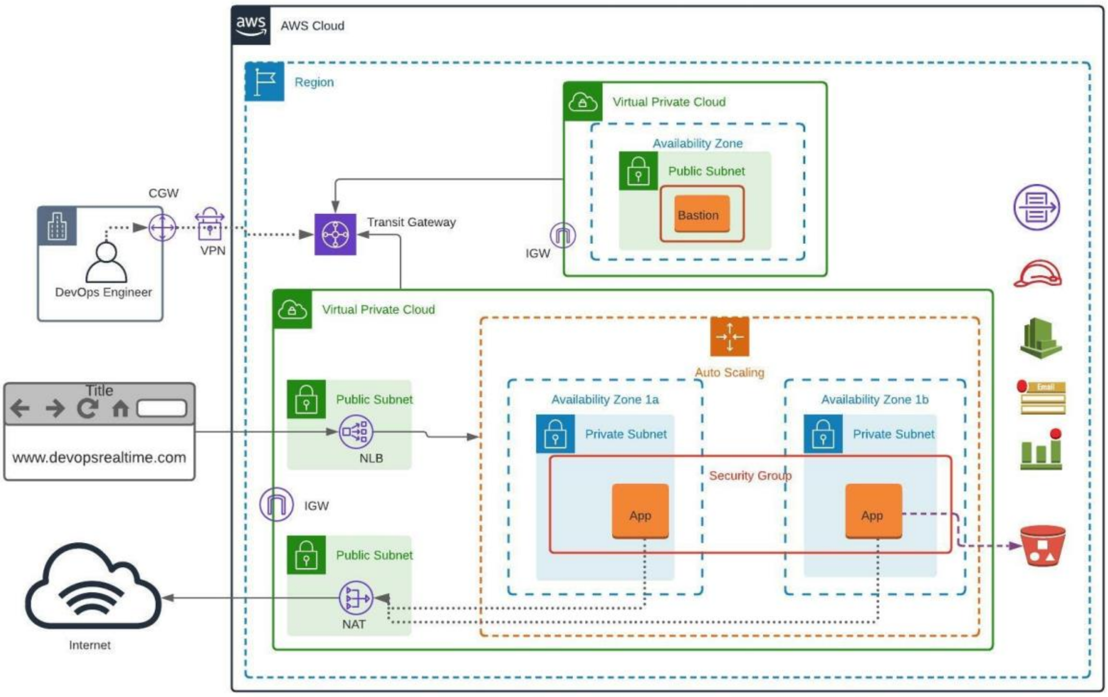
Our requirement is to access S3 service from CLI (Command Line Interface) in private server

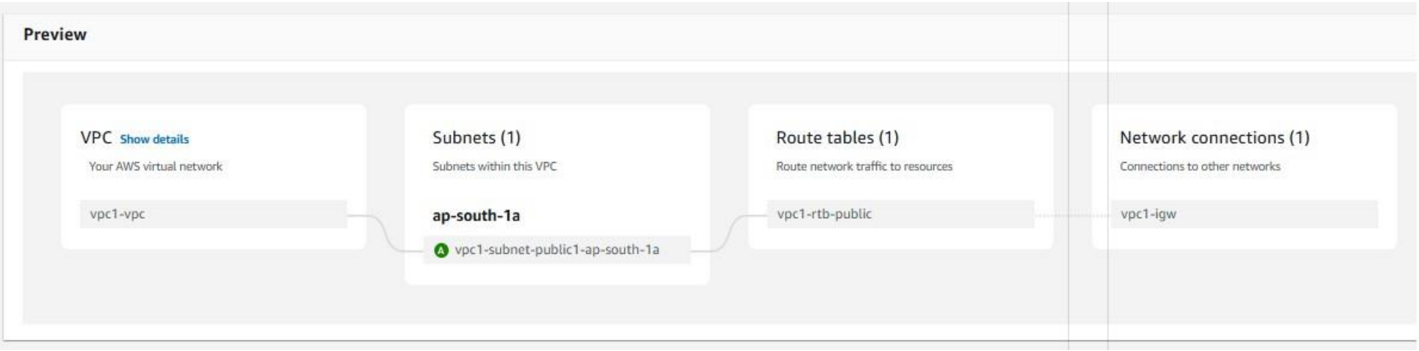
Requirement Architecture



Step to complete this task

Step 1:

1. Create two VPC's as shown in architecture
2. First create one VPC with
   - One availability zone
   - One public subnet
   - One public route table
   - One internet gateway as shown in below figure



Preview

VPC Show details
Your AWS virtual network
vpc1-vpc

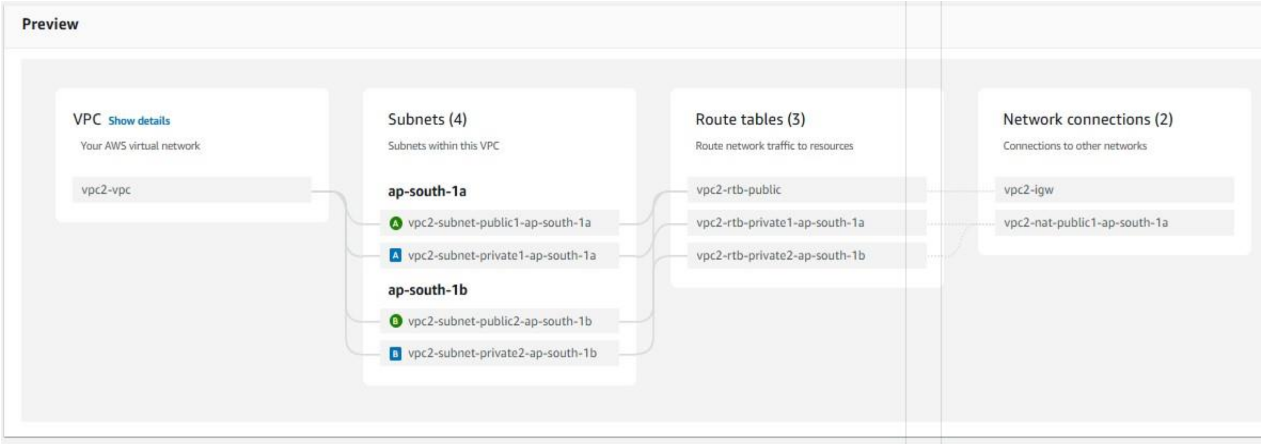
Subnets (1)
Subnets within this VPC
ap-south-1a
vpc1-subnet-public1-ap-south-1a

Route tables (1)
Route network traffic to resources
vpc1-rtb-public

Network connections (1)
Connections to other networks
vpc1-igw

Step 2:

1. Create second VPC with
    - Two availability zones
    - Two public subnets
    - Two private subnets
    - One public and two private route tables
    - One internet gateway
    - One NAT gateway as shown in below figure


Preview
VPC Show details
Your AWS virtual network
vpc2-vpc
Subnets (4)
Subnets within this VPC
ap-south-1a
A vpc2-subnet-public1-ap-south-1a
A vpc2-subnet-private1-ap-south-1a
ap-south-1b
B vpc2-subnet-public2-ap-south-1b
B vpc2-subnet-private2-ap-south-1b
Route tables (3)
Route network traffic to resources
vpc2-rtb-public
vpc2-rtb-private1-ap-south-1a
vpc2-rtb-private2-ap-south-1b
Network connections (2)
Connections to other networks
vpc2-igw
vpc2-nat-public1-ap-south-1a

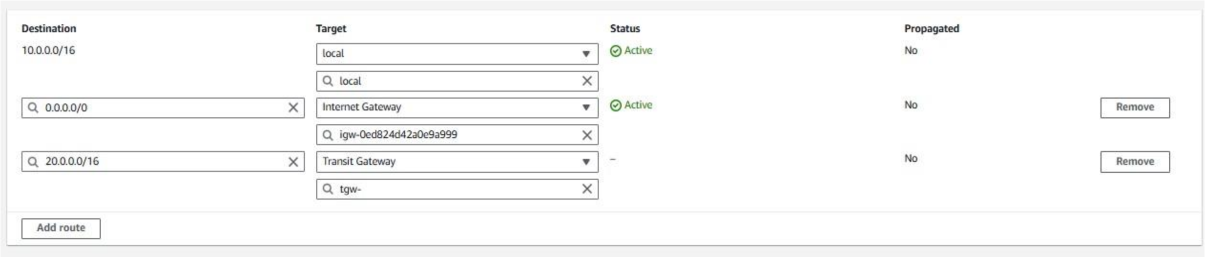
Step 3:

1. Create one transit gateway
2. Create two transit gateway attachments
    - Create the name for attachments
    - Select the transit gateway ID
    - Select the attachment type as VPC
    - Select the created VPC 1 in VPC ID
    - Create the second attachment same as first attachment
    - But select the created VPC 2 in VPC ID

Step 4:

1. Go to route tables
2. Edit the routes

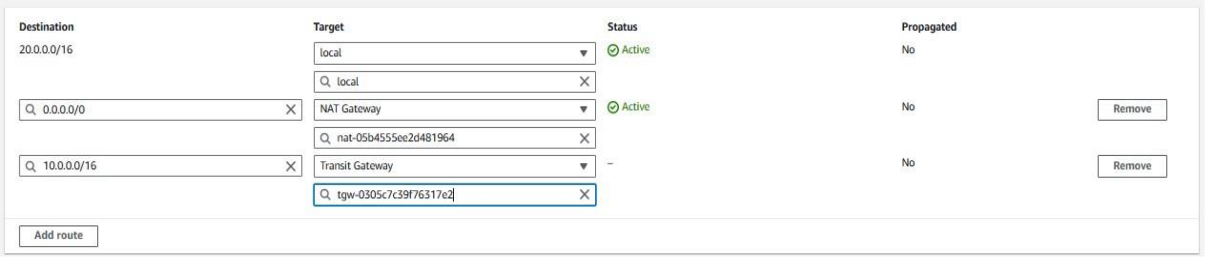
3. Select VPC 1 public route table
    - Click on edit routes in that
    - Click on add routes
    - Give VPC 2 CIDR range and select transit gateway as shown in below figure



Destination	Target	Status	Propagated	
10.0.0.0/16	local local	Active	No	
0.0.0.0/0	Internet Gateway igw-0ed824d42a0e9a999	Active	No	Remove
20.0.0.0/16	Transit Gateway tgw-	-	No	Remove

Add route

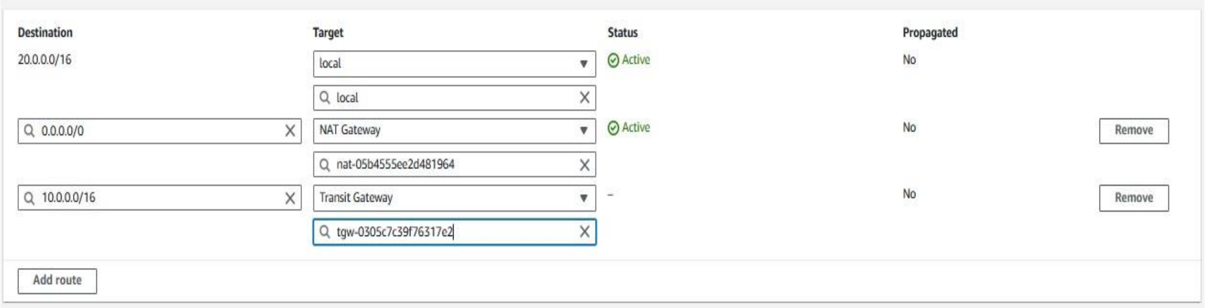
4. Select VPC 2 private 1 route table
    - Click on edit routes in that
    - Click on add routes
    - Give VPC 1 CIDR range and select transit gateway as shown in below figure



Destination	Target	Status	Propagated	
20.0.0.0/16	local local	Active	No	
0.0.0.0/0	NAT Gateway nat-05b4555ee2d481964	Active	No	Remove
10.0.0.0/16	Transit Gateway tgw-0305c7c39f76317e2	-	No	Remove

Add route

5. Select VPC 2 private 2 route table
    - Click on edit routes in that
    - Click on add routes
    - Give VPC 1 CIDR range and select transit gateway as shown in below figure

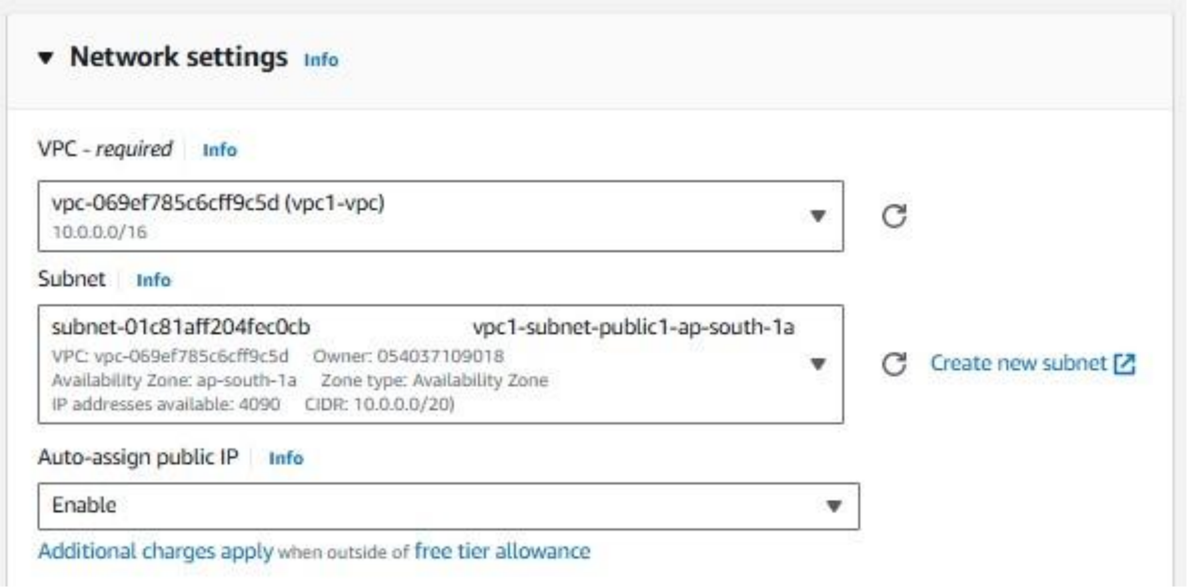


Destination	Target	Status	Propagated	
20.0.0.0/16	local local	Active	No	
0.0.0.0/0	NAT Gateway nat-05b4555ee2d481964	Active	No	Remove
10.0.0.0/16	Transit Gateway tgw-0305c7c39f76317e2	-	No	Remove

Add route

Step 5:

1. Select EC2 service
2. Create one instance as bastion host
    - Click on launch instance
    - Select the name as bastion host
    - Select the key pair
    - Edit the network settings
    - Select created VPC 1 in network settings
    - Enable the auto assign public IP



▼ Network settings Info

VPC - *required* Info

vpc-069ef785c6cff9c5d (vpc1-vpc)
10.0.0.0/16

Subnet Info

subnet-01c81aff204fec0cb vpc1-subnet-public1-ap-south-1a
VPC: vpc-069ef785c6cff9c5d Owner: 054037109018
Availability Zone: ap-south-1a Zone type: Availability Zone
IP addresses available: 4090 CIDR: 10.0.0.0/20)

Auto-assign public IP Info

Enable

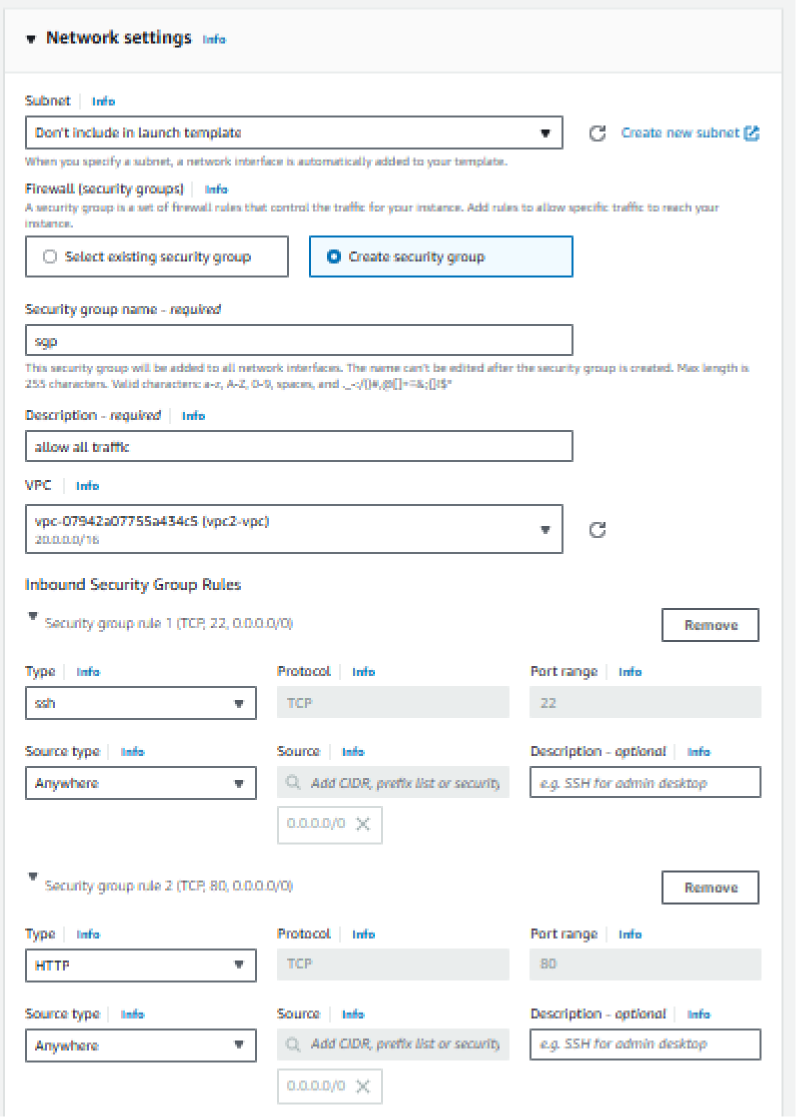
Additional charges apply when outside of free tier allowance

- Click on launch instance

Step 6:

1. Select launch templets in EC2
2. Click on create launch templet
    - Give the name for launch template
    - Choose the instance type as free tire
    - Select the created key pair or generate a new key pair
    - Update the network settings by clicking on edit option

- No need to change subnet
- Click on create security group
- Enter the security group name
- Enter the description it’s your choice
- Select the created VPC 2
- Update the inbound rules
- One is ssh and another is http port 80
- Click on launch the templates as shown in below figure



▾ Network settings Info

Subnet | Info

Don't include in launch template ▾

Create new subnet

When you specify a subnet, a network interface is automatically added to your template.

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.

○ Select existing security group ● Create security group

Security group name - *required*

sgp

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

Description - *required* | Info

allow all traffic

VPC | Info

vpc-07942a07755a434c5 (vpc2-vpc)
20.0.0.0/16

Inbound Security Group Rules

▾ Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

Type: ssh
Protocol: TCP
Port range: 22
Source type: Anywhere
Source: *Add CIDR, prefix list or security*
0.0.0.0/0 ✕
Description - *optional*: *e.g. SSH for admin desktop*

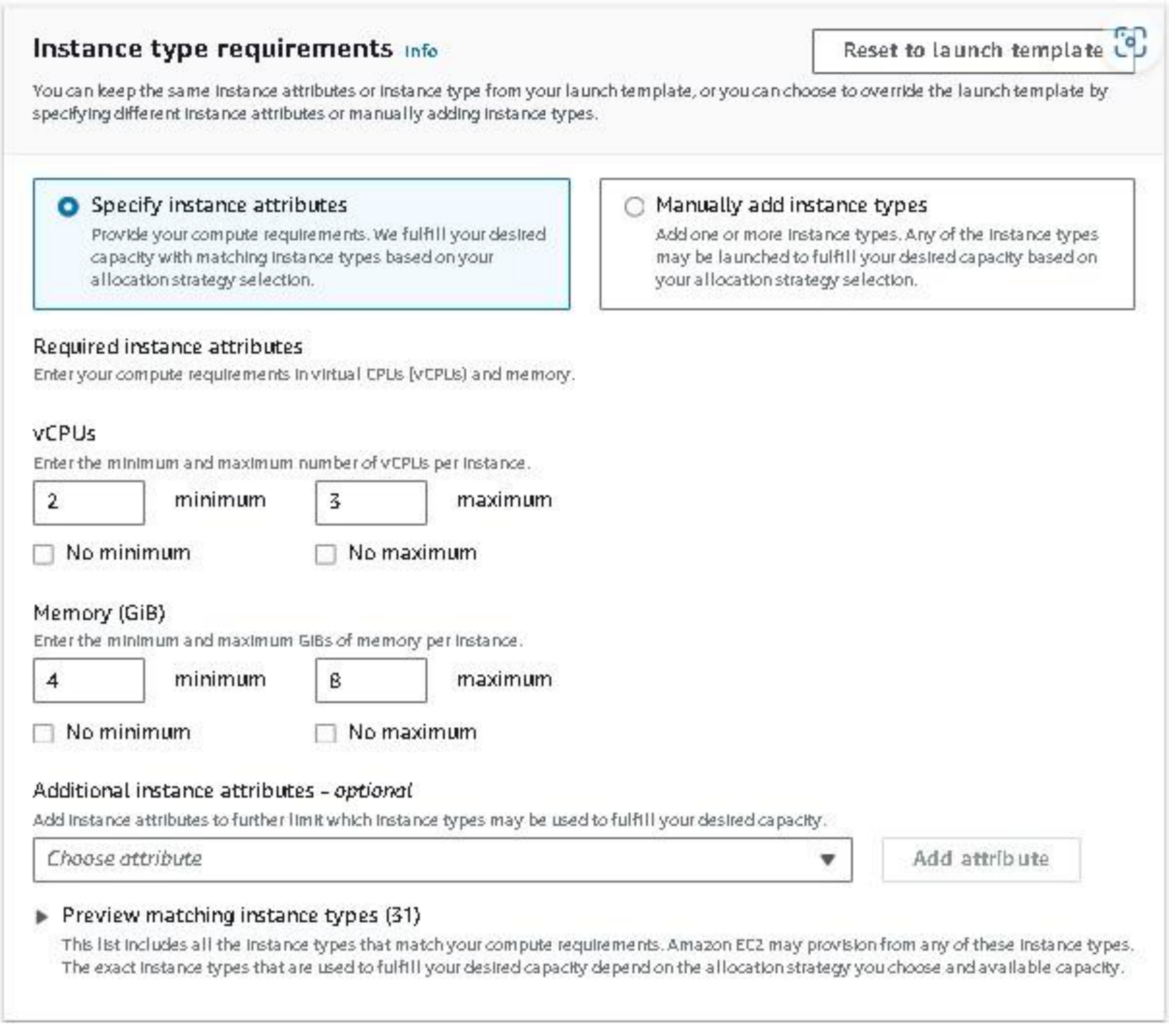
▾ Security group rule 2 (TCP, 80, 0.0.0.0/0) Remove

Type: HTTP
Protocol: TCP
Port range: 80
Source type: Anywhere
Source: *Add CIDR, prefix list or security*
0.0.0.0/0 ✕
Description - *optional*: *e.g. SSH for admin desktop*

Step 7:

1. Click on auto scaling groups

2. Click on create auto scaling group
3. Give the name for auto scaling group
4. Select the created launch templates
5. Choose instance type requirement
    - Choose VCPUs minimum and maximum
    - (2 is minimum) and (3 is maximum)
    - Choose memory minimum and maximum
    - (4 is minimum) and (8 is maximum)
    - As shown in below figure



Instance type requirements Info

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

● Specify instance attributes
Provide your compute requirements. We fulfill your desired capacity with matching instance types based on your allocation strategy selection.

○ Manually add instance types
Add one or more instance types. Any of the instance types may be launched to fulfill your desired capacity based on your allocation strategy selection.

Required instance attributes
Enter your compute requirements in virtual CPUs (vCPUs) and memory.

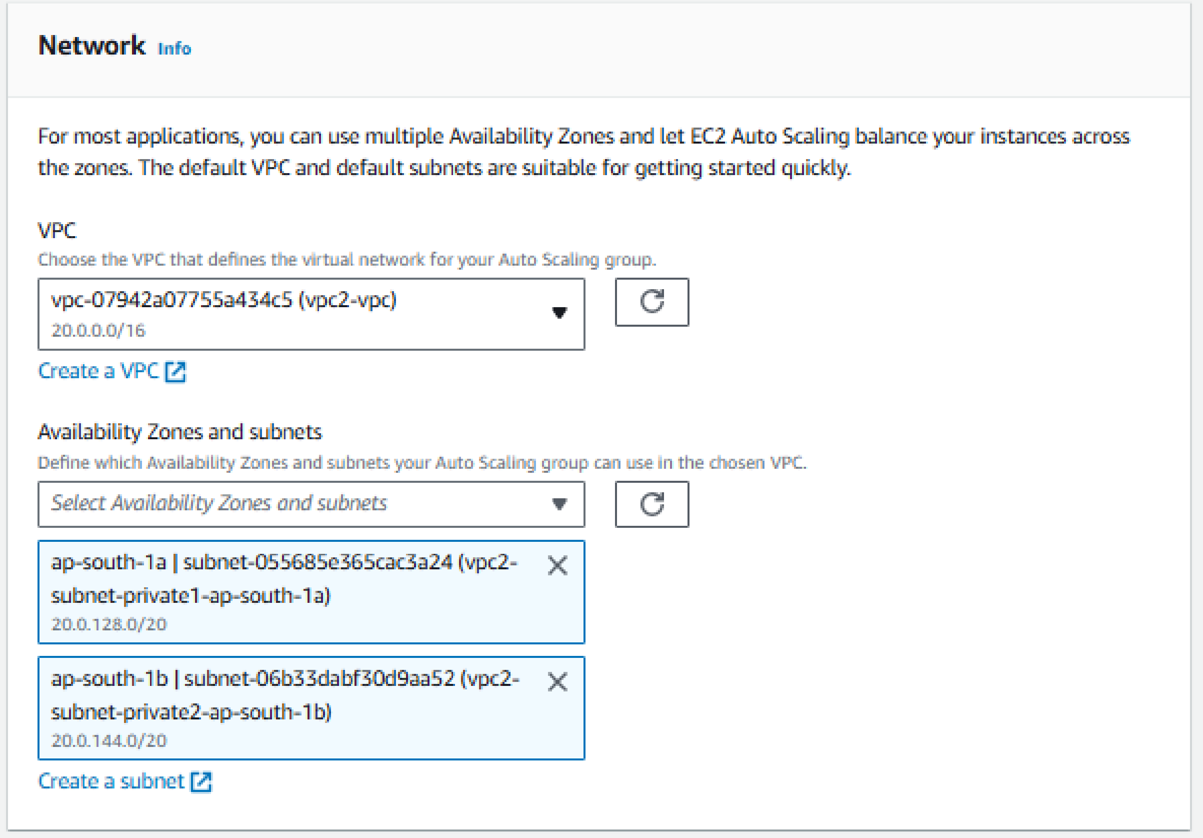
vCPUs
Enter the minimum and maximum number of vCPUs per instance.
2 minimum 3 maximum
□ No minimum □ No maximum

Memory (GiB)
Enter the minimum and maximum GiBs of memory per instance.
4 minimum 8 maximum
□ No minimum □ No maximum

Additional instance attributes - *optional*
Add instance attributes to further limit which instance types may be used to fulfill your desired capacity.

▶ Preview matching instance types (31)
This list includes all the instance types that match your compute requirements. Amazon EC2 may provision from any of these instance types. The exact instance types that are used to fulfill your desired capacity depend on the allocation strategy you choose and available capacity.

6. Select created VPC 2
7. Select 2 private availability zones as shown in below figure



Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-07942a07755a434c5 (vpc2-vpc)
20.0.0.0/16

Create a VPC

Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

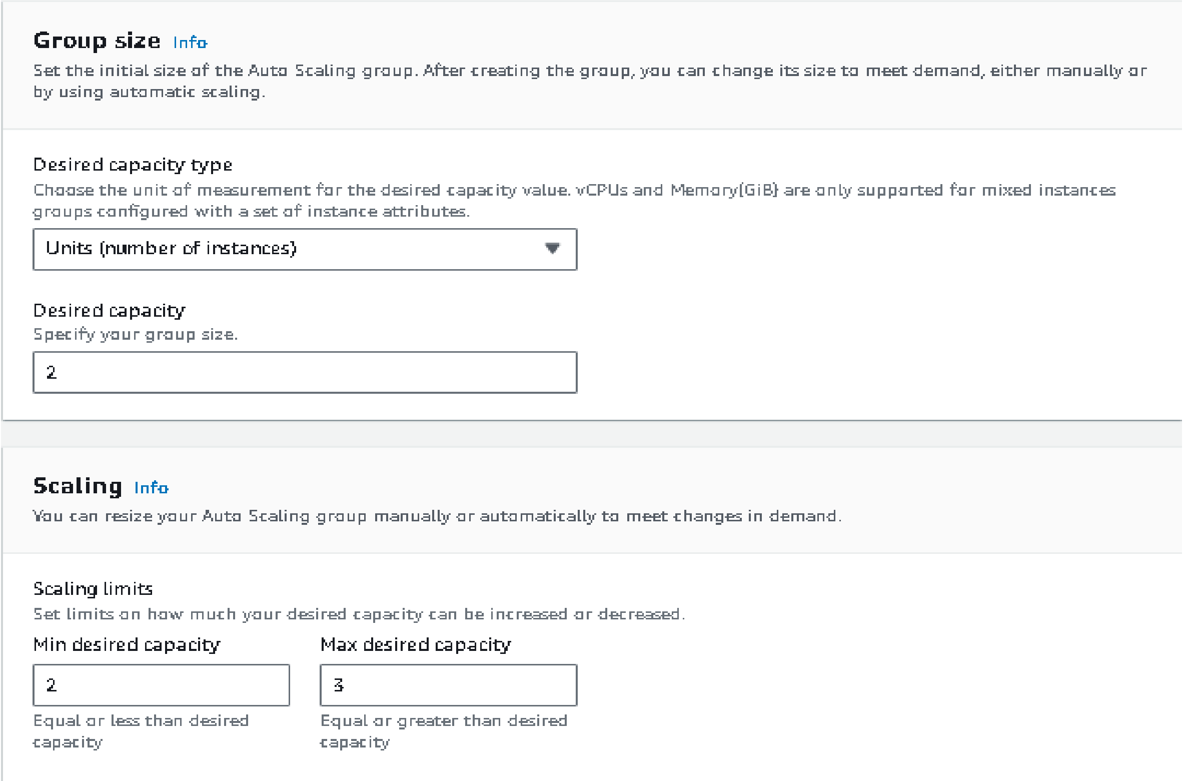
Select Availability Zones and subnets

ap-south-1a | subnet-055685e365cac3a24 (vpc2-subnet-private1-ap-south-1a)
20.0.128.0/20

ap-south-1b | subnet-06b33dabf30d9aa52 (vpc2-subnet-private2-ap-south-1b)
20.0.144.0/20

Create a subnet

8. No need to select load balancer leave as no load balancer
9. Choose desired capacity and scaling option as shown in below figure



Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

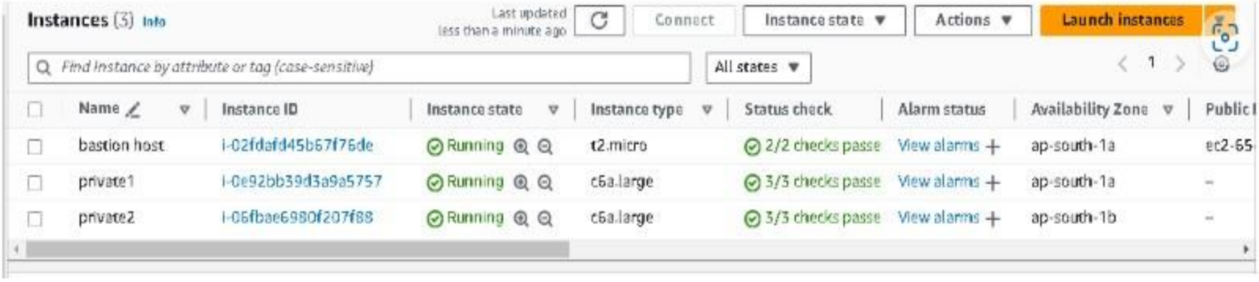
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity: 2
Equal or less than desired capacity

Max desired capacity: 3
Equal or greater than desired capacity

Step 8:

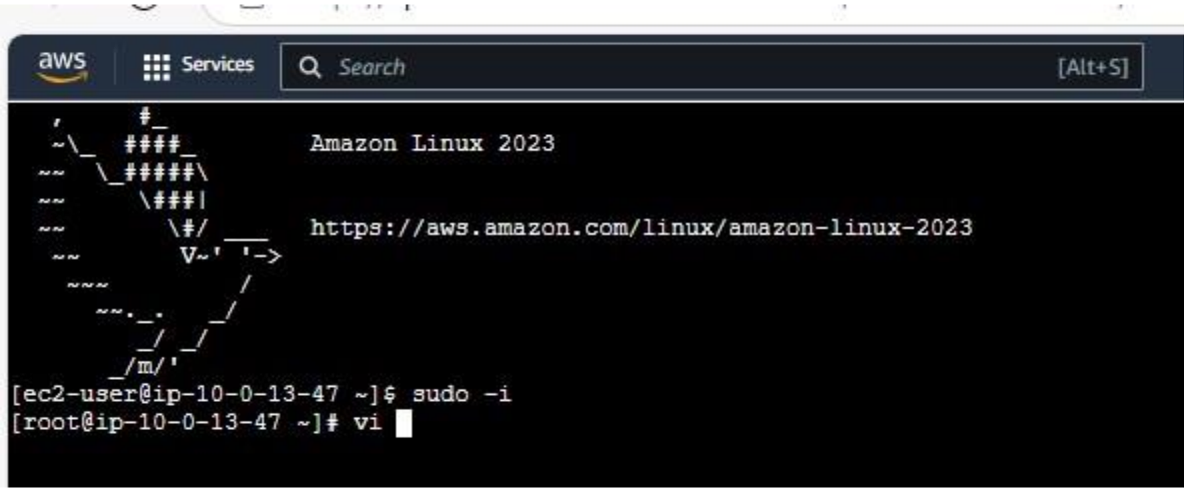
1. Go to instance
2. Check the instance their 2 private instances are created from auto scaling group
3. Give the name for 2 private instances as private 1 and private 2 to avoid confusions as shown in figure



Instances (3) Info
Last updated less than a minute ago
Connect
Instance state ▼
Actions ▼
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states ▼
1

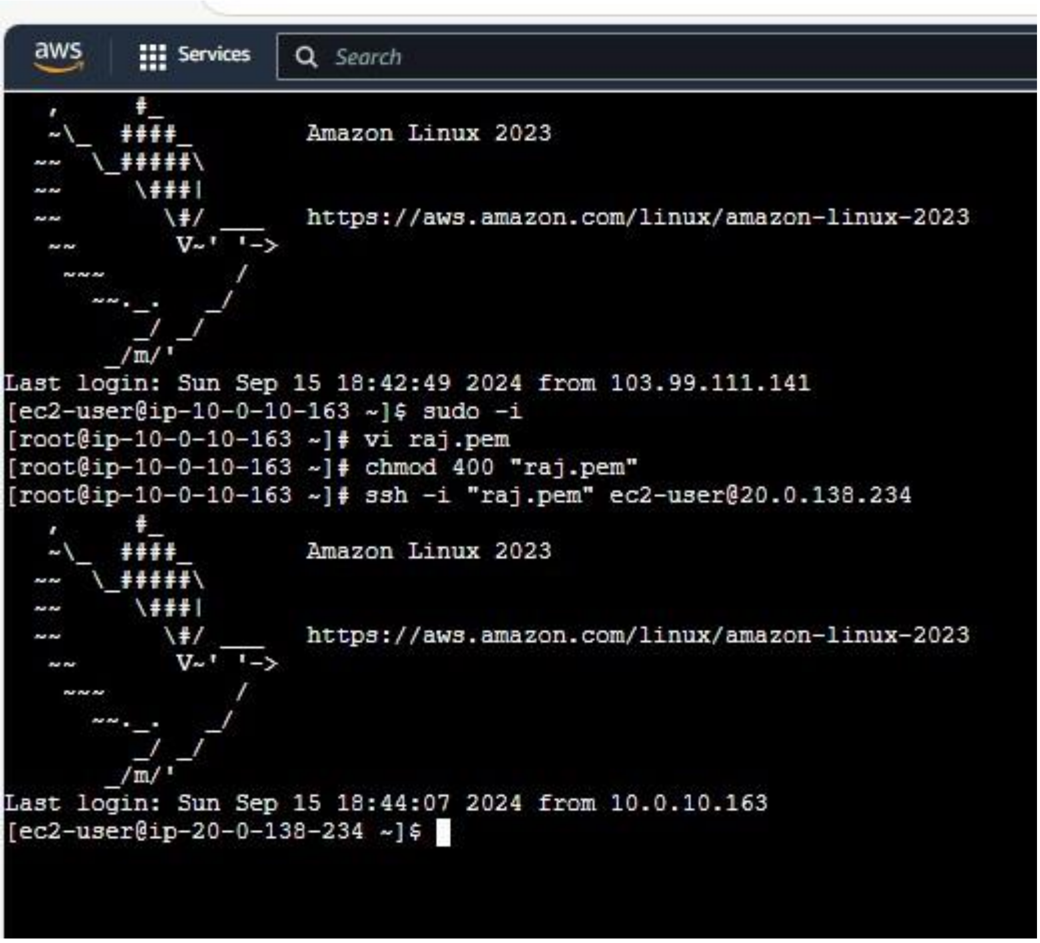
□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public I
□	bastion host	i-02fdafd45b67f76de	✓ Running	t2.micro	✓ 2/2 checks passe	View alarms +	ap-south-1a	ec2-65
□	private1	i-0e92bb39d3a9a5757	✓ Running	c6a.large	✓ 3/3 checks passe	View alarms +	ap-south-1a	–
□	private2	i-06fbae6980f207f88	✓ Running	c6a.large	✓ 3/3 checks passe	View alarms +	ap-south-1b	–

4. Connect the bastion host through external terminal or direct connect as shown in below figure



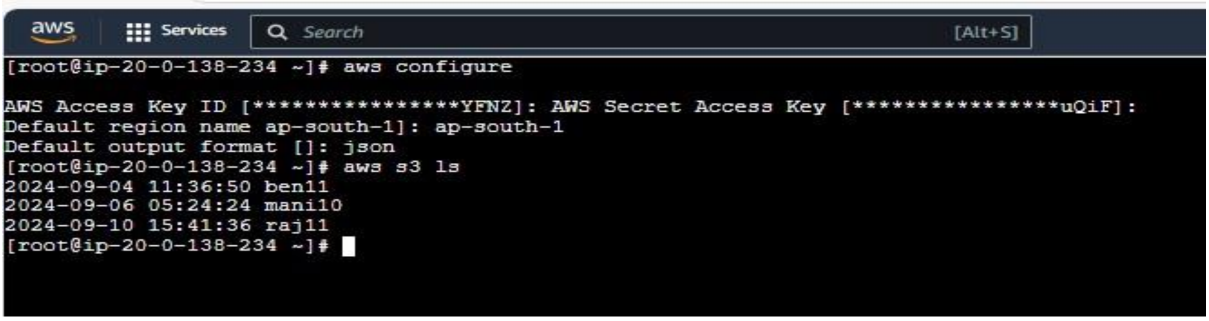
```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-13-47 ~]$ sudo -i
[root@ip-10-0-13-47 ~]# vi
```

5. Select the private 1 and connect through ssh client
   - Go to bastion sever
   - Click vi key pair and copy the key pair and paste in editor
   - Click chmod 400 key pair
   - Copy the example and paste bastion host it will switch from bastion host to private 1 server
   - As shown in below figure



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sun Sep 15 18:42:49 2024 from 103.99.111.141
[ec2-user@ip-10-0-10-163 ~]$ sudo -i
[root@ip-10-0-10-163 ~]# vi raj.pem
[root@ip-10-0-10-163 ~]# chmod 400 "raj.pem"
[root@ip-10-0-10-163 ~]# ssh -i "raj.pem" ec2-user@20.0.138.234
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sun Sep 15 18:44:07 2024 from 10.0.10.163
[ec2-user@ip-20-0-138-234 ~]$
```

- Click AWS configure and provide access key and secret key and choose region and format
- To list the buckets from CLI just click AWS S3 LS
- Then you see the all buckets in S3 as shown in below figure



```
[root@ip-20-0-138-234 ~]# aws configure

AWS Access Key ID [****************YFNZ]: AWS Secret Access Key [****************uQiF]:
Default region name ap-south-1]: ap-south-1
Default output format []: json
[root@ip-20-0-138-234 ~]# aws s3 ls
2024-09-04 11:36:50 ben11
2024-09-06 05:24:24 mani10
2024-09-10 15:41:36 raj11
[root@ip-20-0-138-234 ~]#
```

6. Repeat the same process for private 2 server