CLOUD COMPUTING SERVICES LAB (AWS)

WEEK 10: Create VPC and Launch Windows EC2

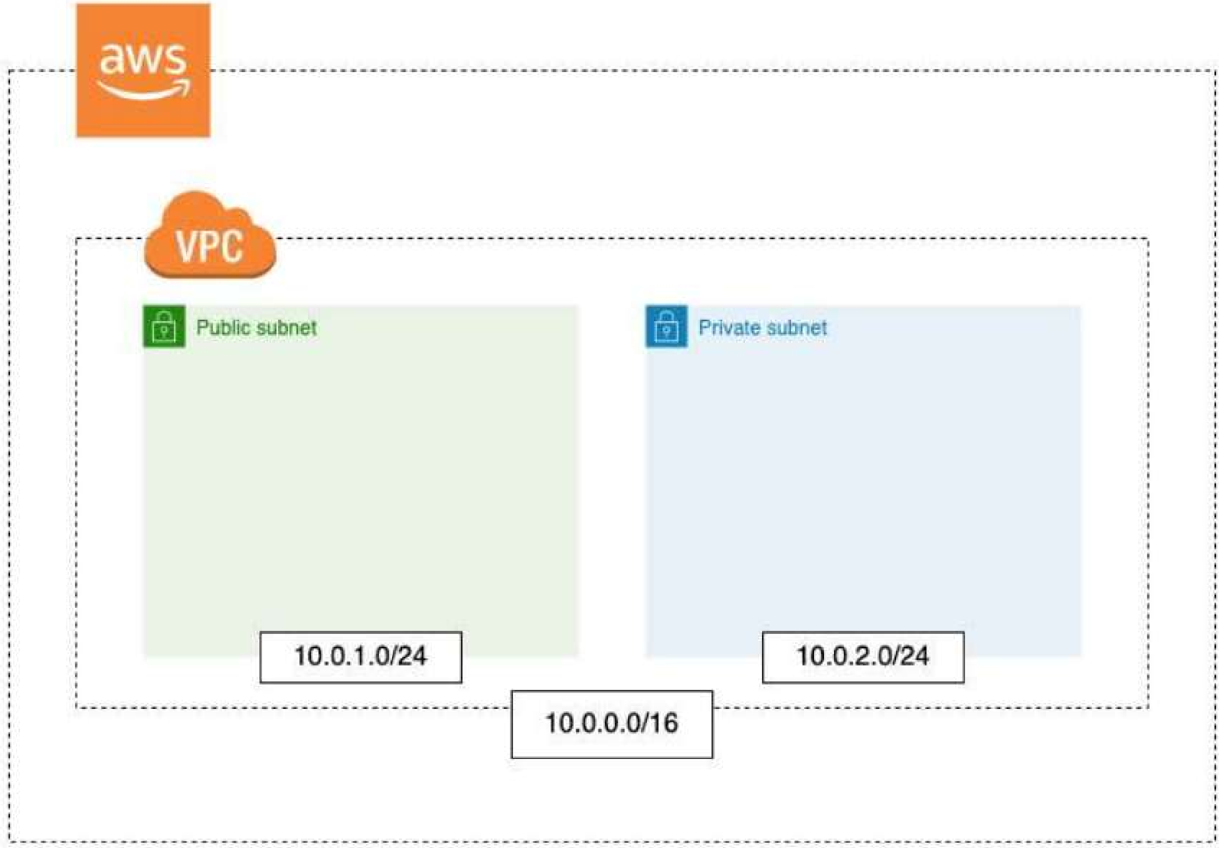
OBJECTIVE: Create VPC and Launch Windows EC2

PROCEDURE:

Phase 1: Create Virtual Private Cloud (VPC) with 2 Subnets (1 Public and 1 Private)

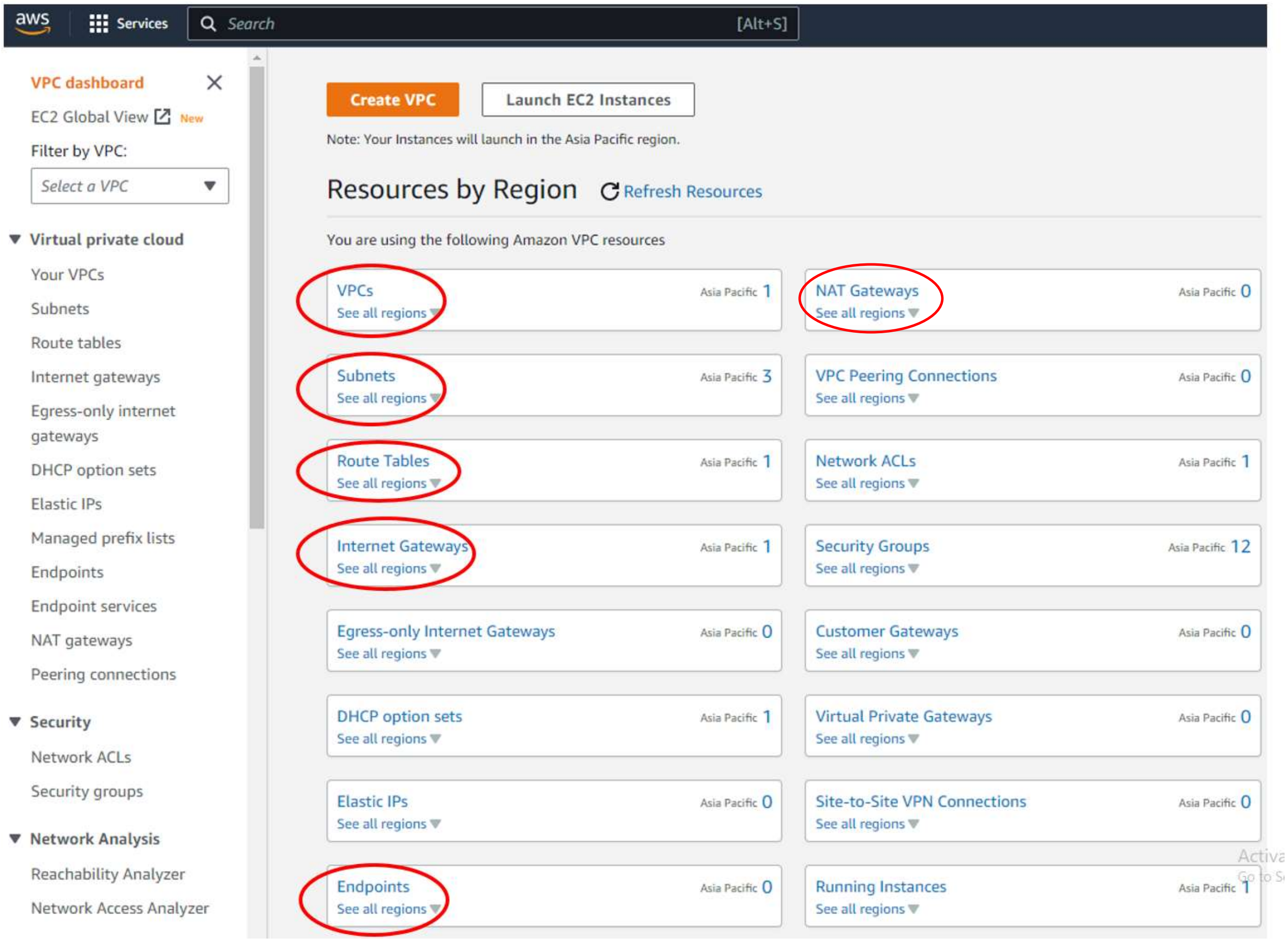
Phase 2: Launch Two Instances of Windows EC2

- 1st Instance: To deploy Angular Application (Front-End) on Public Subnet (10.0.1.4)
- 2nd Instance: To deploy FastAPI Application (Back-End) on Private Subnet (10.0.2.4)



Phase 1: Create Virtual Private Cloud (VPC)

Step 1: Login to AWS Management Console → Go to VPC Dashboard



VPC dashboard
EC2 Global View New
Filter by VPC:
Select a VPC

- Virtual private cloud
  - Your VPCs
  - Subnets
  - Route tables
  - Internet gateways
  - Egress-only internet gateways
  - DHCP option sets
  - Elastic IPs
  - Managed prefix lists
  - Endpoints
  - Endpoint services
  - NAT gateways
  - Peering connections
- Security
  - Network ACLs
  - Security groups
- Network Analysis
  - Reachability Analyzer
  - Network Access Analyzer

Create VPC | Launch EC2 Instances

Note: Your Instances will launch in the Asia Pacific region.

Resources by Region Refresh Resources

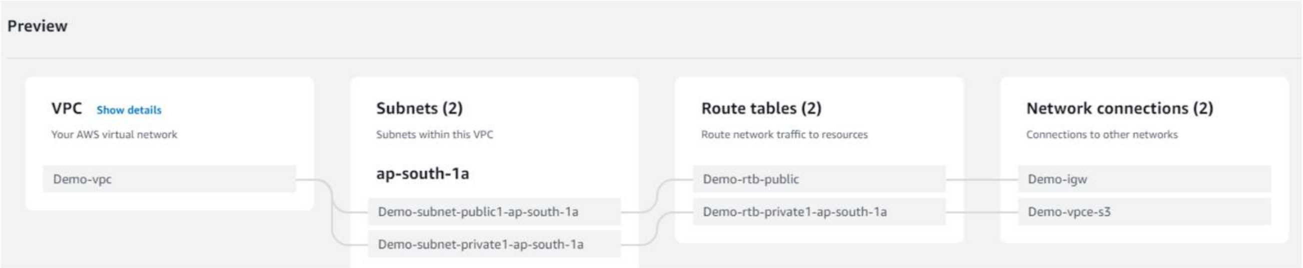
You are using the following Amazon VPC resources

Resource	Region	Count	Resource	Region	Count
VPCs See all regions	Asia Pacific	1	NAT Gateways See all regions	Asia Pacific	0
Subnets See all regions	Asia Pacific	3	VPC Peering Connections See all regions	Asia Pacific	0
Route Tables See all regions	Asia Pacific	1	Network ACLs See all regions	Asia Pacific	1
Internet Gateways See all regions	Asia Pacific	1	Security Groups See all regions	Asia Pacific	12
Egress-only Internet Gateways See all regions	Asia Pacific	0	Customer Gateways See all regions	Asia Pacific	0
DHCP option sets See all regions	Asia Pacific	1	Virtual Private Gateways See all regions	Asia Pacific	0
Elastic IPs See all regions	Asia Pacific	0	Site-to-Site VPN Connections See all regions	Asia Pacific	0
Endpoints See all regions	Asia Pacific	0	Running Instances See all regions	Asia Pacific	1

Step 2: Create VPC

IPv4 CIDR: 10.0.0.0/16

- Click on Create VPC



Preview

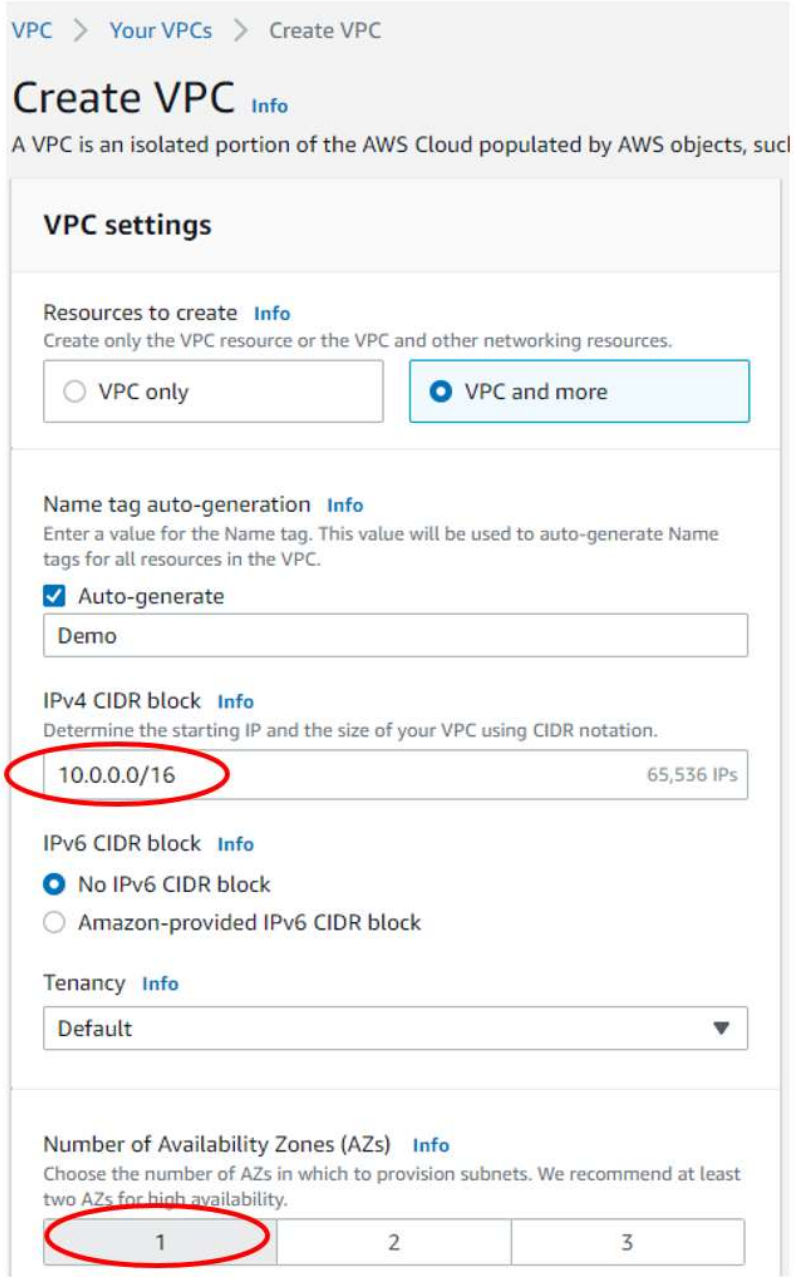
VPC Show details
Your AWS virtual network
Demo-vpc

Subnets (2)
Subnets within this VPC
ap-south-1a
Demo-subnet-public1-ap-south-1a
Demo-subnet-private1-ap-south-1a

Route tables (2)
Route network traffic to resources
Demo-rtb-public
Demo-rtb-private1-ap-south-1a

Network connections (2)
Connections to other networks
Demo-igw
Demo-vpce-s3

- Select VPC and more
- Enter IPv4 CIDR Block
  10.0.0.0/16


VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, sucl

VPC settings

Resources to create Info

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

○ VPC only ● VPC and more

Name tag auto-generation Info

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☑ Auto-generate

Demo

IPv4 CIDR block Info

Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs

IPv6 CIDR block Info

● No IPv6 CIDR block

○ Amazon-provided IPv6 CIDR block

Tenancy Info

Default ▼

Number of Availability Zones (AZs) Info

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

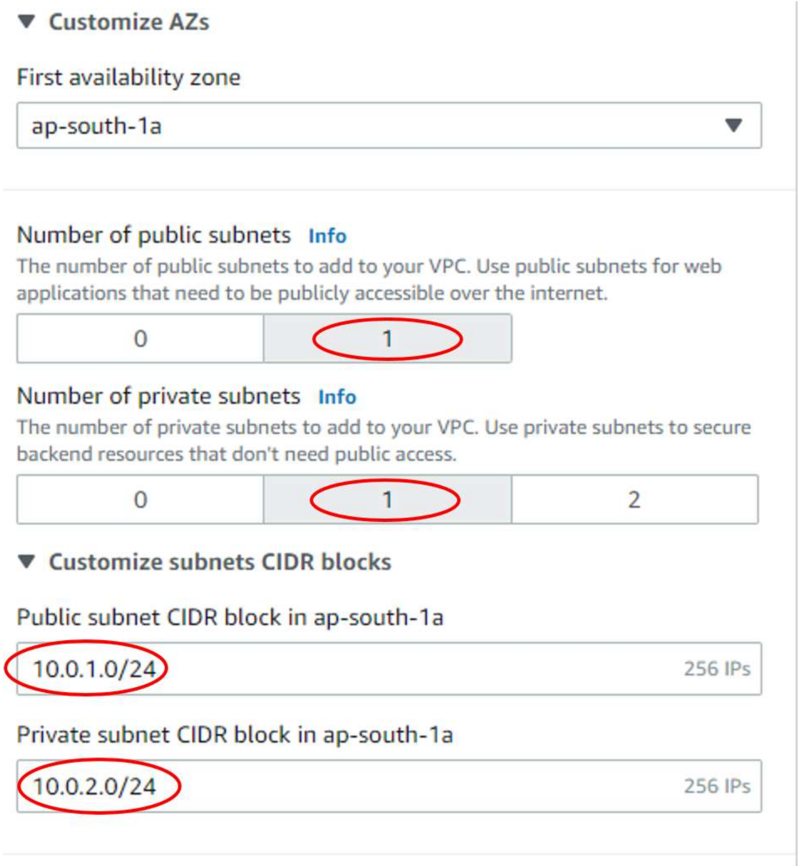
1 | 2 | 3

Step 3: Create Private and Public Subnets with Internet Gateway

IPv4 CIDR: 10.0.1.0/24 – for Public Subnet

IPv4 CIDR: 10.0.2.0/24 – for Private Subnet

- No. of Availability of Zones
  1
- No. of Public Subnets
  1
- No. of Private Subnets
  1
- Public Subnet CIDR Block
  10.0.1.0/24
- Private Subnet CIDR Block
  10.0.2.0/24



▼ Customize AZs

First availability zone

ap-south-1a ▼

Number of public subnets Info

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 [✓ 1]

Number of private subnets Info

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 [✓ 1] 2

▼ Customize subnets CIDR blocks

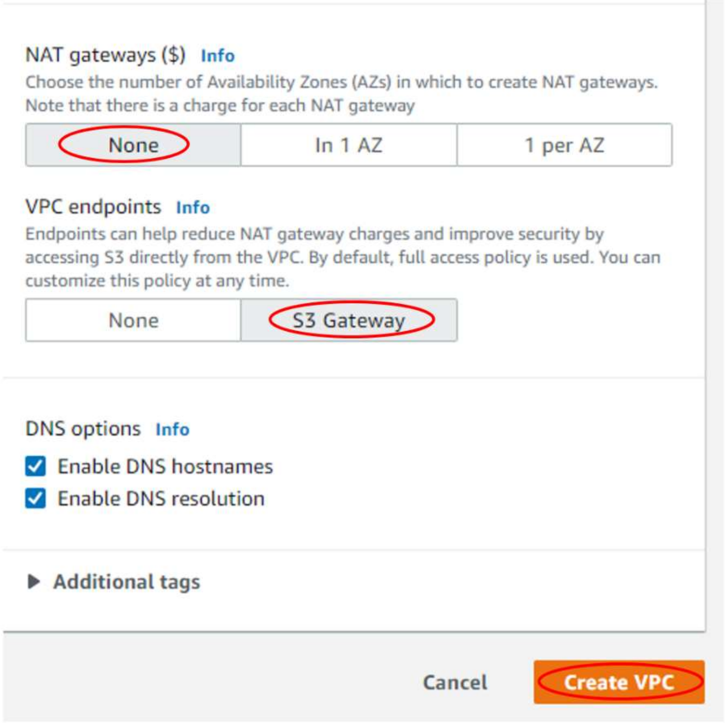
Public subnet CIDR block in ap-south-1a

[✓ 10.0.1.0/24] 256 IPs

Private subnet CIDR block in ap-south-1a

[✓ 10.0.2.0/24] 256 IPs

- NAT Gateway
  None
- VPC endpoints (Optional – If your Back-end Application needs access to S3 Bucket)
  S3 Gateway



NAT gateways ($) Info

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

[✓ None] In 1 AZ 1 per AZ

VPC endpoints Info

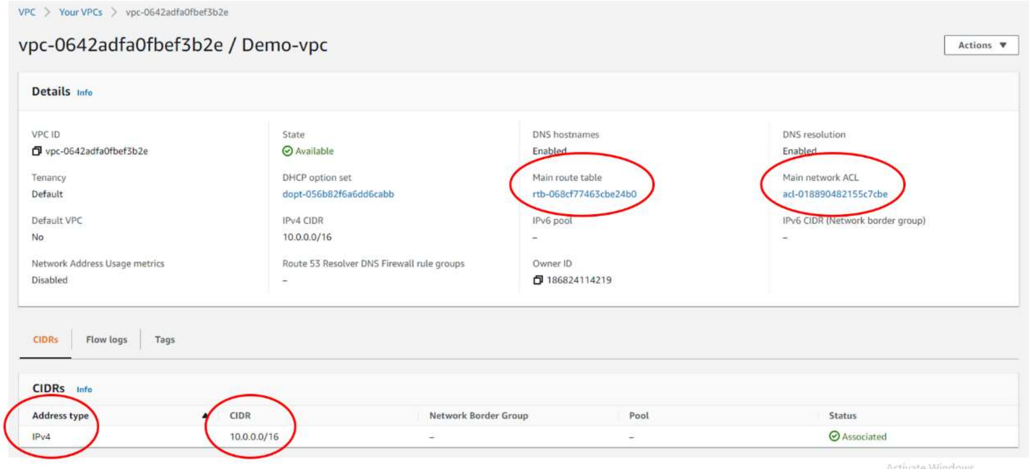
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

None [✓ S3 Gateway]

DNS options Info

☑ Enable DNS hostnames

☑ Enable DNS resolution

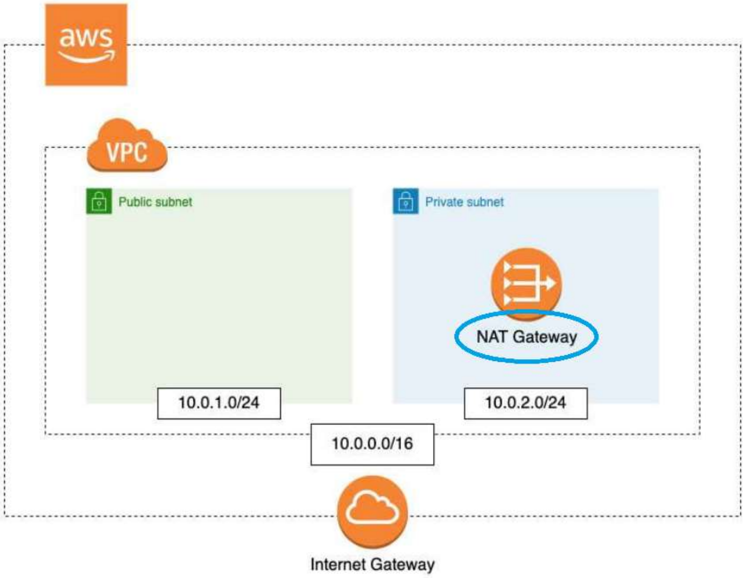


vpc-0642adfa0fbef3b2e / Demo-vpc

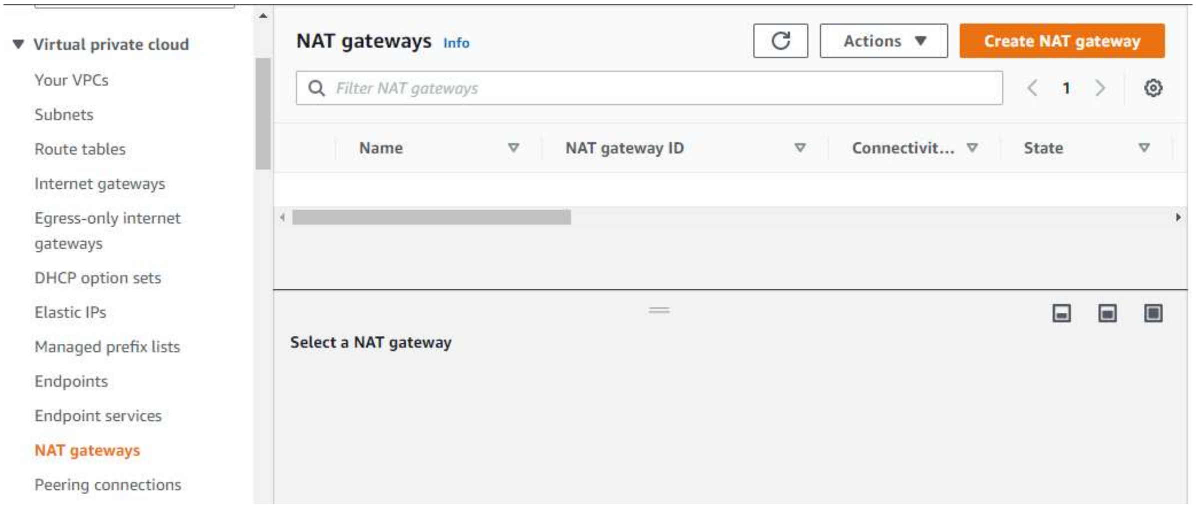
VPC ID: vpc-0642adfa0fbef3b2e
State: Available
DNS hostnames: Enabled
DNS resolution: Enabled
Tenancy: Default
DHCP option set: dopt-056b82f6a6dd6cabb
Main route table: rtb-068cf77463cbe24b0
Main network ACL: acl-018890482155c7cbe
Default VPC: No
IPv4 CIDR: 10.0.0.0/16
IPv6 pool: –
IPv6 CIDR (Network border group): –
Network Address Usage metrics: Disabled
Route 53 Resolver DNS Firewall rule groups: –
Owner ID: 186824114219

Address type	CIDR	Network Border Group	Pool	Status
IPv4	10.0.0.0/16	–	–	Associated

Step 4: Add NAT Gateway (Optional – Create only if Internet Access is required to Private Subnet)
A Network Address Translation NAT Gateway is used to provide Internet Traffic OR AWS Resources to EC2 instances in Private Cloud.



- NAT Gateway

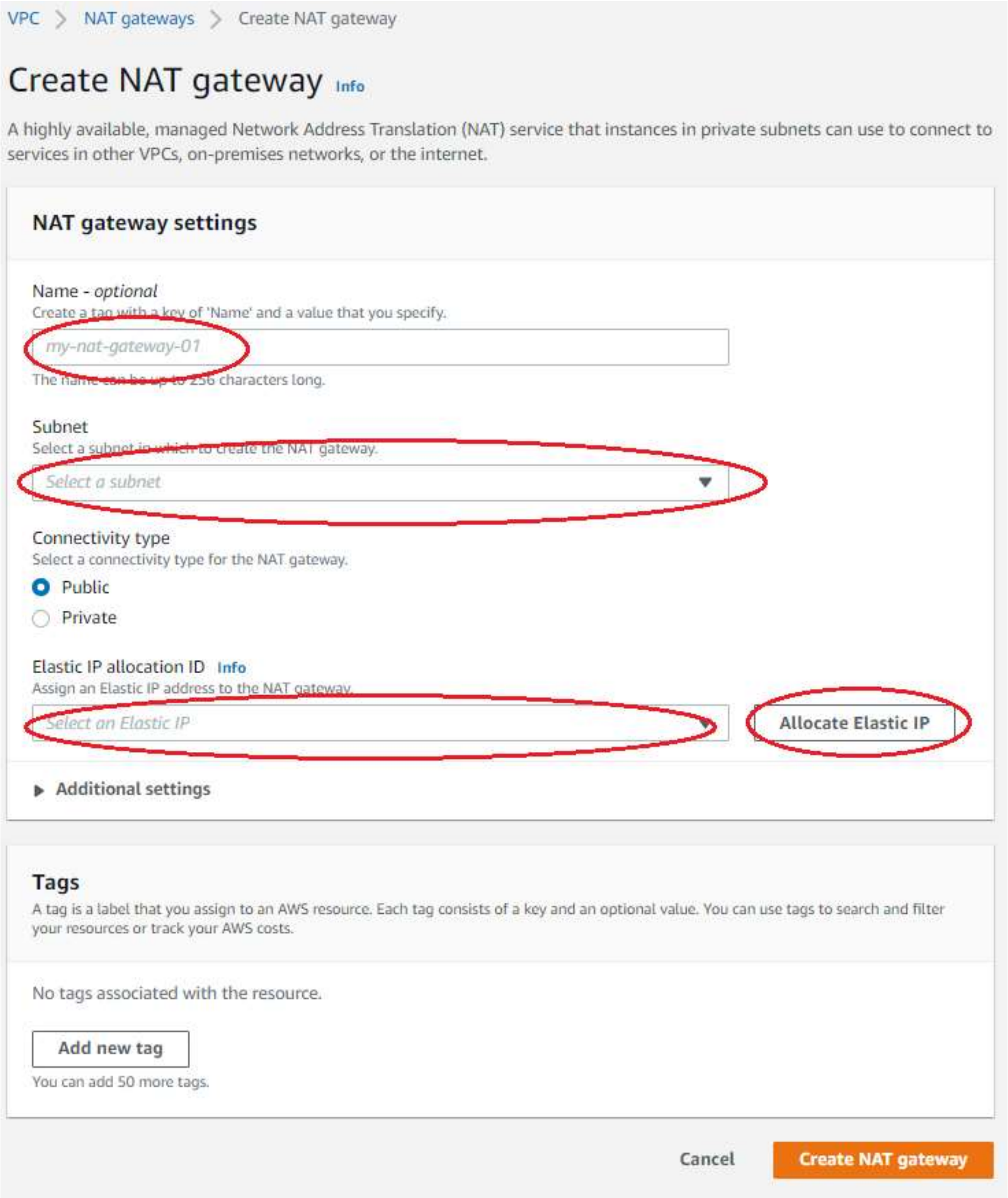


NAT gateways Info

Name	NAT gateway ID	Connectivit...	State

Select a NAT gateway

- ➢ Enter NAT Gateway Name
  My-nat-gw-1
- ➢ Select Subnet
  Public Subnet
- ➢ Click Allocate Elastic IP



VPC > NAT gateways > Create NAT gateway

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
my-nat-gateway-01
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
Select a subnet

Connectivity type
Select a connectivity type for the NAT gateway.
● Public
○ Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.
Select an Elastic IP
Allocate Elastic IP

▶ Additional settings

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.

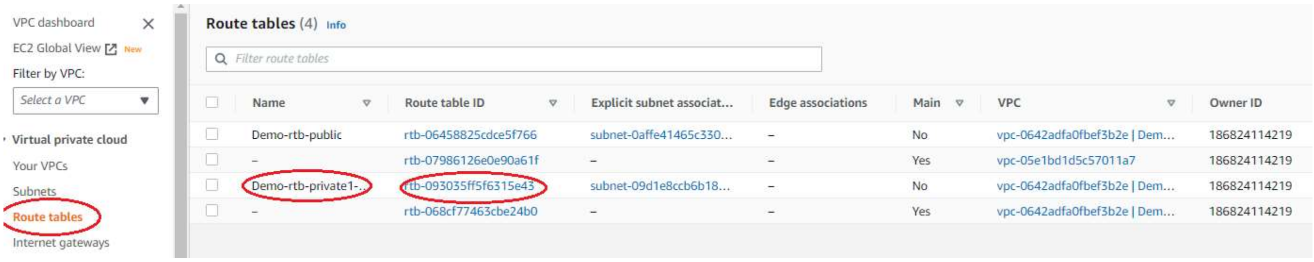
No tags associated with the resource.

Add new tag
You can add 50 more tags.

Cancel Create NAT gateway

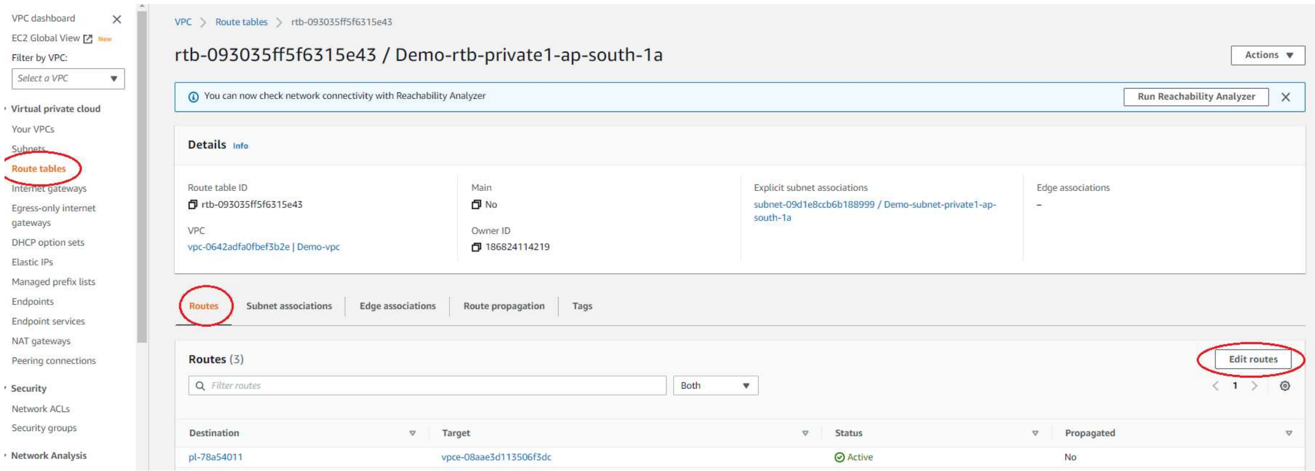
Step 6: Associate Route Tables to Private Subnet

- ➢ Select Route tables from left menu
- ➢ Select Private Subnet
- ➢ Select Routes
- ➢ Edit Routes
- ➢ Enter Destination 0.0.0.0/0
- ➢ Target → Select NAT Gateway → Save Changes



Route tables (4) Info

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
Demo-rtb-public	rtb-06458825cdce5f766	subnet-0affe41465c330...	–	No	vpc-0642adfa0fbef3b2e | Dem...	186824114219
–	rtb-07986126e0e90a61f	–	–	Yes	vpc-05e1bd1d5c57011a7	186824114219
Demo-rtb-private1-...	rtb-093035ff5f6315e43	subnet-09d1e8ccb6b18...	–	No	vpc-0642adfa0fbef3b2e | Dem...	186824114219
–	rtb-068cf77463cbe24b0	–	–	Yes	vpc-0642adfa0fbef3b2e | Dem...	186824114219



VPC > Route tables > rtb-093035ff5f6315e43

rtb-093035ff5f6315e43 / Demo-rtb-private1-ap-south-1a

You can now check network connectivity with Reachability Analyzer

Details Info

Route table ID: rtb-093035ff5f6315e43

Main: No

Explicit subnet associations: subnet-09d1e8ccb6b188999 / Demo-subnet-private1-ap-south-1a

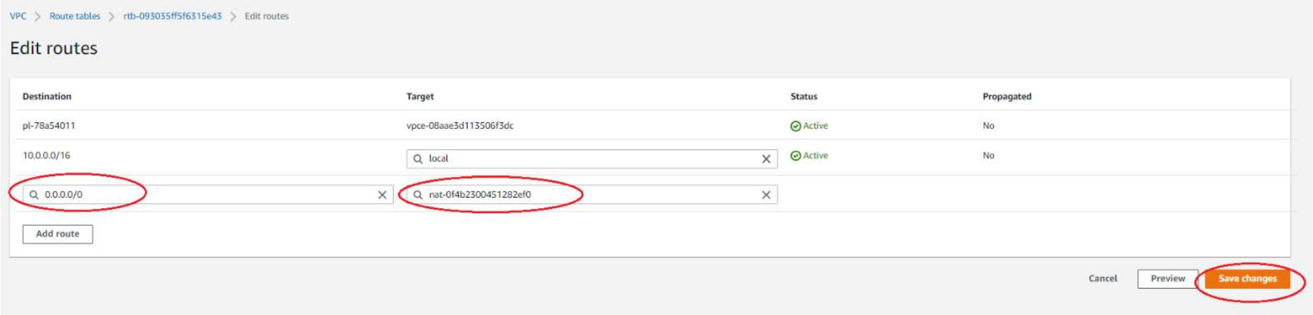
Edge associations: –

VPC: vpc-0642adfa0fbef3b2e | Demo-vpc

Owner ID: 186824114219

Routes (3)

Destination	Target	Status	Propagated
pl-78a54011	vpce-08aae3d113506f3dc	Active	No



VPC > Route tables > rtb-093035ff5f6315e43 > Edit routes

Edit routes

Destination	Target	Status	Propagated
pl-78a54011	vpce-08aae3d113506f3dc	Active	No
10.0.0.0/16	local	Active	No
0.0.0.0/0	nat-0f4b2300451282ef0		

Step 7: Create a Security Group (Optional)

PHASE 2.1: Launch EC2 on Public Subnet

- Go to EC2 Dashboard
- Launch Instance
- Enter Name: public-subnet-1
- Select AMI: Windows
- Instance Type: t2.micro (FREE Tier)
- Create Key-Pair: demo
- Select Key-Pair: demo
- Edit Network Setting → Select Your VPC → Select Public Subnet → Enable Auto-Assign Public IP → Select existing Security Group
- Open Advanced Network Configuration → Enter Description: Angular → Primary IP: 10.0.1.4
- Launch Instance

PHASE 2.2: Launch EC2 on Private Subnet

- Go to EC2 Dashboard
- Launch Instance
- Enter Name: private-subnet-1
- Select AMI: Windows
- Instance Type: t2.micro (FREE Tier)
- Create Key-Pair: demo
- Select Key-Pair: demo
- Edit Network Setting → Select Your VPC → Select Private Subnet → Disable Auto-Assign Public IP → Select existing Security Group
- Open Advanced Network Configuration → Enter Description: FastAPI → Primary IP: 10.0.2.4
- Launch Instance

PHASE 2.3: Connect Public EC2 using RDP Client

PHASE 2.3: Connect Private EC2 using RDP Client from Public EC2 Desktop

- Check for Internet Access → Go to Command Prompt → Type ping www.google.com → Check Response → Request Timed Out (No Internet Access)
- Add NAT Gateway (see PHASE 1 → Step 4) if Internet Access required to install Software
- Delete NAT Gateway if Internet is not required

  Go to VPC Dashboard → Select NAT Gateway → Select Actions → Delete NAT Gateway → Delete Elastic IP → Go to EC2 Dashboard → Select Elastic IP → Release Elastic IP Address

Note: If NAT Gateway is added, Cost will be incurred per Hour of usage

---

Create VPC and Launch Linux EC2

(Reference: https://www.hostdime.com/kb/hd/linux-server/connect-using-putty-to-a-linux-server)

Procedure:

Step 1: Create VPC

Step 2: Launch Linux EC2 (Public and Private)

Step 3: Connect Public Linux EC2

1. Download Putty from https://www.chiark.greenend.org.uk/~sgtatham/putty/
2. Convert **.ppm** to **.ppk**
   using puTTYgen
3. Connect Public EC2
   using PuTTY
4. Connect Private EC2 from Public EC2
   ssh ec2-user@10.0.2.4
5. ping www.google.com