

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 : Sylashri Rajendran
Department: IT

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

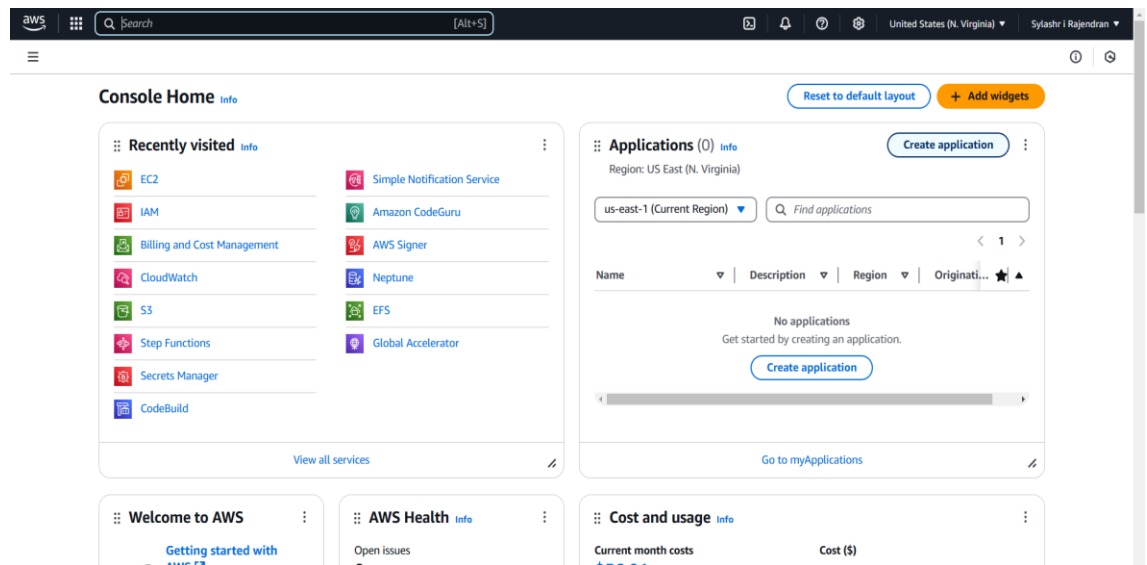
Importance

- **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



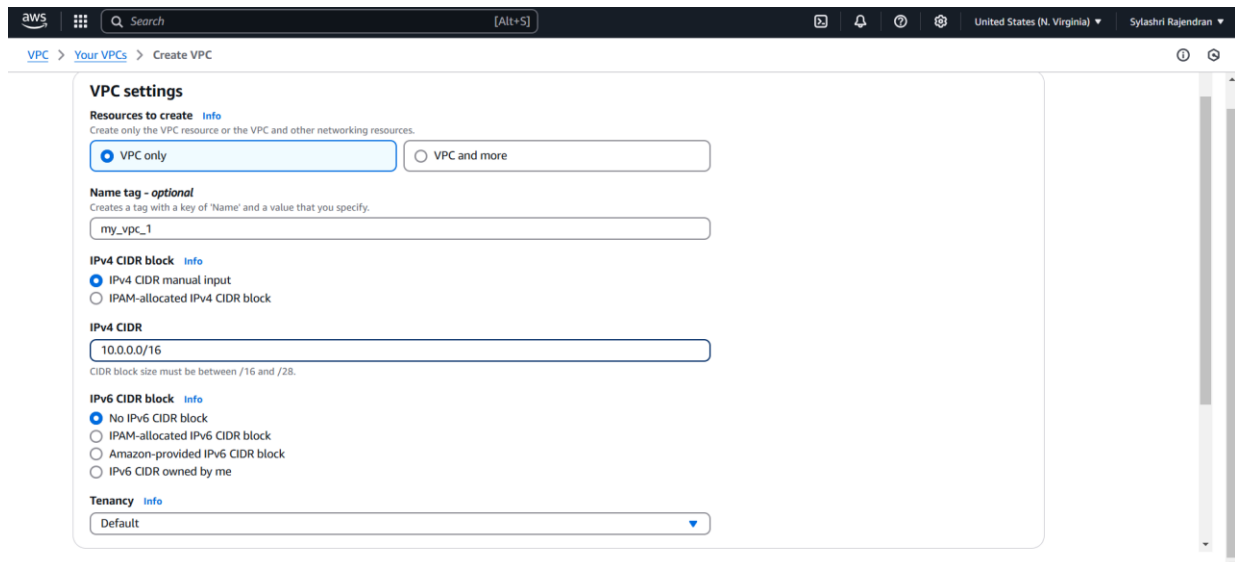
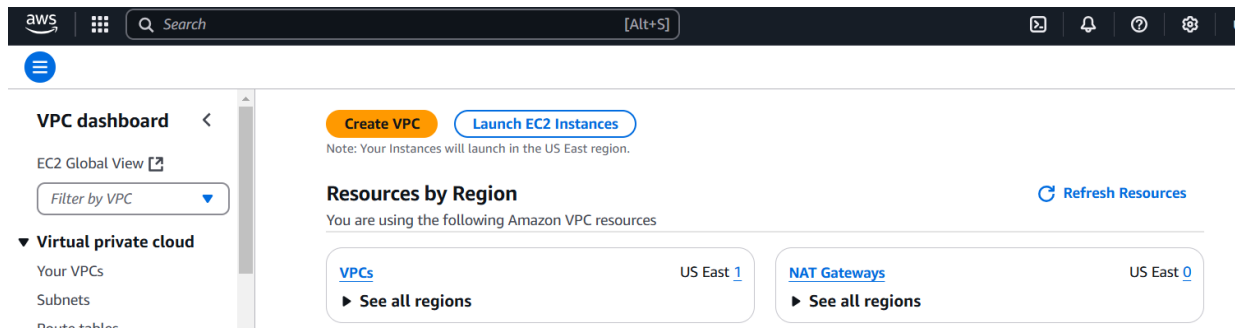
Step 2:

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:**

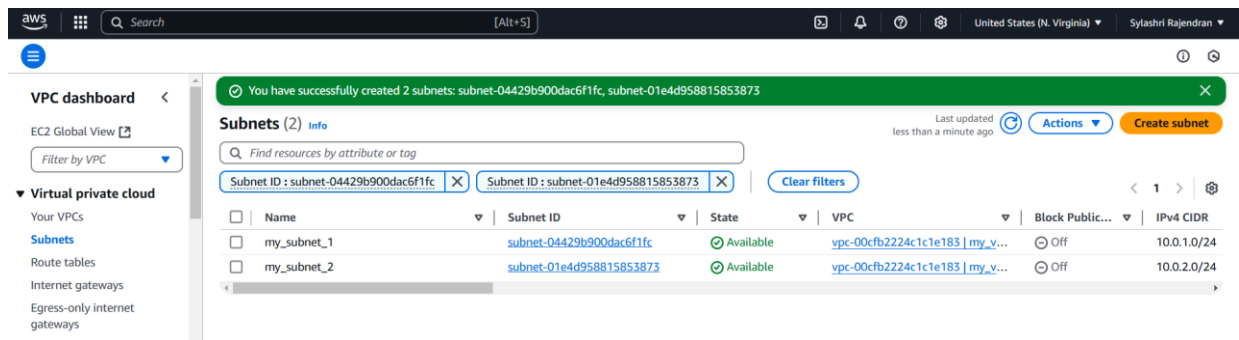
Subnet 1 (Private-Subnet-A)

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



The screenshot shows the AWS VPC dashboard. A green notification bar at the top states: "You have successfully created 2 subnets: subnet-04429b900dac6f1fc, subnet-01e4d958815853873". Below this, the "Subnets (2)" section is active. It features a search bar and a table of subnets. The table has columns for Name, Subnet ID, State, VPC, Block Public..., and IPv4 CIDR. Two subnets are listed: "my_subnet_1" and "my_subnet_2", both in an "Available" state. The left sidebar shows the "VPC dashboard" menu with options like "EC2 Global View", "Filter by VPC", and "Virtual private cloud".

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
my_subnet_1	subnet-04429b900dac6f1fc	Available	vpc-00cfb2224c1c1e183 my_v...	Off	10.0.1.0/24
my_subnet_2	subnet-01e4d958815853873	Available	vpc-00cfb2224c1c1e183 my_v...	Off	10.0.2.0/24

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.

sub-1

The name can be up to 256 characters long.

Availability Zone

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

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.0.0.0/24

IPv4 subnet CIDR block

10.0.0.0/16

65,536 IPs

Tags - optional

Key

Name

Value - optional

sub-1

Remove

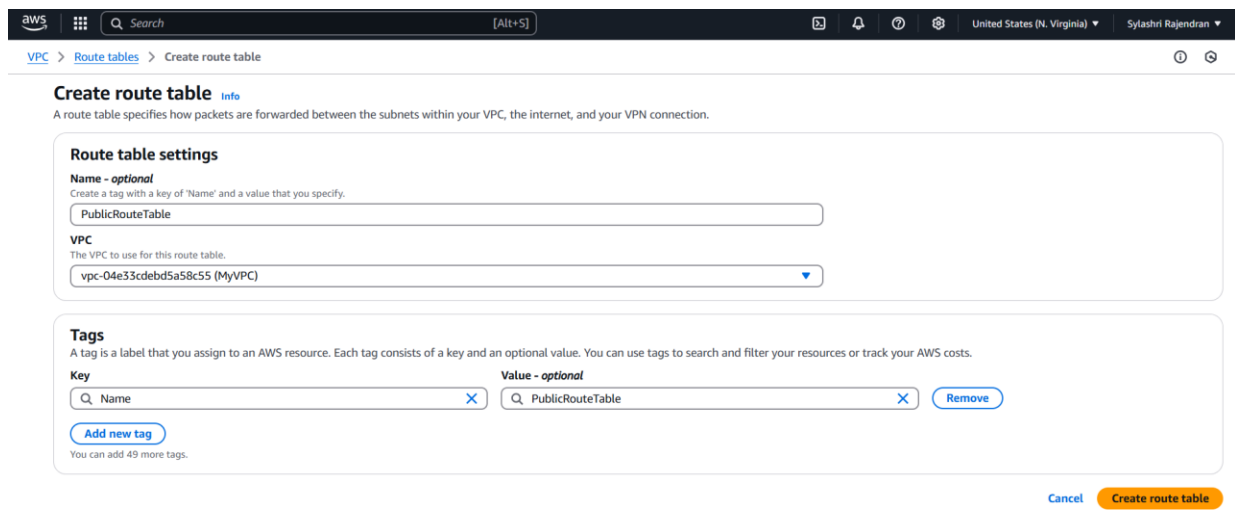
Add new tag

You can add 49 more tags.

Step 4:

Configure Route Tables for Internal Communication

1. Go to Route Tables → Click Create Route Table.
2. Name it (e.g., PublicRouteTable).
3. Select MyPrivateVPC.
4. Click Create.



The screenshot shows the AWS Management Console interface for creating a new route table. The breadcrumb navigation at the top indicates the path: VPC > Route tables > Create route table. The main heading is 'Create route table' with an 'Info' link. Below this, a descriptive sentence states: 'A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.'

The 'Route table settings' section contains two fields: 'Name - optional' with the value 'PublicRouteTable' and 'VPC' with a dropdown menu showing 'vpc-04e33cdebd5a58c55 (MyVPC)'. Below this is the 'Tags' section, which includes a description: '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.' It features a table with two columns: 'Key' and 'Value - optional'. The first row has 'Name' as the key and 'PublicRouteTable' as the value. There are 'Add new tag' and 'Remove' buttons. At the bottom right, there are 'Cancel' and 'Create route table' buttons.

Step 5:

Associate the subnets:

- Go to Subnet Associations → Click Edit subnet associations.
- Select Private-Subnet-A and Private-Subnet-B.
- Click Save associations.

AWS Search [Alt+S] United States (N. Virginia) Sylashri Rajendran

VPC > Route tables > rtb-02d42b802dce5587e > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/2)

Filter subnet associations

<input checked="" type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	my_subnet_1	subnet-04429b900dac6f1fc	10.0.1.0/24	-	rtb-02d42b802dce5587e / PublicRout...
<input checked="" type="checkbox"/>	my_subnet_2	subnet-01e4d958815853873	10.0.2.0/24	-	rtb-02d42b802dce5587e / PublicRout...

Selected subnets

subnet-04429b900dac6f1fc / my_subnet_1 subnet-01e4d958815853873 / my_subnet_2

Step 6:

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

rtb-02d42b802dce5587e / PublicRouteTable

Details Info

Route table ID rtb-02d42b802dce5587e	Main <input type="checkbox"/> No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-00cfb2224c1c1e183 my_vpc_1	Owner ID 842676011451		

Routes Subnet associations Edge associations Route propagation Tags

Routes (1)

Filter routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Step 7:

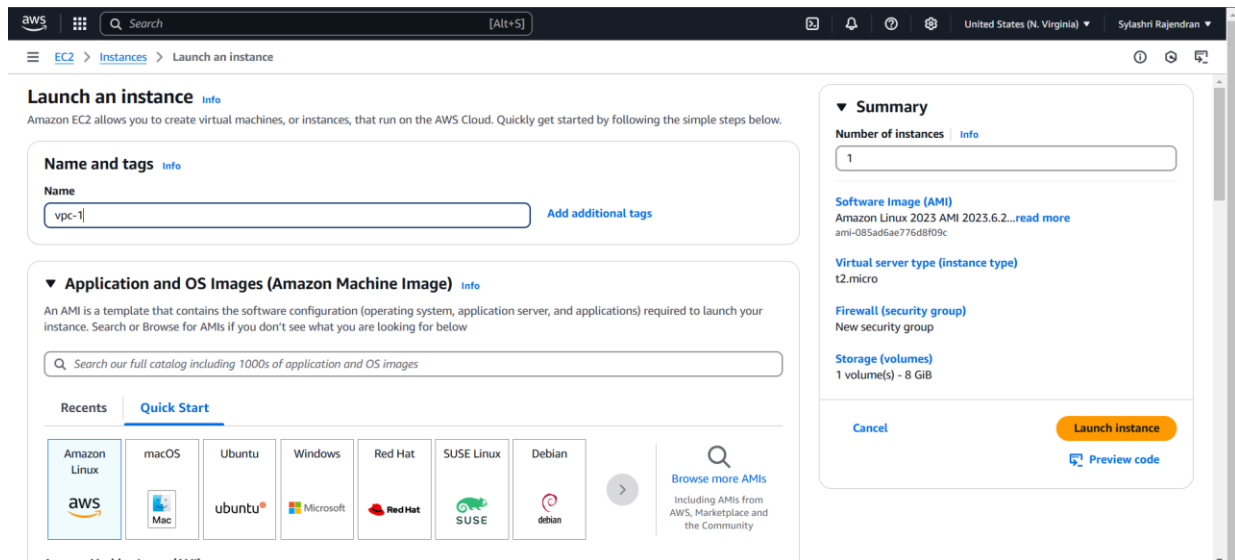
Launch Instances in Private Subnets

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.

