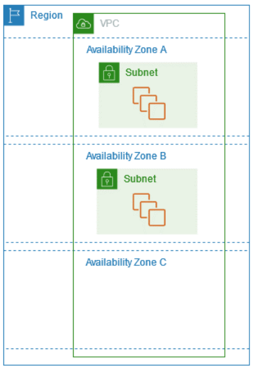
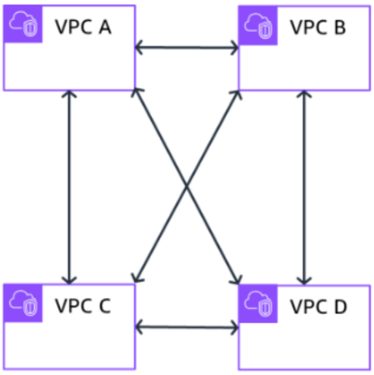
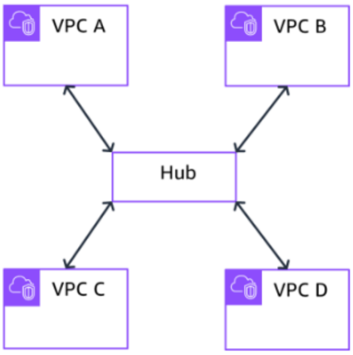
NETWORKING

A. Vocabulary

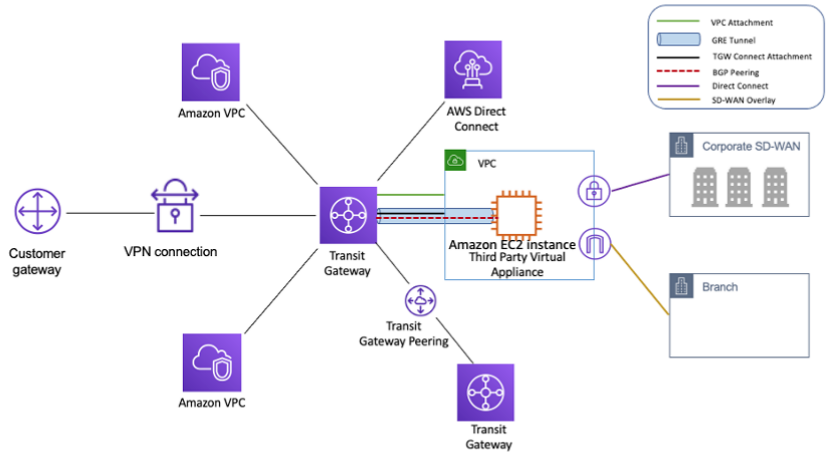
Regions	Isolated from other regions. Greatest possible fault tolerance and stability (us-east-1).
Availability Zone (AZ)	Isolated locations within a region (us-east-1a). VPC can cross multiple AZ.  Region VPC Availability Zone A Subnet Availability Zone B Subnet Availability Zone C

B. Network design

 	**Full mesh architecture** Work for small number of VPC Does not expand well Number of connections N(N-1)/2
 	**Hub and Spoke architecture** Central hub manages connectivity Works for large number of VPC Number of connections N More latency due to Hub

C. Transit Gateway

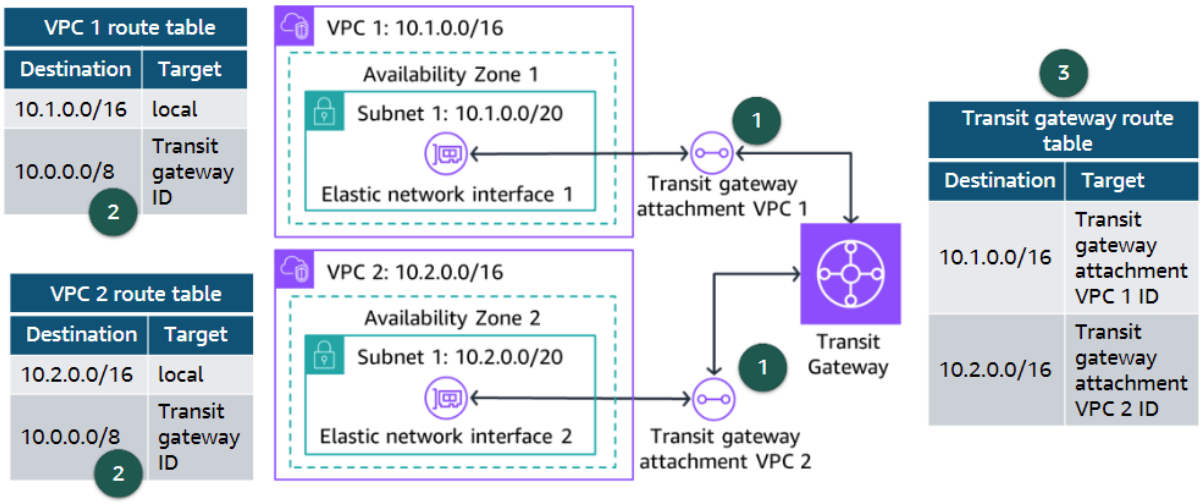
Transit Gateway provides Hub and Spoke design for connecting VPC and on-premises networks.



Hub and spoke design with AWS Transit Gateway

Manages service	Yes (high availability and scalability).
Charges	Per hour for the number of connections and the amount of traffic.
Routing	Dynamic Requires routers to discover routing paths Static Routes configured before traffic can be routed
IP addresses	IPv4 and IPv6
Logs	Transit Gateway Flow Logs to CloudWatch, Amazon S3, Kinesis Data Firehose

Example – No Internet Access



VPC 1 route table

Destination	Target
10.1.0.0/16	local
10.0.0.0/8	Transit gateway ID

VPC 2 route table

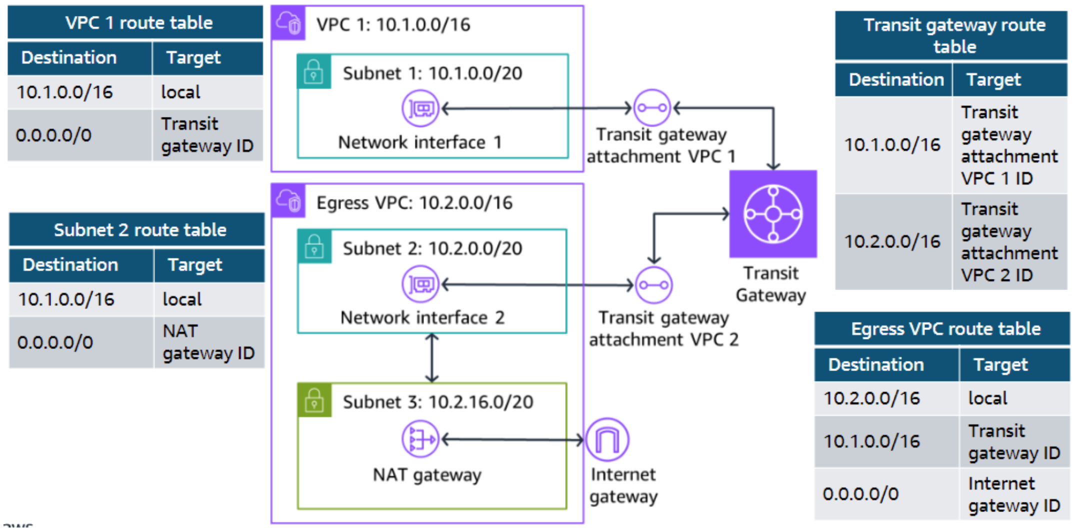
Destination	Target
10.2.0.0/16	local
10.0.0.0/8	Transit gateway ID

Transit gateway route table

Destination	Target
10.1.0.0/16	Transit gateway attachment VPC 1 ID
10.2.0.0/16	Transit gateway attachment VPC 2 ID

1. Connect the VPC to the **Transit Gateway** using a **Transit Gateway Attachment** through an **Elastic Network Interface** (like a network card).
2. Add a route for the Transit Gateway. In this case, 10.0.0.0/8 includes 10.X.0.0/16 (10.0.0.0/8 -> 10.0.0.0/10.255.255.255 / 10.X.0.0/16 -> 10.X.0.0/10.X.255.255). Use this tool
3. Configure the Transit Gateway route table to route the traffic to the correct VPN.

Example – With Internet Access



VPC 1 route table

Destination	Target
10.1.0.0/16	local
0.0.0.0/0	Transit gateway ID

Subnet 2 route table

Destination	Target
10.1.0.0/16	local
0.0.0.0/0	NAT gateway ID

Transit gateway route table

Destination	Target
10.1.0.0/16	Transit gateway attachment VPC 1 ID
10.2.0.0/16	Transit gateway attachment VPC 2 ID

Egress VPC route table

Destination	Target
10.2.0.0/16	local
10.1.0.0/16	Transit gateway ID
0.0.0.0/0	Internet gateway ID

Internet access is obtained using the **NAT Gateway** in the PUBLIC subnet 3.
The **NAT Gateway** is **NOT** in a separate VPC.
VPC 1 and VPC 2 route table sends all traffic 0.0.0.0/0 (except local one) to:

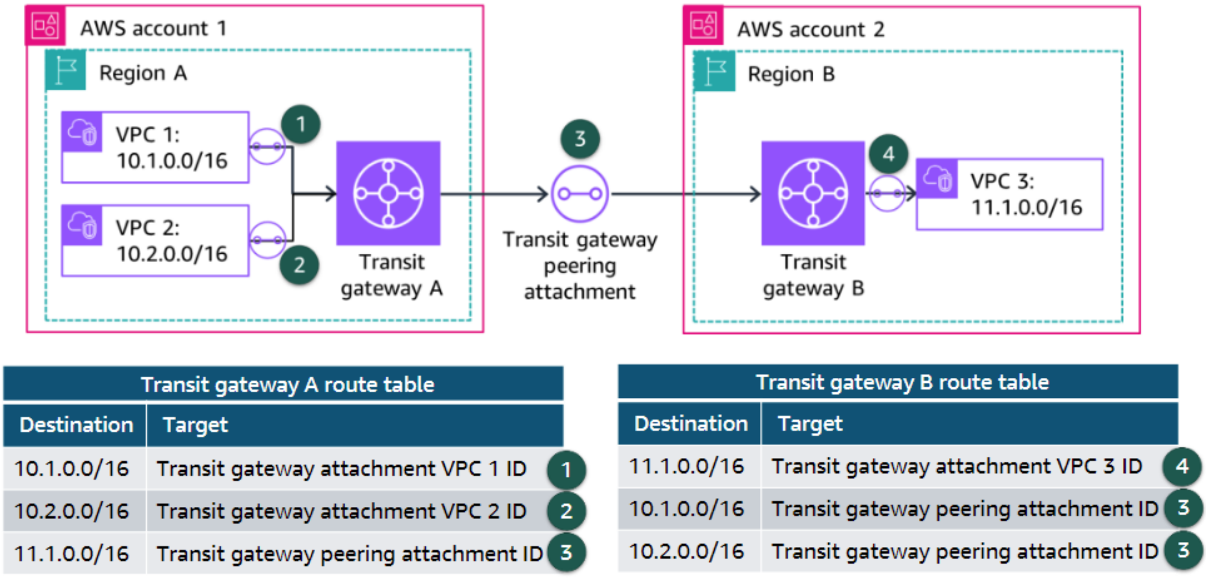
- Transit Gateway for VPC 1
- NAT Gateway for VPC 2

The **Transit Gateway** is responsible to route the traffic between VPC 1 and VPC 2.
The **NAT Gateway** routes VPC 1 traffic back to the **Transit Gateway**, and all other traffic 0.0.0.0/0 to the **Internet Gateway**.

This design is cheaper and simpler to use.
For redundancy, you can run a NAT Gateway for each Availability zone.

D. Peering

If you need network traffic to flow between AWS Regions or different AWS accounts, you can create a transit gateway peering connection between transit gateways. Traffic **DOES NOT** traverse the public internet (more secure)



Transit gateway A route table		
Destination	Target	
10.1.0.0/16	Transit gateway attachment VPC 1 ID	1
10.2.0.0/16	Transit gateway attachment VPC 2 ID	2
11.1.0.0/16	Transit gateway peering attachment ID	3

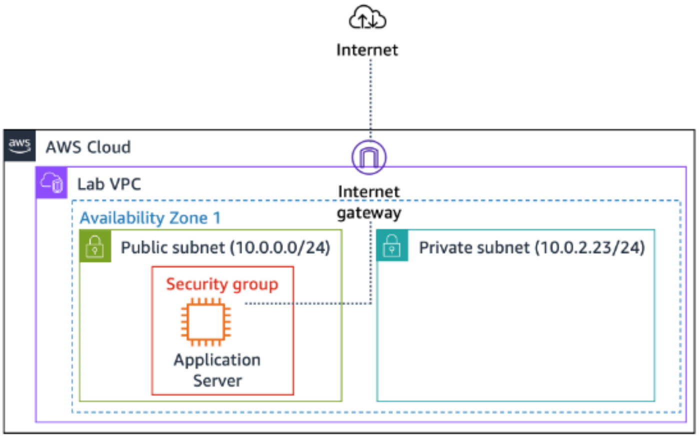
Transit gateway B route table		
Destination	Target	
11.1.0.0/16	Transit gateway attachment VPC 3 ID	4
10.1.0.0/16	Transit gateway peering attachment ID	3
10.2.0.0/16	Transit gateway peering attachment ID	3

VPC 1 and VPC 2 belongs to a different region and account than VPC 3.

Transit Gateway A routing (same idea for B):

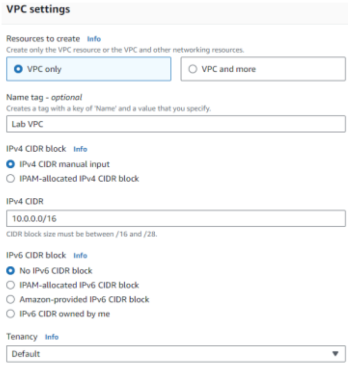
- 10.X.0.0/16 to the corresponding VPC
- 11.1.0.0/16 to the Transit Gateway B.

E. LAB - Creating a Virtual Private Cloud



1. Creating the VPC

Go to the **VPC Dashboard** (search VPC) -> Create VPC



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 - *optional*

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

Lab VPC

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

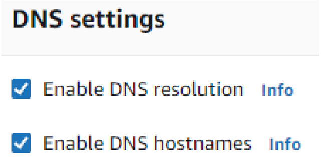
○ Amazon-provided IPv6 CIDR block

○ IPv6 CIDR owned by me

Tenancy Info

Default

Action -> **Edit VPC Settings**



DNS settings

☑ Enable DNS resolution Info

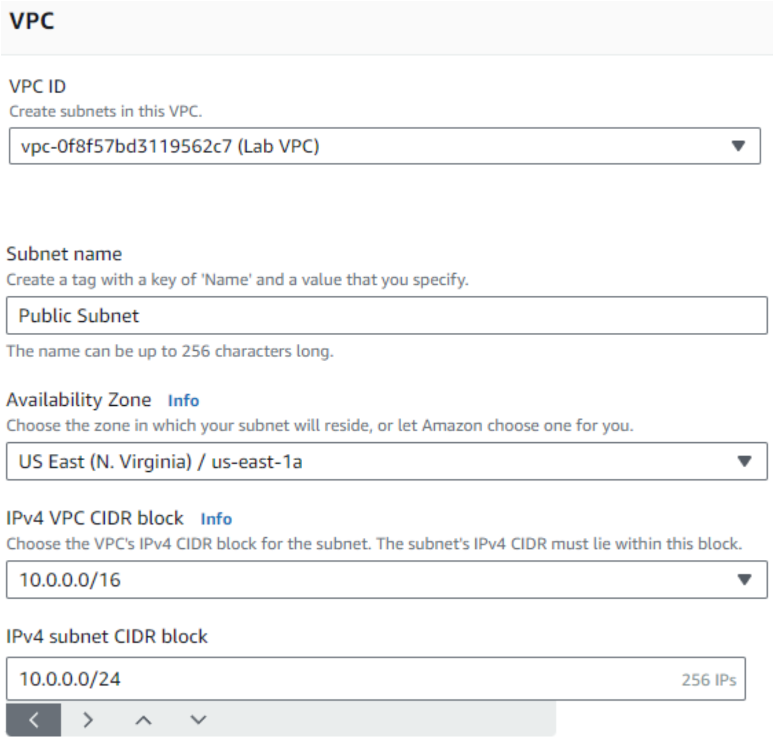
☑ Enable DNS hostnames Info

DNS resolution	Whether DNS resolution through the Amazon DNS server is supported for the VPC.
DNS hostnames	Any EC2 instances that are launched into the VPC now automatically receive a DNS hostname.

2. Creating the public subnet

In the **VPC Dashboard** -> Subnets -> Create Subnet

Choose the VPC it belongs to:



VPC

VPC ID
Create subnets in this VPC.
vpc-0f8f57bd3119562c7 (Lab VPC) ▼

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Public Subnet
The name can be up to 256 characters long.

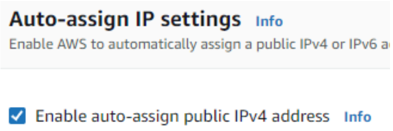
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a ▼

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 CIDR block of the subnet should be included in the VPC CIDR block.

Action -> **Edit Subnet Settings**



Auto-assign IP settings Info
Enable AWS to automatically assign a public IPv4 or IPv6 a

☑ Enable auto-assign public IPv4 address Info

Determines if, when you launch an EC2 instance, the primary network interface is assigned a public IPv4 address or IPv6 address by default. You can override this setting at instance level.

3. Creating the private subnet

Same step, but without editing the **Subnet Settings**.

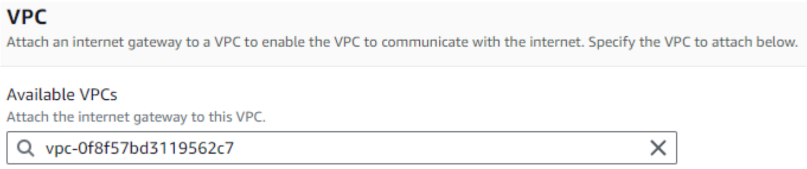
Remark: Now, there is hardly any difference between the 2 subnets. One will become public once it has a connection to the internet gateway.

4. Creating an Internet Gateway

In the **VPC Dashboard** -> Internet Gateway -> Create Internet Gateway.

Nothing special there.
Now we need to attach the **Internet Gateway** to the VPC.

Action -> **Attach to VPC**



VPC

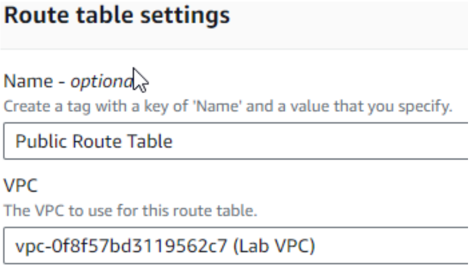
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs
Attach the internet gateway to this VPC.
vpc-0f8f57bd3119562c7

The Internet Gateway is attached, but no routing is configured yet.

5. Configuring routing tables

In the **VPC Dashboard** -> Route tables -> Create route table.

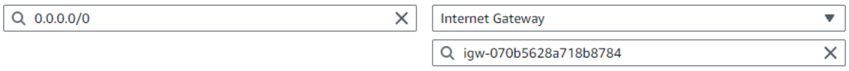


Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
Public Route Table

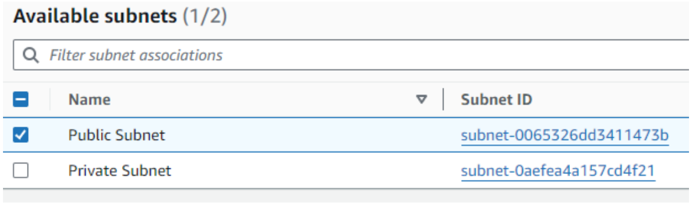
VPC
The VPC to use for this route table.
vpc-0f8f57bd3119562c7 (Lab VPC)

Action -> **Edit route**



0.0.0.0/0	Internet Gateway igw-070b5628a718b8784

Action -> **Edit subnet association**



Available subnets (1/2)

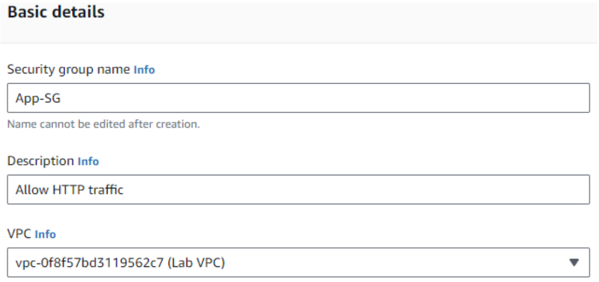
Filter subnet associations

⊟	Name	Subnet ID
☑	Public Subnet	subnet-0065326dd3411473b
☐	Private Subnet	subnet-0aefea4a157cd4f21

The public subnet has now access to the public internet.

F. Creating a security group for the application server

In the **VPC Dashboard** -> Security Groups -> Create security group.



Basic details

Security group name Info

App-SG

Name cannot be edited after creation.

Description Info

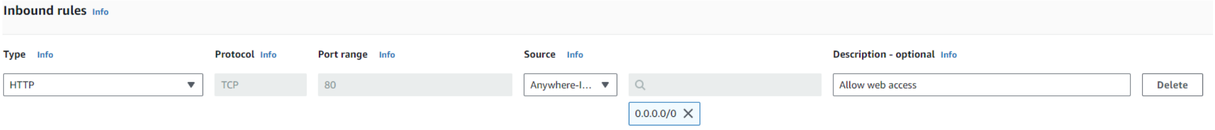
Allow HTTP traffic

VPC Info

vpc-0f8f57bd3119562c7 (Lab VPC) ▼

Attach the security group to the VPC.

Define the inbound traffic rules



Inbound rules Info

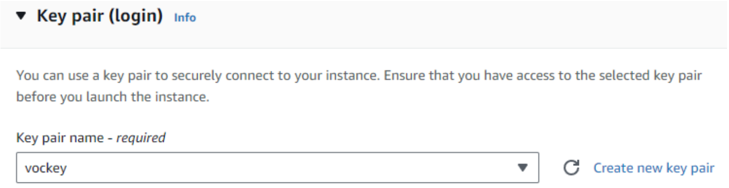
Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
HTTP ▼	TCP	80	Anywhere-I... ▼	0.0.0.0/0 ✕	Allow web access	Delete

We allow all inbound traffic on port 80.

G. Launching an application server in the public subnet

Go to the **EC2 Dashboard** (search EC2) -> Instances -> Launch instances

Choose the type of instance required for the workload.



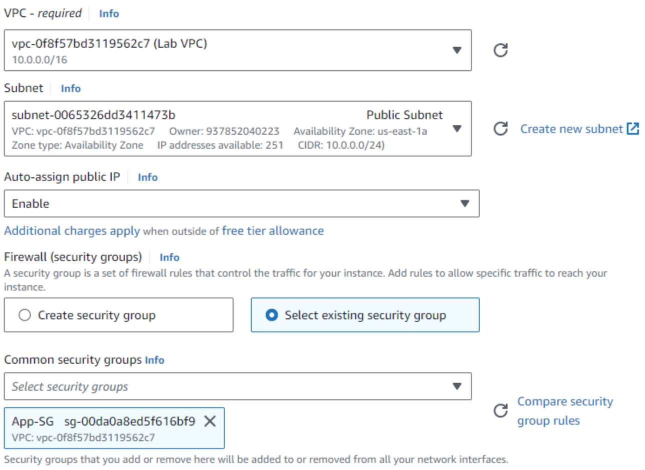
▼ **Key pair (login)** Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

vockey ▼

Network settings -> Edit



VPC - *required* Info

vpc-0f8f57bd3119562c7 (Lab VPC)
10.0.0.0/16 ▼

Subnet Info

subnet-0065326dd3411473b Public Subnet
VPC: vpc-0f8f57bd3119562c7 Owner: 937852040223 Availability Zone: us-east-1a
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.0.0/24) ▼

Auto-assign public IP Info

Enable ▼

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

○ Create security group ● Select existing security group

Common security groups Info

Select security groups ▼

App-SG sg-00da0a8ed5f616bf9 ✕
VPC: vpc-0f8f57bd3119562c7

Security groups that you add or remove here will be added to or removed from all your network interfaces.

You must link your instance with the VPC.

Then you choose the subnet (Public here)
Auto-assign public IP -> This is where you can override the choice made in E.1 (DNS Hostnames).