

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 : Sandhana jefrina J
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

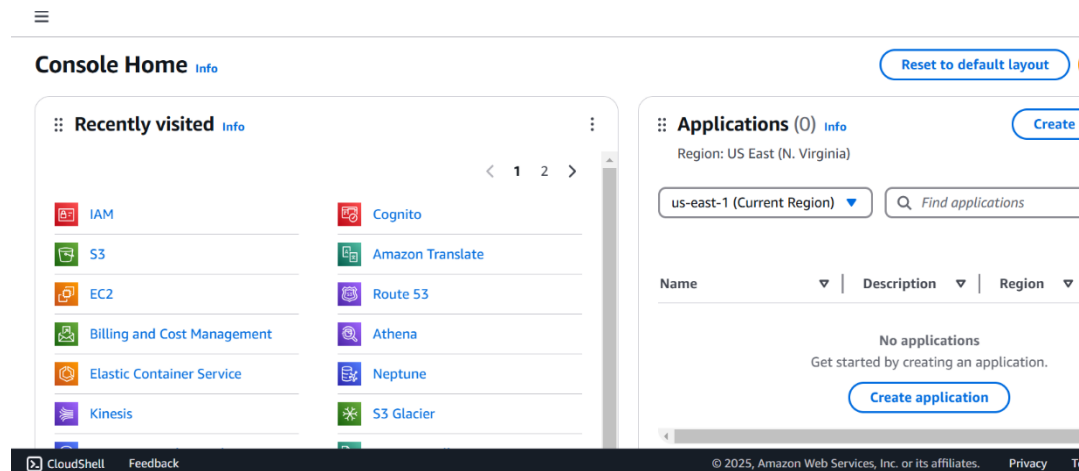
Importance

- **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."

[Create VPC](#)[Launch EC2 Instances](#)

Note: Your Instances will launch in the US East region.

Resources by Region

[Refresh Resources](#)

You are using the following Amazon VPC resources

[VPCs](#)US East [1](#)[► See all regions](#)[NAT Gateways](#)US East [0](#)[► See all regions](#)

VPC > Your VPCs > Create VPC

Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

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

vp-1

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.0.0.0/24

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	
Q Name	Q vp-1	Remove tag

[Add tag](#)

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 'Create subnet' page in the AWS VPC console for 'us-east-1'. The page is titled 'Subnet settings' and instructs the user to 'Specify the CIDR blocks and Availability Zone for the subnet.' It is for 'Subnet 1 of 2'. The 'Subnet name' field contains 'sub-1'. The 'Availability Zone' is set to 'US East (N. Virginia) / us-east-1a'. The 'IPv4 VPC CIDR block' is set to '10.0.0.0/16'. The 'IPv4 subnet CIDR block' is set to '10.0.0.0/24', with a note indicating it provides 256 IPs.

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 [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
10.0.0.0/24 256 IPs

The screenshot shows the 'Create subnet' page in the AWS VPC console for 'us-east-1', specifically for 'Subnet 2 of 2'. The 'Subnet name' field contains 'sub-2'. The 'Availability Zone' is set to 'US East (N. Virginia) / us-east-1b'. The 'IPv4 VPC CIDR block' is set to '10.0.0.0/16'. The 'IPv4 subnet CIDR block' is set to '10.1.0.0/24', with a note indicating it provides 256 IPs. Below the CIDR block field, there are navigation arrows and a 'Tags - optional' section.

Subnet 2 of 2

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
sub-2
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-1b

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
10.1.0.0/24 256 IPs

[Tags - optional](#)

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#subnets:subnetId=subnet-06efa634a6a3aff36,subnet-0cd5d31eab6394ad6

Search [Alt+S]

United States (N. Virginia)

voclabs/user3520451=jefrinajames@gmail.com @ 1535-9647-6481

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

You have successfully created 2 subnets: subnet-06efa634a6a3aff36, subnet-0cd5d31eab6394ad6

Subnets (2)

Info

Last updated less than a minute ago

Actions

Create subnet

Find resources by attribute or tag

Subnet ID : subnet-06efa634a6a3aff36

Subnet ID : subnet-0cd5d31eab6394ad6

Clear filters

< 1 >

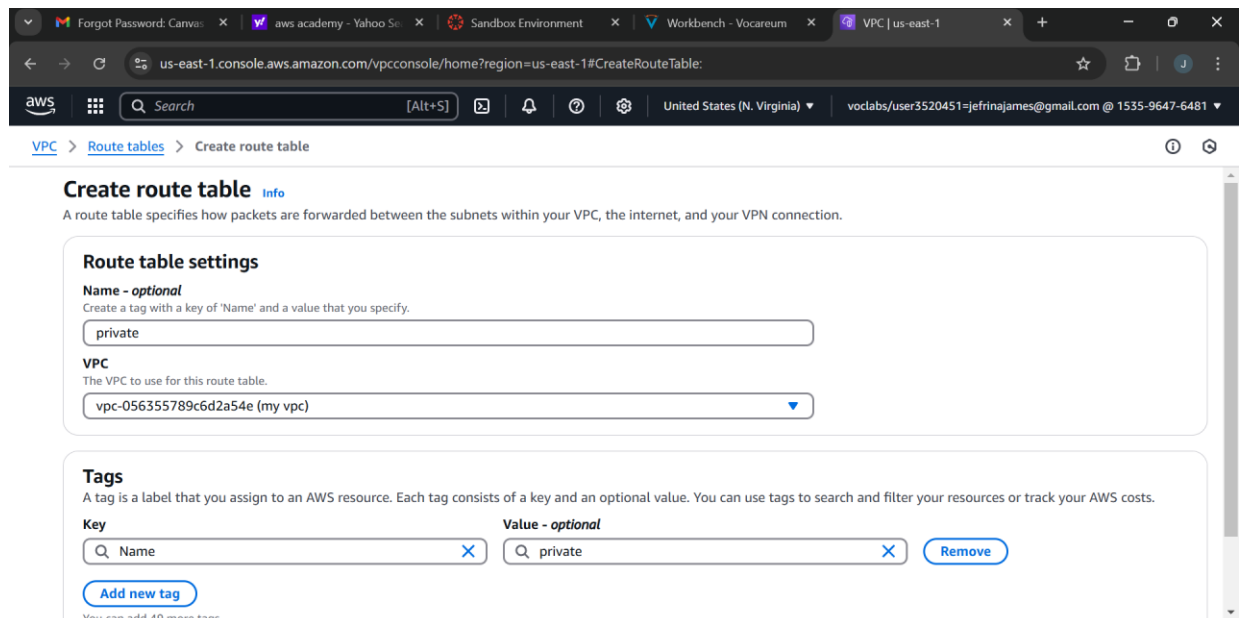
<input type="checkbox"/>	Name	Subnet ID	State	VPC
<input type="checkbox"/>	sub-2	subnet-0cd5d31eab6394ad6	Available	vpc-056355789c6d2a54e my ...
<input type="checkbox"/>	-	subnet-06efa634a6a3aff36	Available	vpc-056355789c6d2a54e my ...

Select a subnet

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.



The screenshot shows the AWS Management Console interface for creating a new route table. The browser address bar indicates the URL: `us-east-1.console.aws.amazon.com/vpconsole/home?region=us-east-1#CreateRouteTable:`. The page title is "Create route table" with an "Info" link. Below the title, 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 form is divided into three main sections: "Route table settings", "Tags", and a "You can add 40 more tags" note at the bottom. In the "Route table settings" section, the "Name - optional" field contains the text "private", and the "VPC" dropdown menu is set to "vpc-056355789c6d2a54e (my vpc)". The "Tags" section shows a table with one tag: Key "Name" and Value "private". There is an "Add new tag" button and a "Remove" button for the existing tag.

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.

private

VPC
The VPC to use for this route table.

vpc-056355789c6d2a54e (my vpc)

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	private

[Add new tag](#) [Remove](#)

You can add 40 more tags.

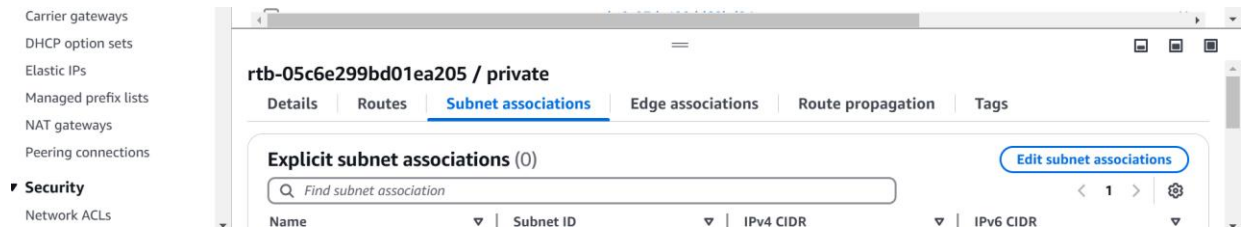
Step 5:

Associate the subnets:

Go to Subnet Associations → Click Edit subnet associations.

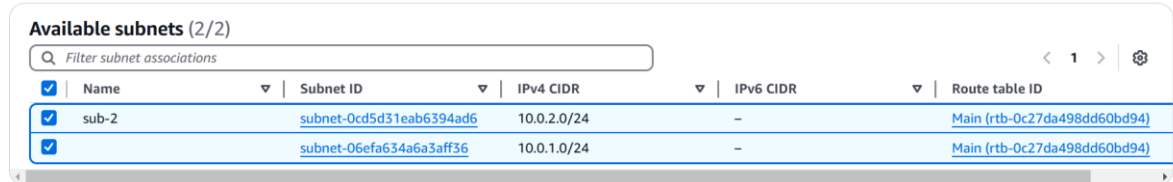
Select Private-Subnet-A and Private-Subnet-B.

Click Save associations.



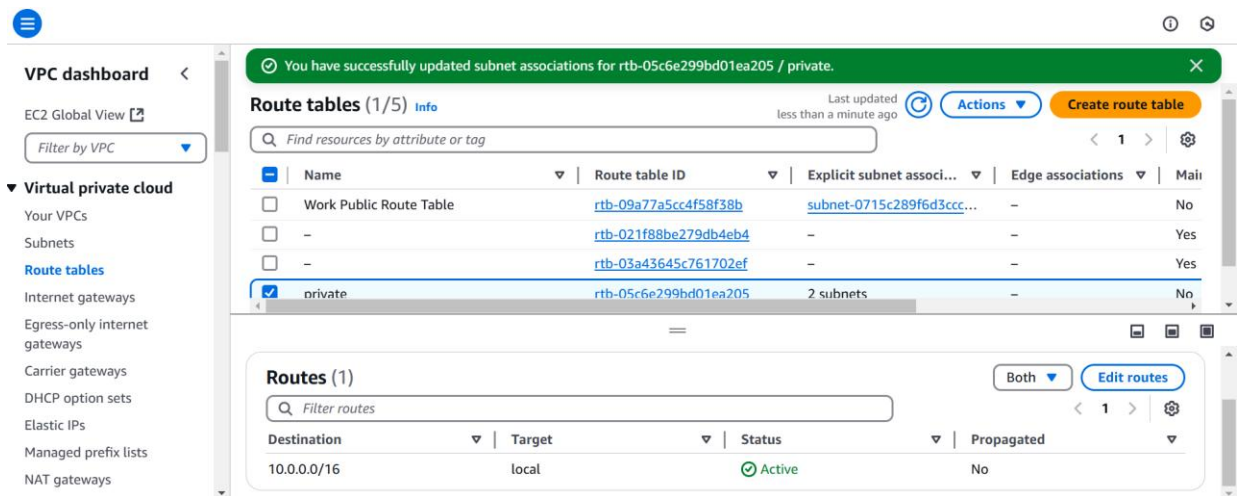
Edit subnet associations

Change which subnets are associated with this route table.



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).

The screenshot shows the AWS Management Console interface for creating a new EC2 instance. The 'Name and tags' section has a text input for 'my instance'. The 'Application and OS Images (Amazon Machine Image)' section includes a search bar and a grid of AMIs. The 'Summary' section on the right provides a overview of the configuration: 1 instance, Amazon Linux 2023.6.2 AMI, t2.micro instance type, and a new security group. The 'Launch instance' button is prominent.

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.

