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:Vishnu K

Department:CSE

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

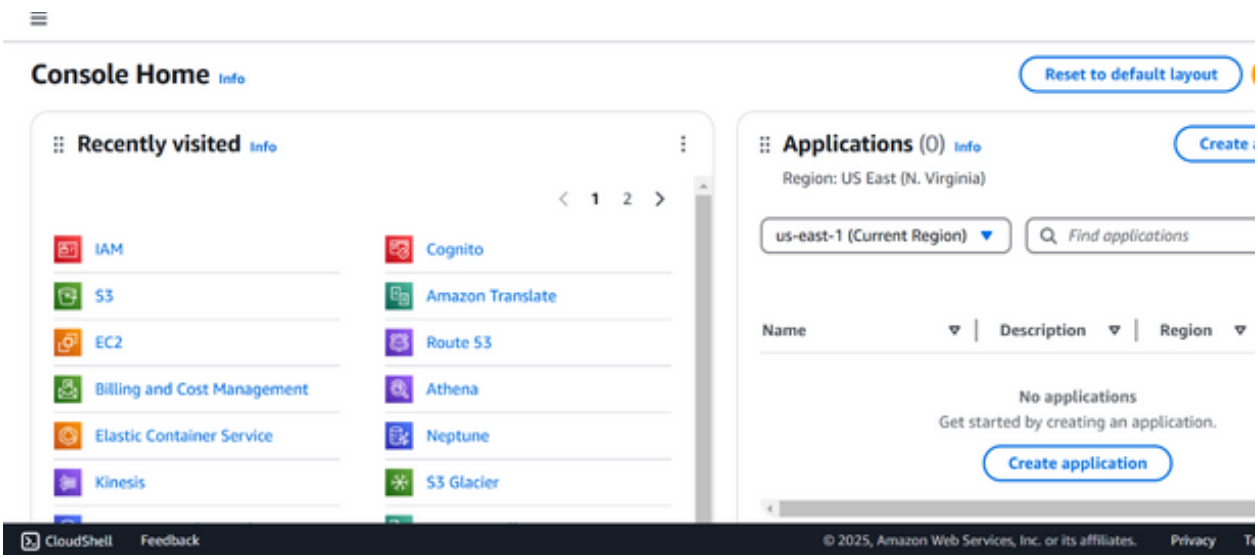
Importance

- 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



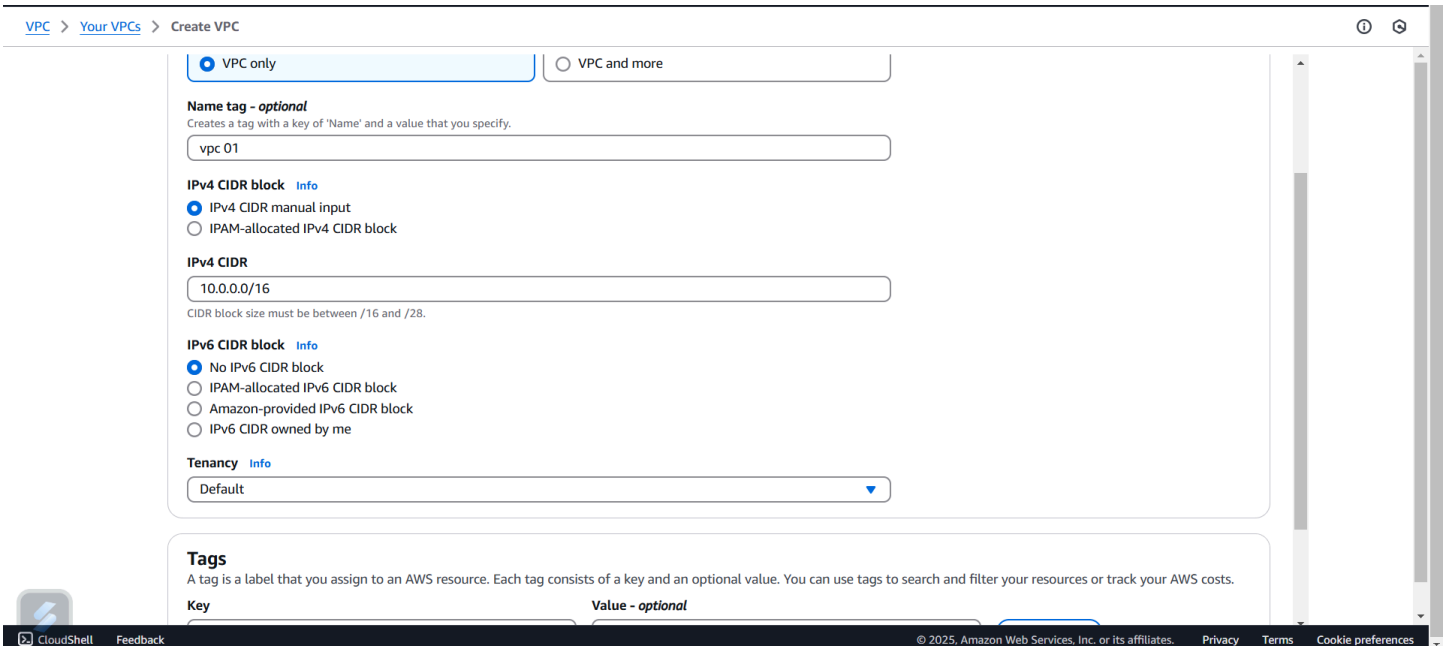
Step 2:

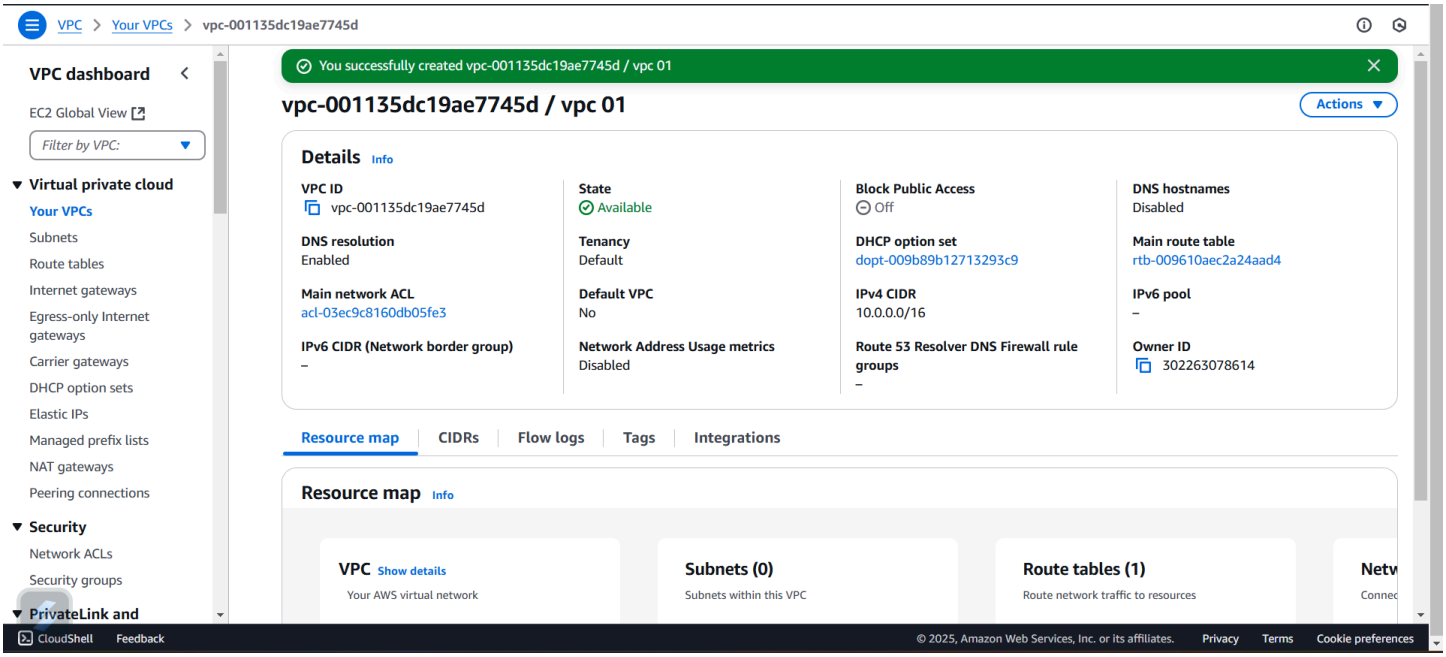
Navigate to the VPC Dashboard

- 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."


VPC > Your VPCs > Create VPC
VPC only
VPC and more
Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.
vpc 01
IPv4 CIDR block Info
IPv4 CIDR manual input
IPAM-allocated IPv4 CIDR block
IPv4 CIDR
10.0.0.0/16
CIDR block size must be between /16 and /28.
IPv6 CIDR block Info
No IPv6 CIDR block
IPAM-allocated IPv6 CIDR block
Amazon-provided IPv6 CIDR block
IPv6 CIDR owned by me
Tenancy Info
Default
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
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


VPC > Your VPCs > vpc-001135dc19ae7745d
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and
You successfully created vpc-001135dc19ae7745d / vpc 01
vpc-001135dc19ae7745d / vpc 01
Actions
Details Info
VPC ID
vpc-001135dc19ae7745d
State
Available
Block Public Access
Off
DNS hostnames
Disabled
DNS resolution
Enabled
Tenancy
Default
DHCP option set
dopt-009b89b12713293c9
Main route table
rtb-009610aec2a24aad4
Main network ACL
acl-03ec9c8160db05fe3
Default VPC
No
IPv4 CIDR
10.0.0.0/16
IPv6 pool
–
IPv6 CIDR (Network border group)
–
Network Address Usage metrics
Disabled
Route 53 Resolver DNS Firewall rule groups
–
Owner ID
302263078614
Resource map
CIDRs
Flow logs
Tags
Integrations
Resource map Info
VPC Show details
Your AWS virtual network
Subnets (0)
Subnets within this VPC
Route tables (1)
Route network traffic to resources
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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:

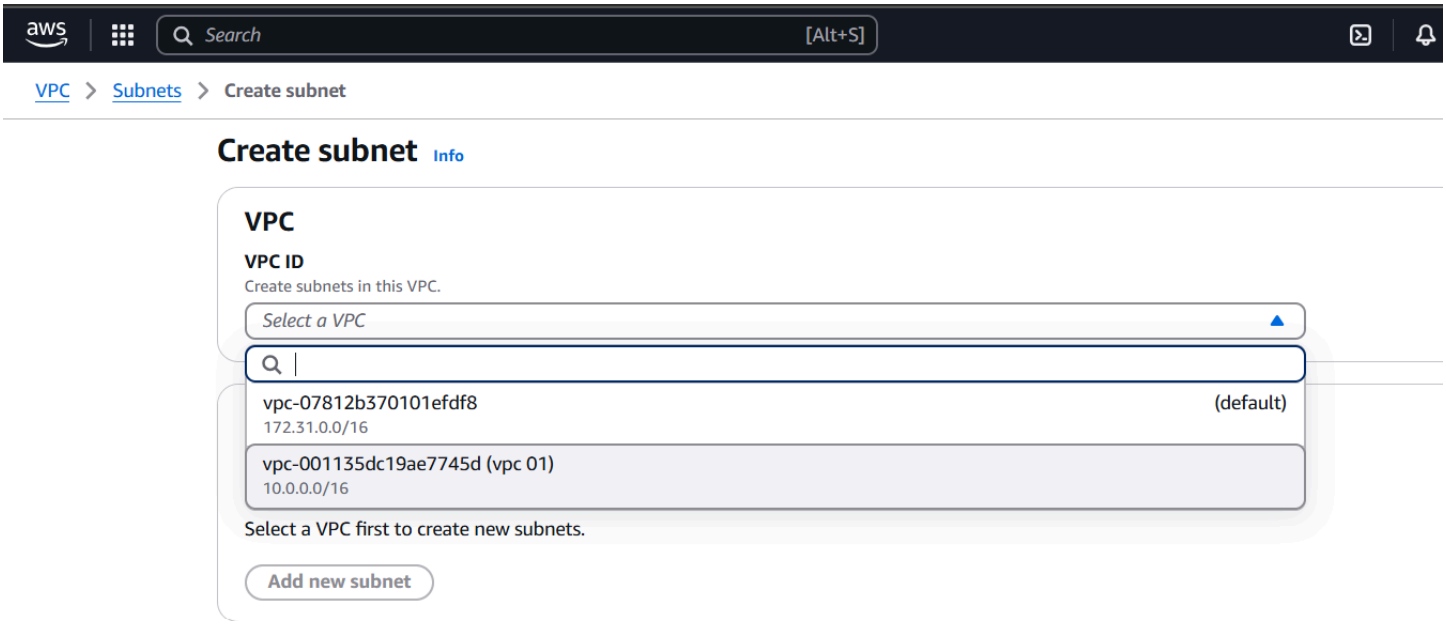
Subnet 1 (Private-Subnet-A)

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



Subnet name

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

sub 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

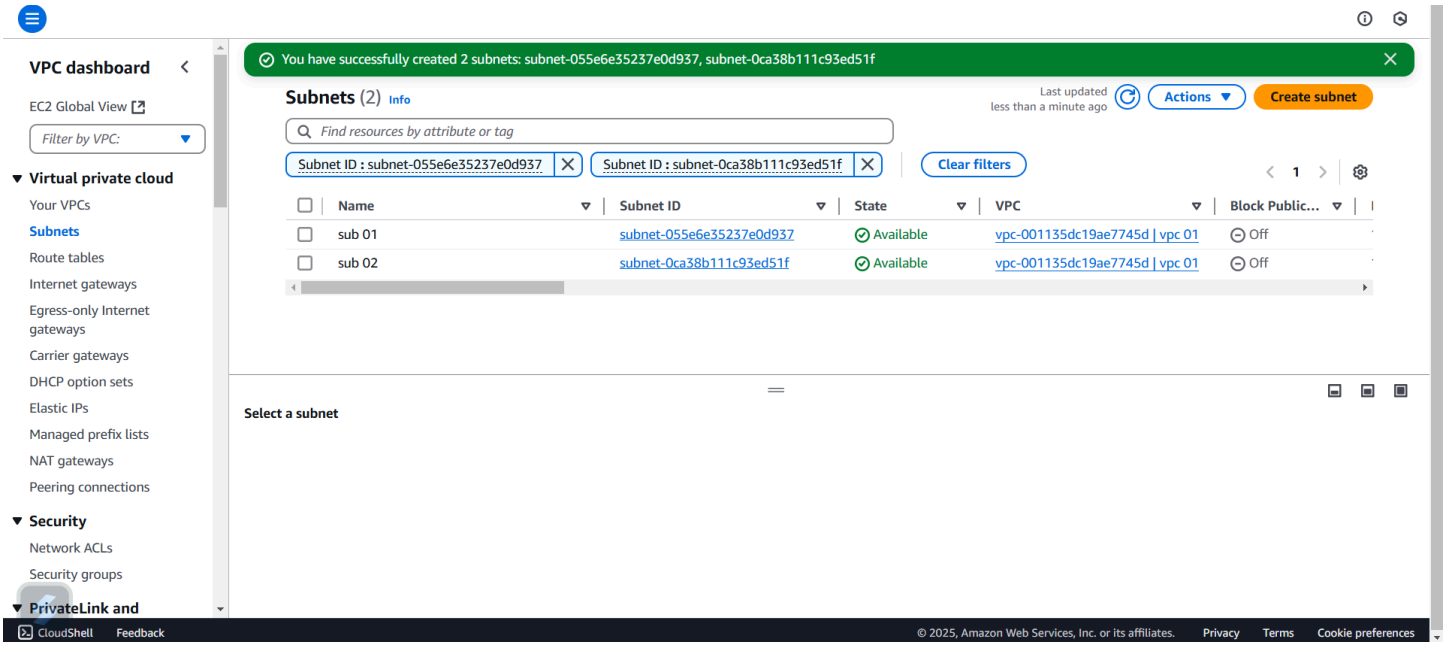
IPv4 VPC CIDR block Info

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

10.0.1.0/24 256 IPs



Step 4:

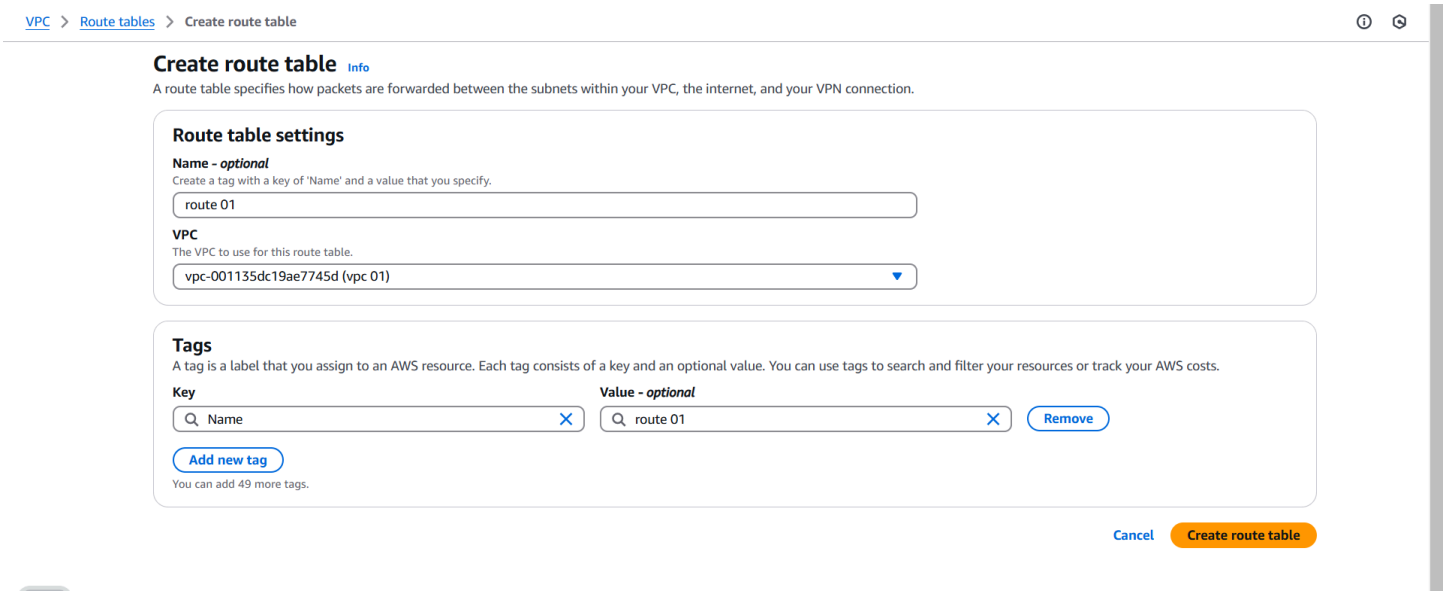
Configure Route Tables for Internal Communication

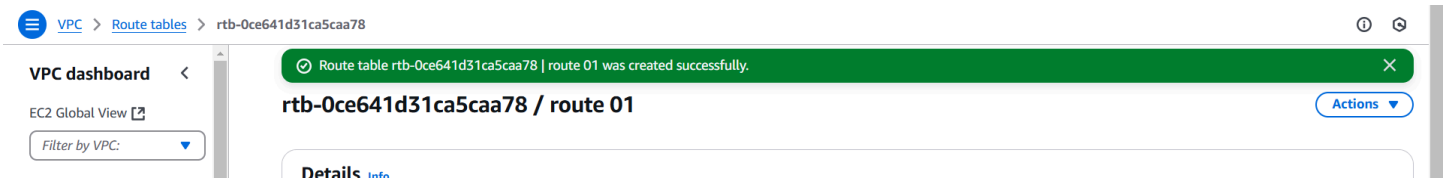
1. Go to Route Tables → Click Create Route Table.

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

3. Select MyPrivateVPC.

4. Click Create.





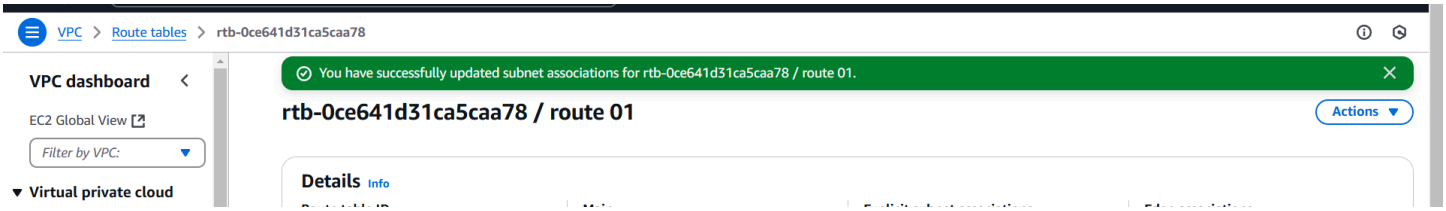
Step 5:

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.).

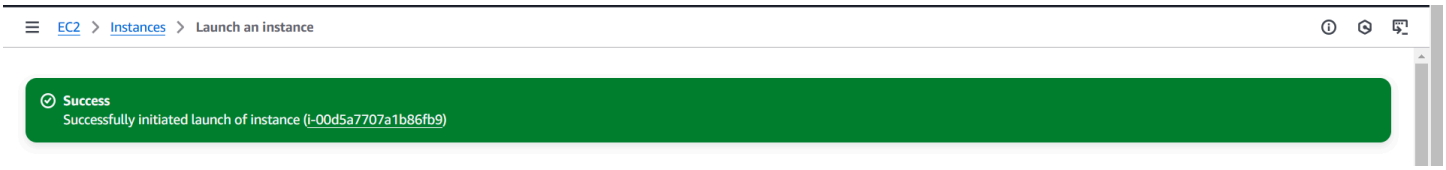
3. Choose an Instance Type (e.g., t2.micro).

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 VPC that is isolated from other networks.
- One or more subnets for your instances, with at least one public subnet that can communicate with the Internet.
- Proper routing configured for internal communication between subnets.