Placement Empowerment Program

Cloud Computing and DevOps Centre

Set a private network in cloud – Create a VPC with subnets for your instances. Configure routing for internal communication between subnets

Name :Nantha Krishnan.G.K.
Department: AML

St. JOSEPH'S COLLEGE OF ENGINEERING | St. JOSEPH'S INSTITUTE OF TECHNOLOGY
AUTONOMOUS INSTITUTIONS, AFFILIATED TO ANNA UNIVERSITY

Introduction

A Virtual Private Cloud (VPC) is a secure and isolated portion of a cloud provider's infrastructure where you can deploy your resources in a controlled environment. Setting up a VPC involves creating subnets, configuring routing, and implementing security measures to manage traffic and access. This setup is essential for applications that require secure internal communication while being accessible to external networks when necessary.

Objectives

1. **Create a VPC**: Establish a private network in the cloud that suits your application requirements.
2. **Configure Subnets**: Design and implement subnets within the VPC for different types of instances (e.g., public and private).
3. **Set Up Routing**: Configure routing tables to enable internal communication between subnets and external access as required.
4. **Implement Security**: Use security groups and network ACLs to control inbound and outbound traffic to your instances.
5. **Ensure High Availability**: Distribute resources across multiple Availability Zones to enhance resilience

Importance

- Security: A VPC allows you to maintain a secure environment, isolating your resources from public internet exposure while enabling controlled access.
- Customization: You can tailor the network architecture to meet specific needs, such as private IP addressing and subnetwork segmentation.
- Cost Efficiency: Efficiently using cloud resources helps in managing costs associated with data transfer and resource allocation.
- Scalability: Easily scale your infrastructure to accommodate growing workloads without compromising security or performance.
- Control: Gain complete control over the networking environment, including IP address ranges, routing, and access controls.

Step-by-Step Overview

Step 1:

1. Go to AWS Management Console.
2. Enter your username and password to log in

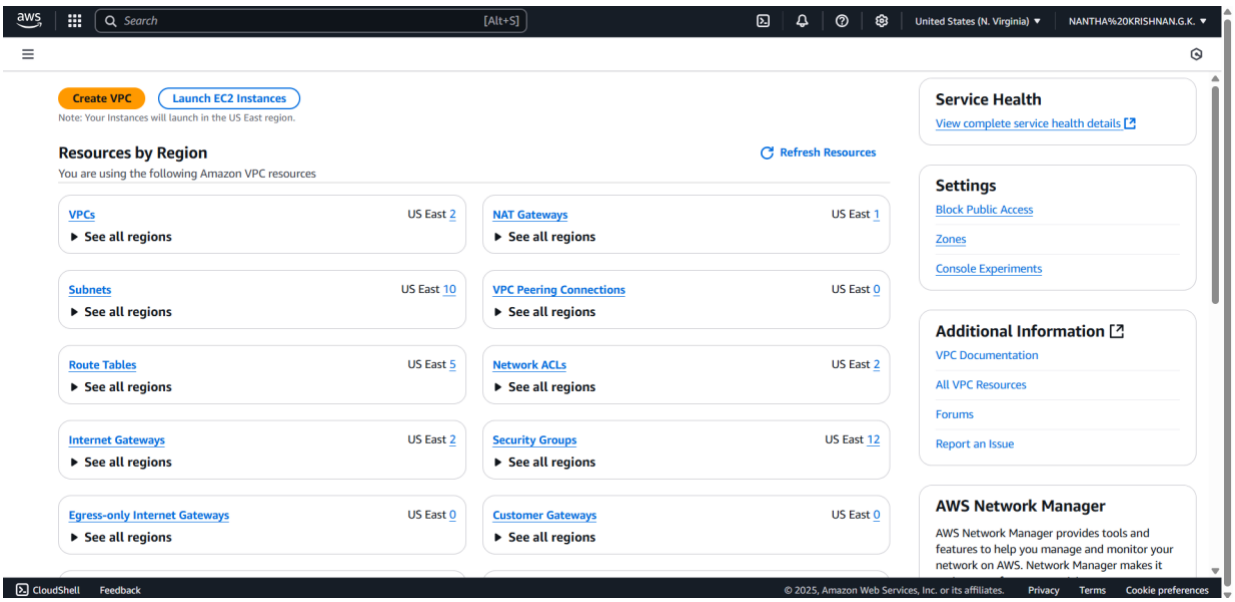
Step 2:

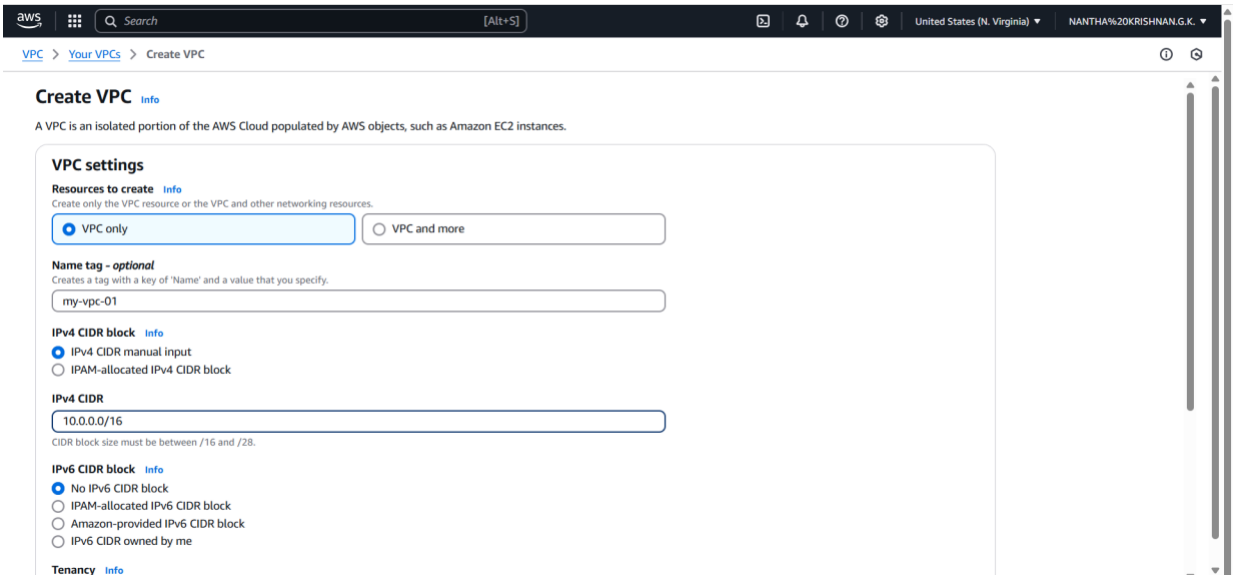
Navigate to the VPC Dashboard

- In the Services menu, select "VPC" to access the VPC Dashboard.
-

Create a VPC

- Click on "Your VPCs" in the left menu, then click "Create VPC."
- Specify the following:
  - **Name tag**: A name for your VPC.
  - **IPv4 CIDR block**: E.g., 10.0.0.0/16 (this gives you 65,536 IP addresses).
  - **IPv6 CIDR block**: (Optional).
  - **Tenancy**: Default is usually sufficient.
- Click "Create."





Step 3:

Create Subnets

You need at least two private subnets for internal communication:

1. Go to Subnets → Click Create Subnet.

2. Select the VPC (MyPrivateVPC) you created earlier.

3. Create two subnets:

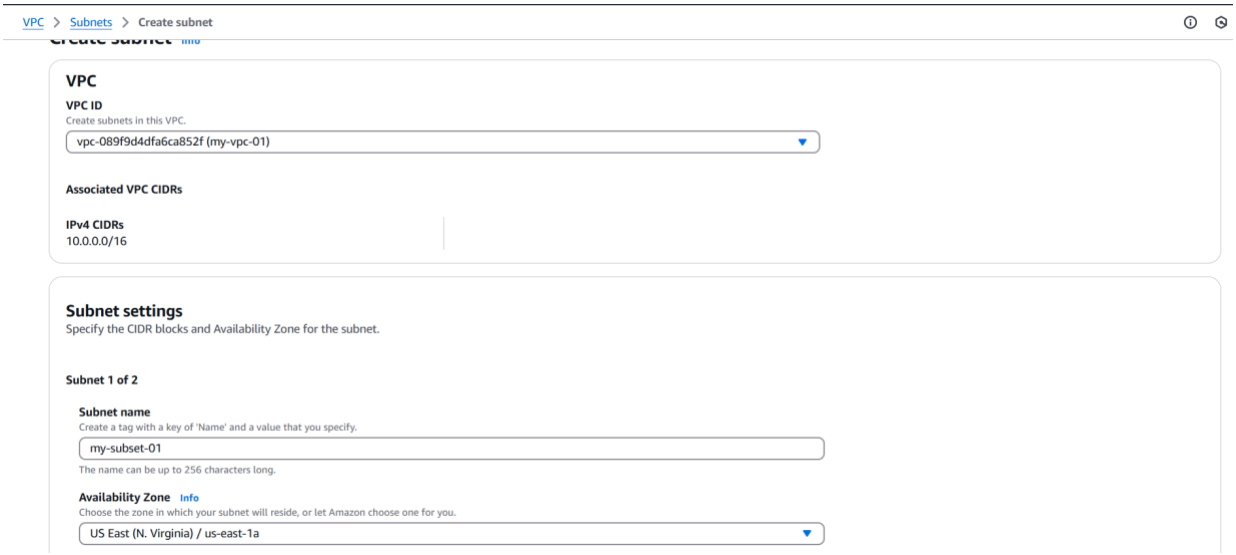
Subnet 1 (Private-Subnet-A)

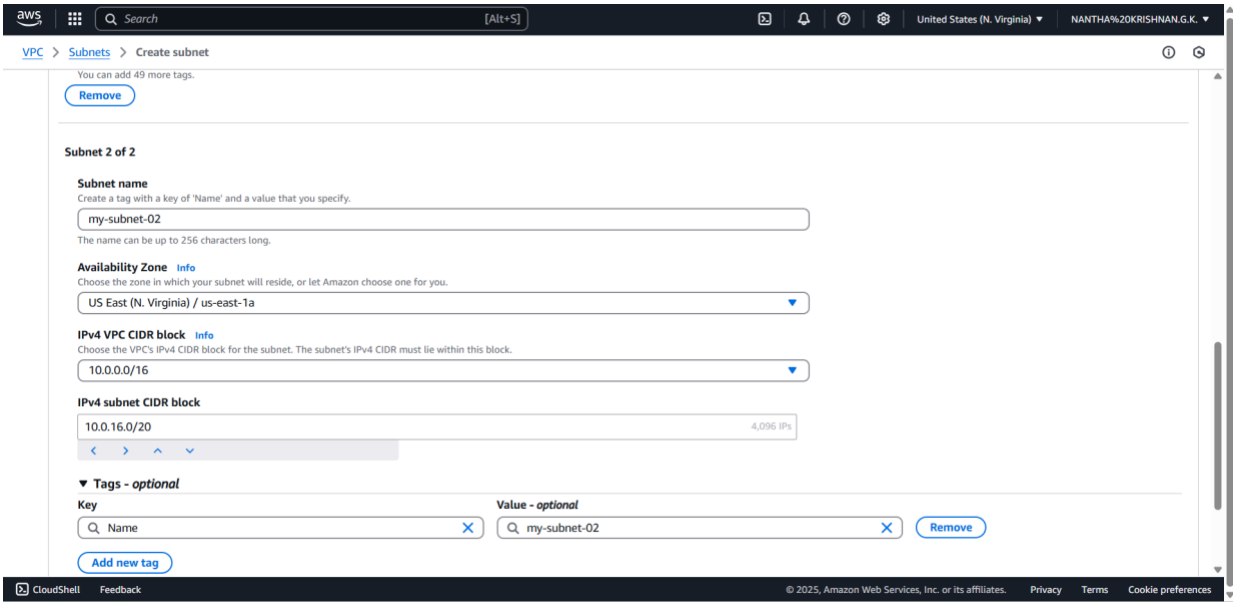
IPv4 CIDR: 10.0.1.0/24

Availability Zone: us-east-1a (example)

Subnet 2 (Private-Subnet-B)

IPv4 CIDR: 10.0.2.0/24


VPC > Subnets > Create subnet
VPC
VPC ID
Create subnets in this VPC.
vpc-089f9d4dfa6ca852f (my-vpc-01)
Associated VPC CIDRs
IPv4 CIDRs
10.0.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 2
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
my-subset-01
The name can be up to 256 characters long.
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a


aws
Search
[Alt+S]
United States (N. Virginia)
NANTHA%20KRISHNAN.G.K.
VPC > Subnets > Create subnet
You can add 49 more tags.
Remove
Subnet 2 of 2
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
my-subnet-02
The name can be up to 256 characters long.
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16
IPv4 subnet CIDR block
10.0.16.0/20
4,096 IPs
Tags - optional
Key
Value - optional
Name
my-subnet-02
Remove
Add new tag
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 4:

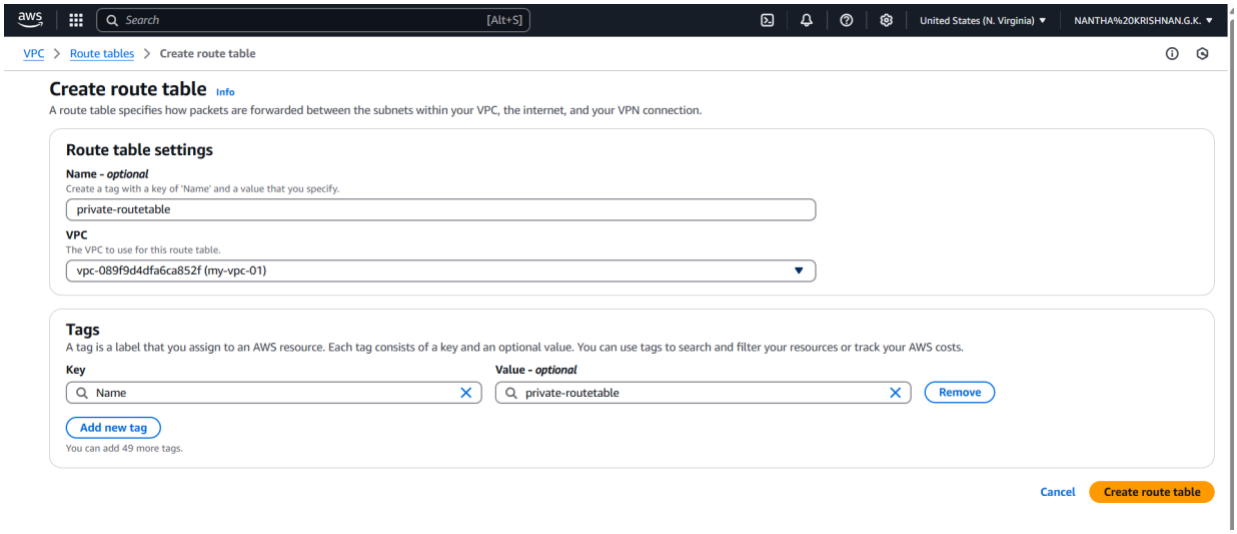
Configure Route Tables for Internal Communication

1. Go to Route Tables → Click Create Route Table.

2. Name it (e.g., PrivateRouteTable).

3. Select MyPrivateVPC.

4. Click Create.



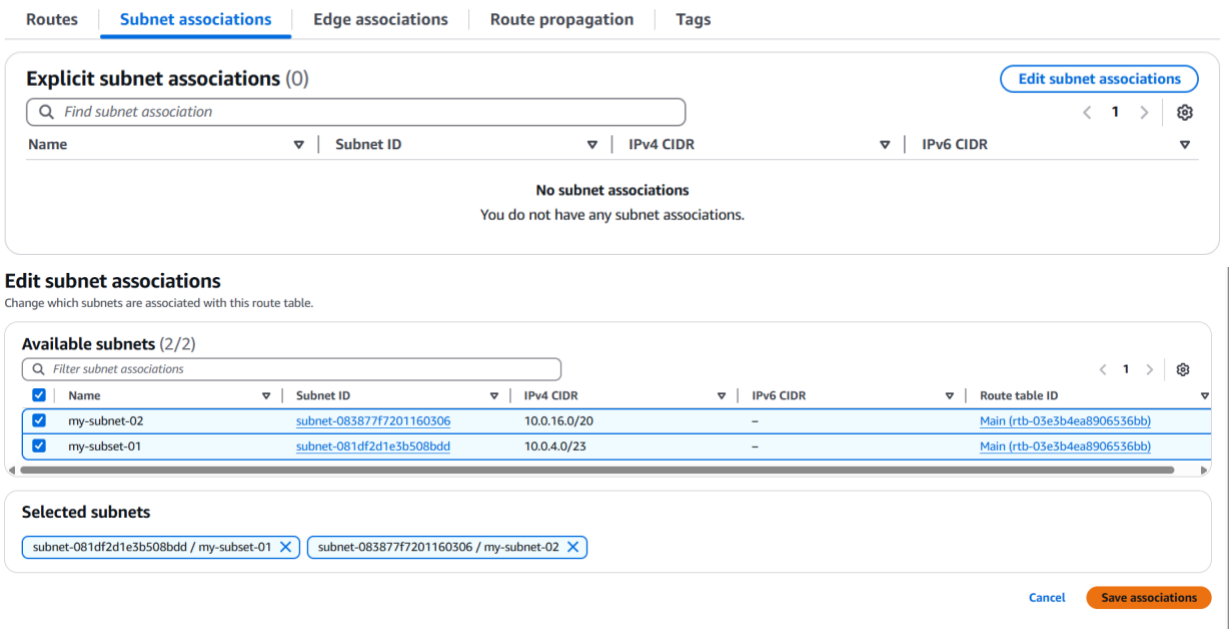
Step 5:

Associate the subnets:

Go to Subnet Associations → Click Edit subnet associations.

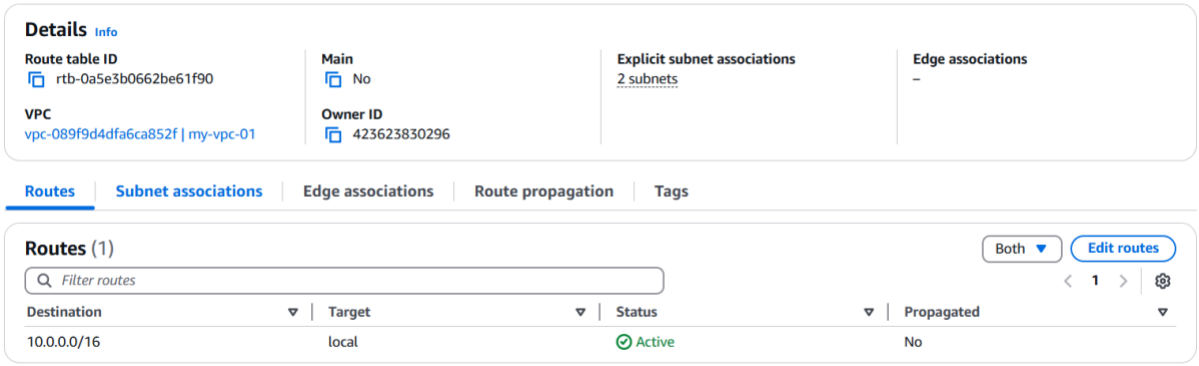
Select Private-Subnet-A and Private-Subnet-B.

Click Save associations.



Step 6:

Default route: 10.0.0.0/16 → local (Automatically added).



Step 7:

Launch Instances in Private Subnets

1. Go to EC2 Dashboard → Launch Instance.

2. Select an AMI (Amazon Linux, Ubuntu, etc.).

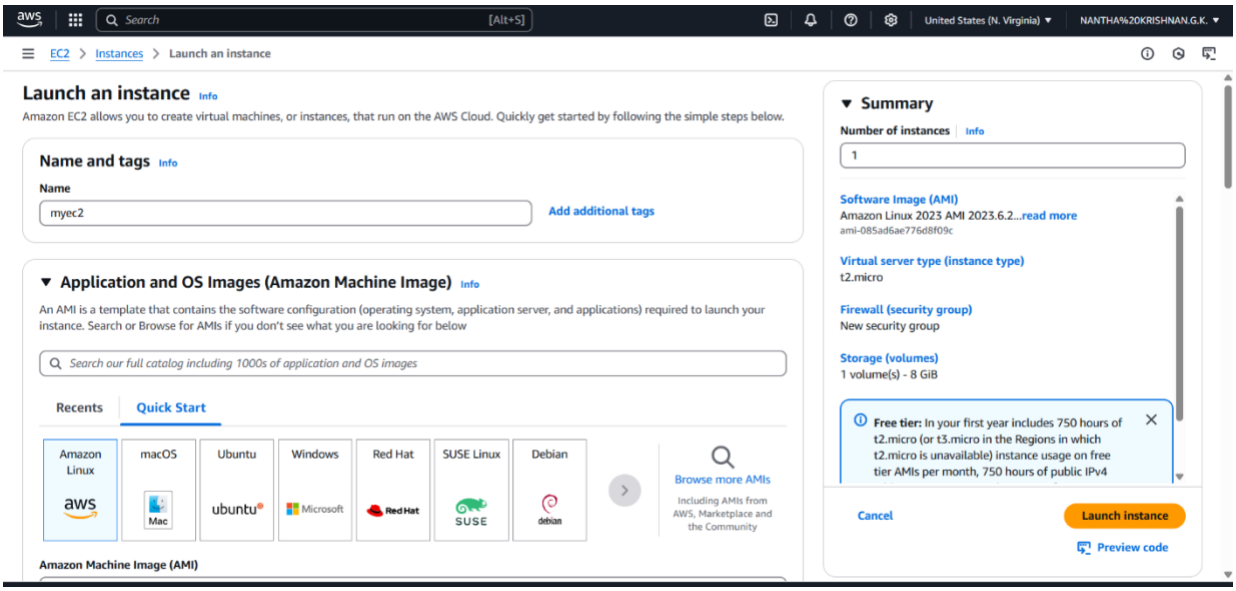
3. Choose an Instance Type (e.g., t2.micro).

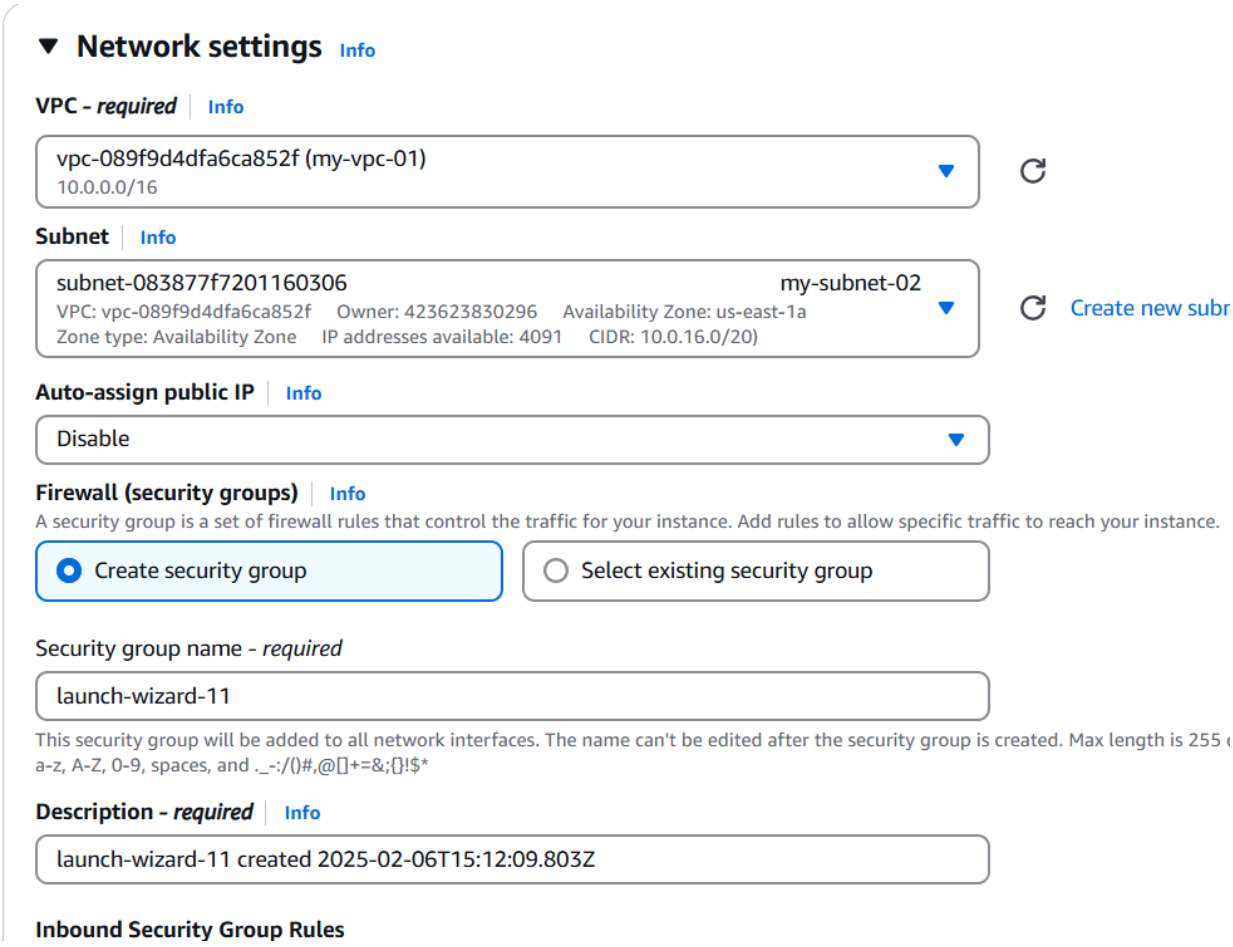
4. Under Network settings:

Select MyPrivateVPC.

Select Private Subnet-A or Private-Subnet-B.

Disable Auto-assign Public IP (to keep it private).





Step 8:

Enable Internal Communication

Instances inside the private subnets can communicate without an internet gateway.

If instances need internet access (for updates, etc.), configure a NAT Gateway in a Public Subnet.

Use Security Groups to allow inbound traffic only from internal sources (e.g., allow SSH from 10.0.0.0/16).

Step 9:

Now, your private network is set up, and instances inside can communicate securely! Let me know if you need extra configurations like VPN, Bastion Host, or NAT setup.

Outcome

After following these steps, you will have:

✅ **A Secure & Isolated VPC** – A Virtual Private Cloud (VPC) that is isolated from other networks, ensuring security and control over your cloud environment.

✅ **Structured Subnet Configuration** – One or more subnets within the VPC, including:

- At least **one public subnet** that can communicate with the internet.
- **Private subnets** (if configured) to host internal resources like databases and backend services.

✅ **Proper Routing & Connectivity** – Correctly configured route tables to enable:

- **Internal communication** between subnets for seamless data exchange.

- **Internet access** for instances in public subnets through an **Internet Gateway (IGW)**.
- **Private subnet internet access** (if required) via a **NAT Gateway**.

✅ **Enhanced Security Measures** (Optional) –

- Security Groups and Network ACLs for traffic control.
- VPC Peering or Transit Gateway for cross-VPC communication.
- Elastic IP and Bastion Host for secure remote access.