



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: HARSHINI A

Department: CSE



Introduction

In cloud environments, securing internal communication is crucial for maintaining data integrity and minimizing exposure to the public internet. A Virtual Private Cloud (VPC) enables organizations to create isolated network environments with controlled access. This PoC demonstrates the process of setting up a **private network** in the cloud, creating **subnets**, and configuring **internal routing** for communication between instances.

Objectives

1. Create a VPC with private and public subnets.
2. Configure routing tables to enable internal communication.
3. Deploy instances within the subnets and verify private communication.
4. Ensure no direct internet access for private subnets while allowing controlled outbound access.

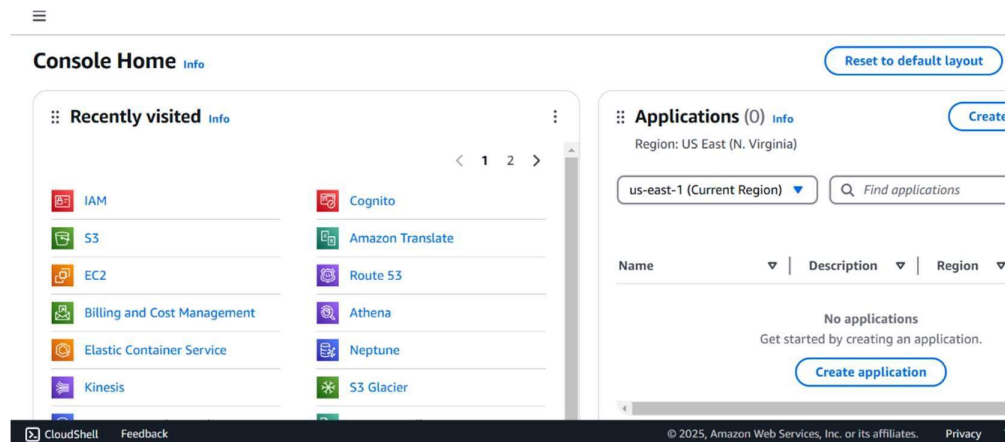
Importance

1. **Security:** Limits exposure to external threats by restricting internet access.
2. **Performance:** Reduces latency by keeping communication within the private network.
3. **Compliance:** Helps meet regulatory requirements for sensitive data handling.
4. **Scalability:** Allows better control over network traffic as workloads grow.

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.

vpc-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 X Q vpc-1 X [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

Availability Zone: us-east-2a (example)

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.

10.0.0.0/24

IPv4 subnet CIDR block

10.1.0.0/16 65,536 IPs

< > ^ v

Tags - optional

Key	Value - optional	
Q Name	Q sub-2	Remove
Add new tag		
You can add 49 more tags.		
Remove		

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.

10.0.0.0/24

IPv4 subnet CIDR block

10.0.0.0/16 65,536 IPs

< > ^ v

Tags - optional

Key	Value - optional	
Q Name	Q 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., PrivateRouteTable).
3. Select MyPrivateVPC.
4. Click Create.

VPC > Route tables > Create route table

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-0b07dbbc4d9e68588 (vpc-1)

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

Q Name X

Value - optional

Q private X Remove

Add new tag

You can add 49 more tags.

Cancel Create route table

Step 5:

Associate the subnets:

Go to Subnet Associations → Click Edit subnet associations.

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

Click Save associations.

VPC vpc-0b07dbbc4d9e68588 | vpc-1

Owner ID 774305605711

Routes Subnet associations Edge associations Route propagation Tags

Explicit subnet associations (0) [Edit subnet associations](#)

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
No subnet associations			

You do not have any subnet associations.

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/2) [Filter subnet associations](#)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
sub-2	subnet-08d686eb3bfd5c1c	10.0.2.0/24	-	Main (rtb-0511a15ded68d344d)
sub-1	subnet-0a23be0f9dc2a24aa	10.0.1.0/24	-	Main (rtb-0511a15ded68d344d)

Selected subnets

subnet-08d686eb3bfd5c1c / sub-2 X subnet-0a23be0f9dc2a24aa / sub-1 X

Cancel Save associations

Step 6:

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

rtb-09bd5c6927b161264 / private Actions

Details Info
Route table ID
rtb-09bd5c6927b161264
VPC
vpc-0b07dbbc4d9e68588 | vpc-1

Main
No
Owner ID
774305605711

Explicit subnet associations
2 subnets

Edge associations
-

Routes Subnet associations Edge associations Route propagation Tags

Routes (1) Both Edit 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).

EC2

Instances

Launch an instance

Launch an instance

Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Info

Name

vpc-1

Add additional tags

Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q

Search our full catalog including 1000s of application and OS images

Recents

My AMIs

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0c614dee691cbbf37 (64-bit (x86), uefi-preferred) / ami-0b29c89c15cfb8a6d (64-bit (Arm), uefi)

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

▼ Summary

Number of

1

Software I

Amazon Lin

ami-0c614de

Virtual ser

t2.micro

Firewall (s

default

Storage (v

1 volume(s

Free (or t

insta

publ

storz

banc

Cancel

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).

Outcome

After following these steps, you will have:

1. The private instance should not have direct internet access.
2. Internal communication should work seamlessly between private subnets.
3. The public instance should be able to connect to the private instance using SSH.
4. Internet access for the private instance should be routed through the NAT Gateway (if configured).

