs.

- Go to policy generator->Select policy type as **S3 bucket policy**  
 ->Effect allow->principal as \*(everyone)->Select checkbox all actions ->paste the ARN(**A Bucket ARN** (Amazon Resource Name) uniquely identifies an **Amazon S3 bucket**) copied from edit bucket policy->add statement->generate policy->copy it.

[aws policymaker.s3.amazonaws.com/policygen.html](https://aws policymaker.s3.amazonaws.com/policygen.html)

### AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#).

#### Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an [IAM Policy](#), an [S3 Bucket Policy](#), an [SNS Topic Policy](#), a [VPC Endpoint Policy](#), and an [SQS Queue Policy](#).

Select Type of Policy: S3 Bucket Policy

#### Step 2: Add Statement(s)

A statement is the formal description of a single permission. See [a description of elements](#) that you can use in statements.

Effect: Allow (radio button selected)

Principal:

AWS Service: Amazon S3 (dropdown menu)

Actions: All Actions (\*) (checkbox checked)

Amazon Resource Name (ARN): arn:aws:s3:::testbucketwipro

Add Conditions (Optional)

**Add Statement**

#### Step 3: Generate Policy



- Paste it over edit policy->save.

[ap-south-1.console.aws.amazon.com/s3/bucket/testbucketwipro/property/policy/edit?region=ap-south-1&bucketType=general](https://ap-south-1.console.aws.amazon.com/s3/bucket/testbucketwipro/property/policy/edit?region=ap-south-1&bucketType=general)

Amazon S3 > Buckets > testbucketwipro > Edit bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

Bucket ARN: arn:aws:s3:::testbucketwipro

Policy:

```

1 * {
2   "Id": "P0110c1747579129441",
3   "Version": "2012-10-17",
4   "Statement": [
5     {
6       "Sid": "Set12345678901234567853",
7       "Action": "s3:*",
8       "Effect": "Allow",
9       "Resource": "arn:aws:s3:::testbucketwipro*",
10      "Principal": "*"
11    }
12  ]
13 }
```

**Edit statement**

Select a statement  
Select an existing statement in the policy or add a new statement.

+ Add new statement

Preview external access

Cancel Save changes

- Open a file and check again get an error.



- Again, in edit policy add a change then only public access is allowed for all files(in resources ARN give /\*).

```
{ "Version": "2012-10-17", "Id": "Policy1747579317651", "Statement": [ { "Sid": "Stmt1747579317651", "Effect": "Allow", "Principal": "*", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::testbucketwipro/*" } ] }
```