AWS VPC Creation SOP

Cognizant

Table of Contents

1. AWS VPC Creation

1.1 Description

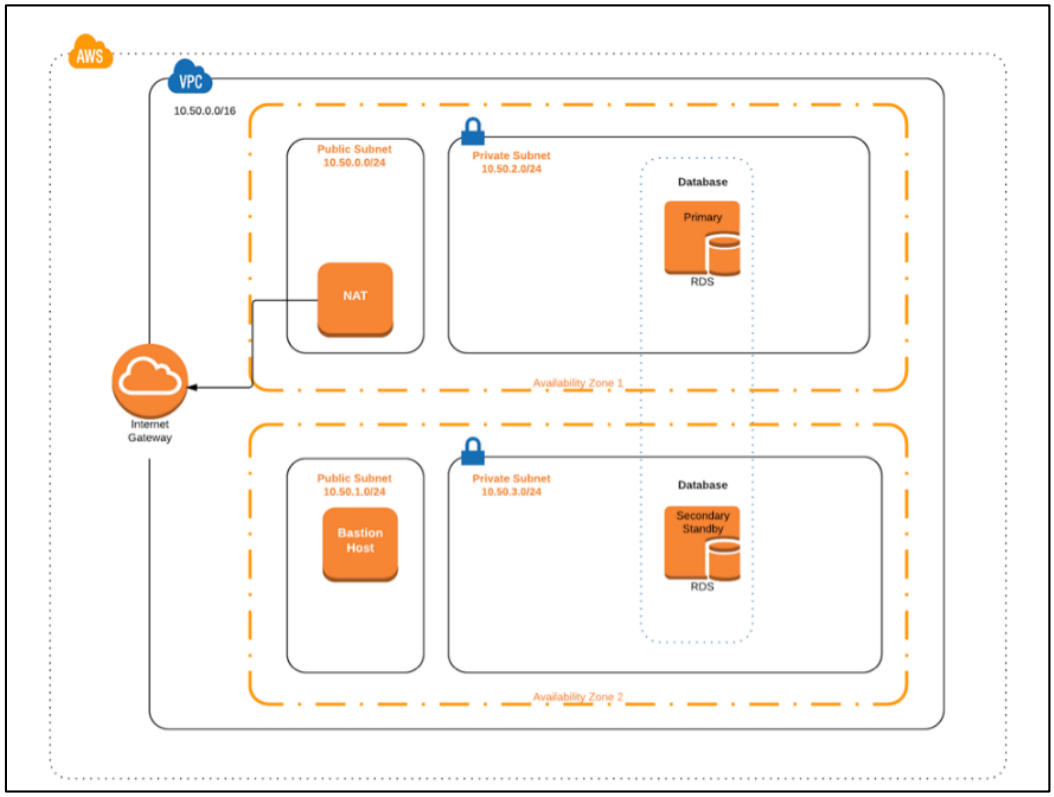
The VPC template is the foundation for everything you build on AWS with the Startup Kit. It creates a VPC with the following network resources:

- Two public subnets, which have routes to a public Internet gateway.
- Two private subnets, which do NOT have routes to the public Internet gateway.
- A NAT Gateway to allow instances in private subnets to communicate with the public Internet, for example, to pull down patches and upgrades, and access AWS services that have public endpoints (though some AWS services may be accessed entirely privately).
- Two route tables, one for public subnets and the other for private subnets.
- Security groups for an app, load balancer, database, and bastion host.

The bastion host template creates a bastion host that provides SSH access to resources you place in private subnets for greater security. Resources placed in private subnets could include application instances, database instances, analytics clusters, and other resources you do not want to be discoverable via the public Internet. For example, along with enabling proper authentication and authorization controls, placing database instances in private subnets can help avoid security problems risked by exposing databases to the public Internet.
After you have created your VPC and bastion host, you can optionally create a relational database using the database template. Either a MySQL or PostgreSQL database is created in the Amazon Relational Database Service (Amazon RDS), which automates much of the heavy lifting of database setup and maintenance. Following best practices, the database is created in your VPC's private subnets and is concealed from the public Internet.

1.2 Architecture Diagram

The diagram below displays a visual representation of the underlying application architecture:



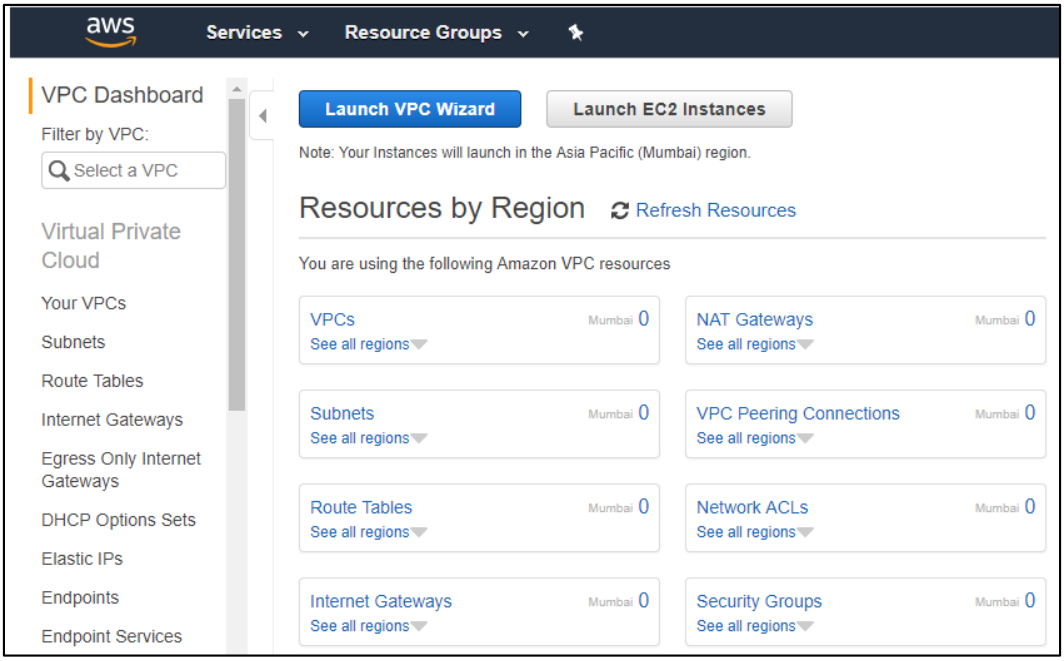
1.3 Lab Steps

Follow the steps outlined below to achieve the objectives of this lab exercise:

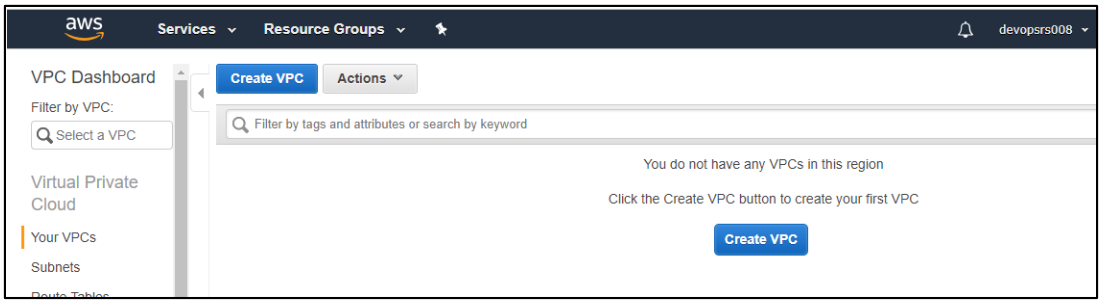
Prerequisites:

- Sign in to the AWS Management Console.
- An Amazon EC2 key pair, if you do not have one.

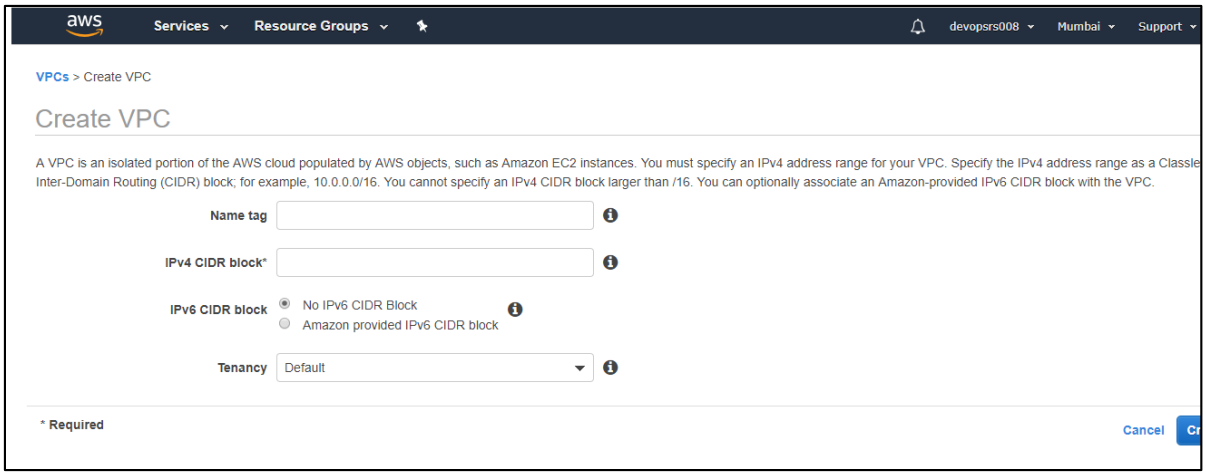
1. Create a VPC:



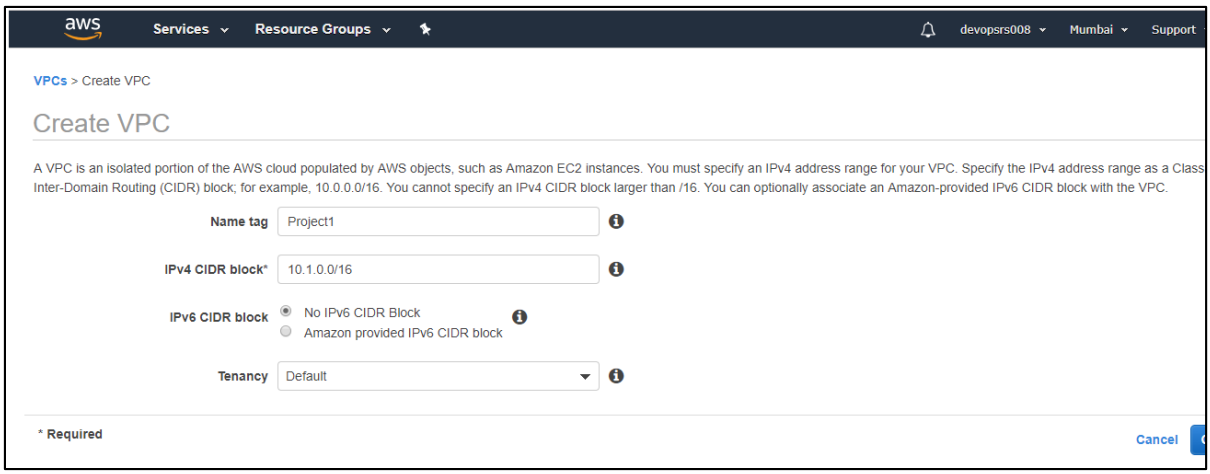
a. Click **Your VPCs** menu:

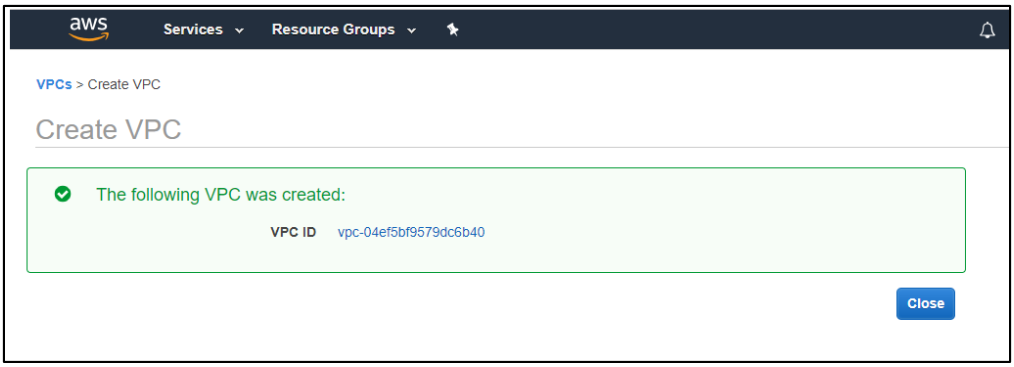


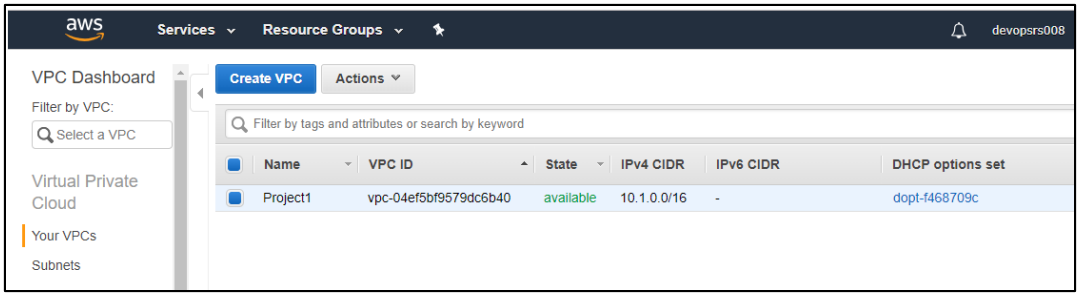
b. Click **Create VPC.**



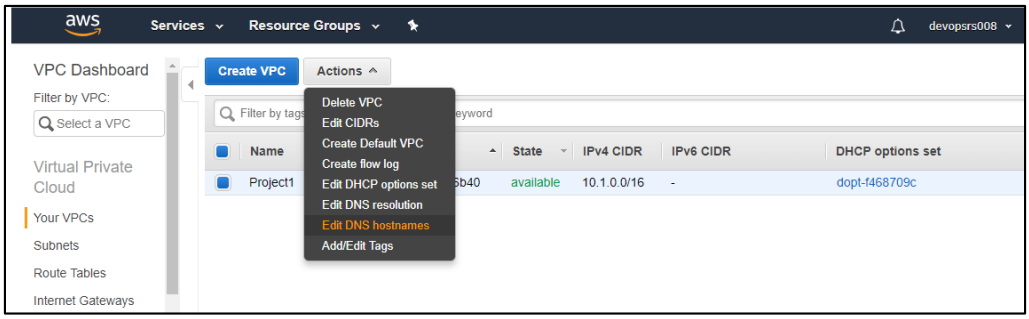
c. Mention the tag and IPv4 address, and click **Create** button:





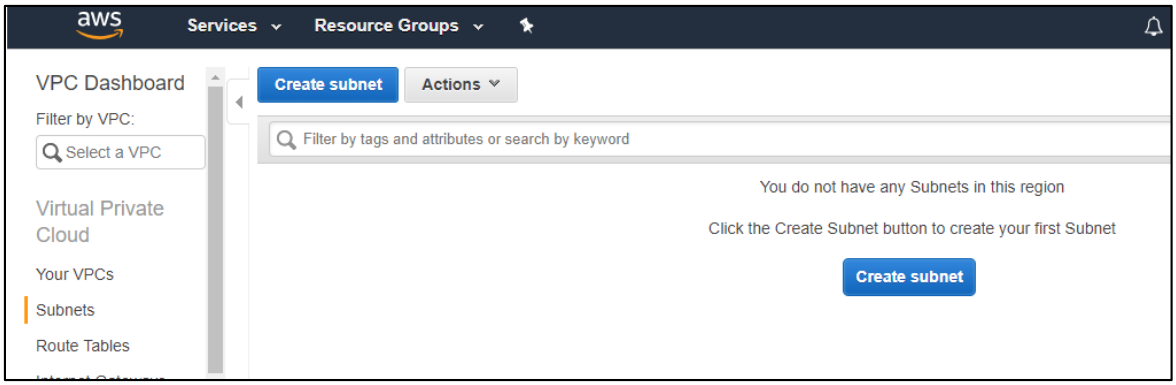


d. Enable the DNS host names:

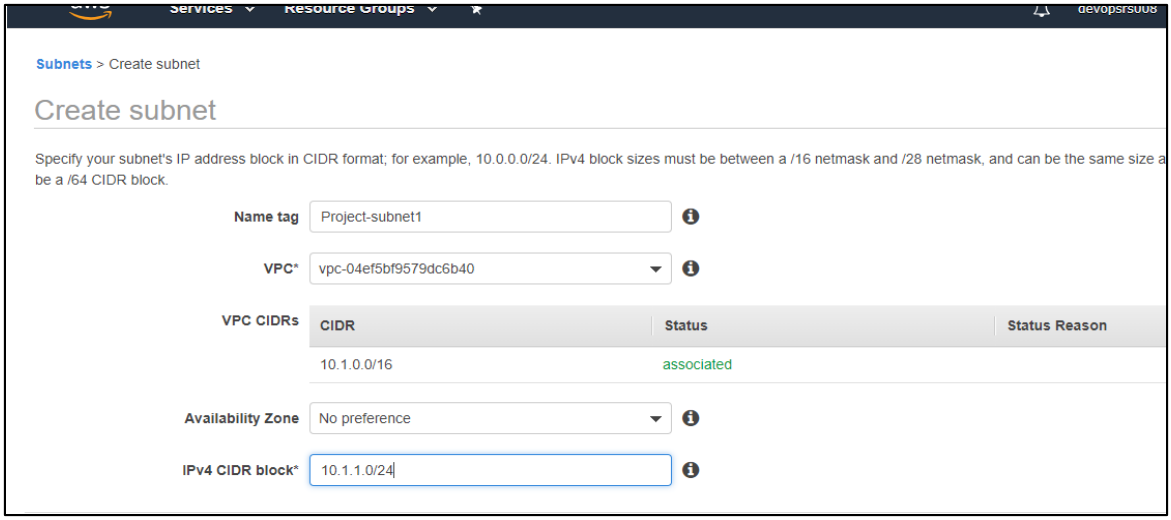


e. Click **Save** button:

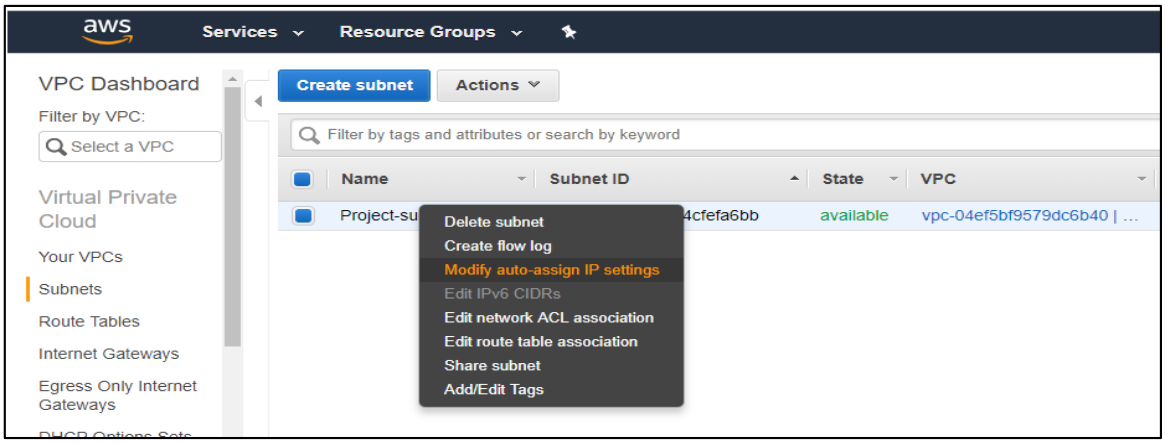
2. **Subnet creation**:
   a. Click the **subnets** menu.

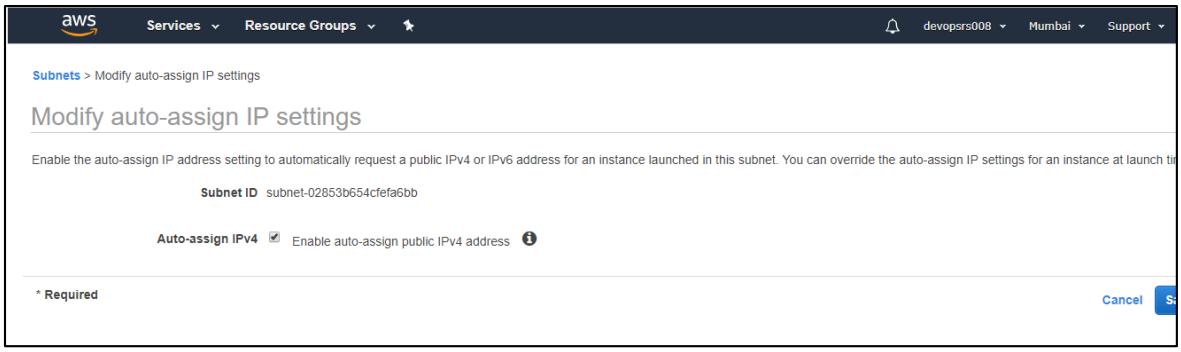


Once you created VPC, you can create subnet which means you have to define your servers within the availability zones.

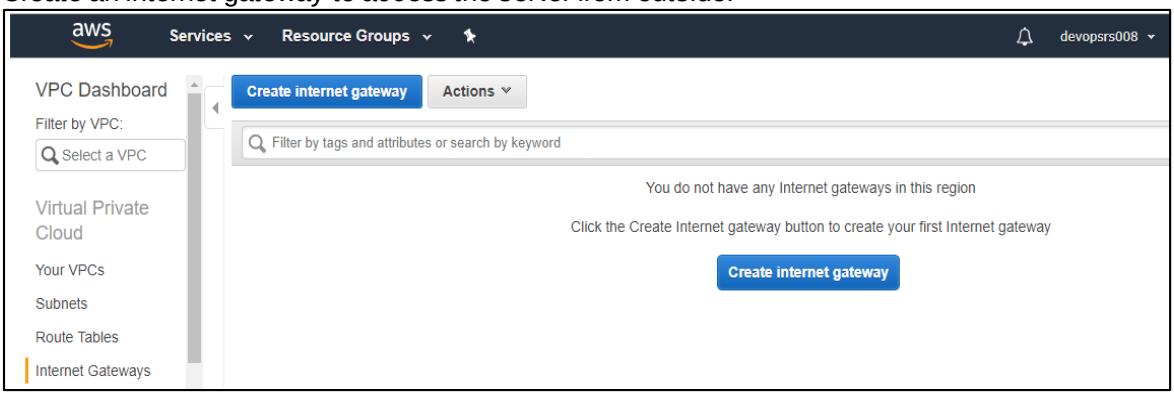


   b. Assign the auto IP address to subnet:

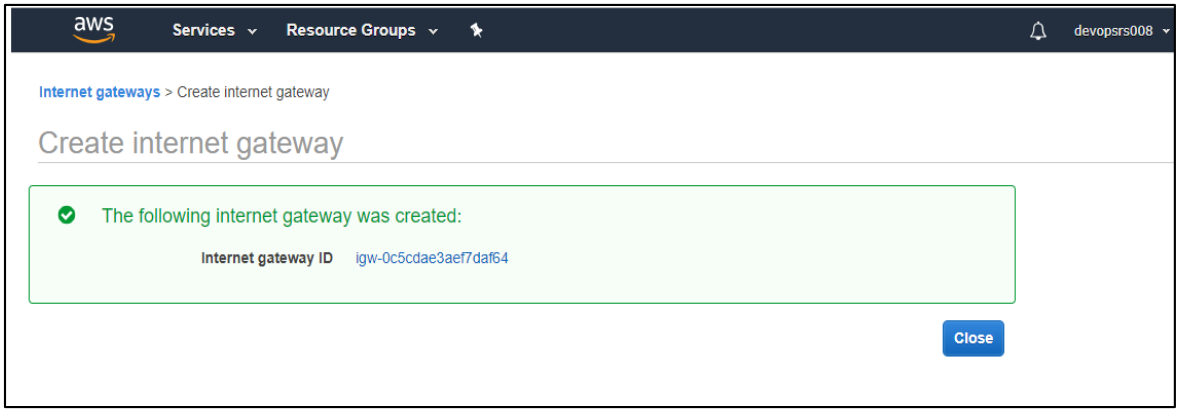




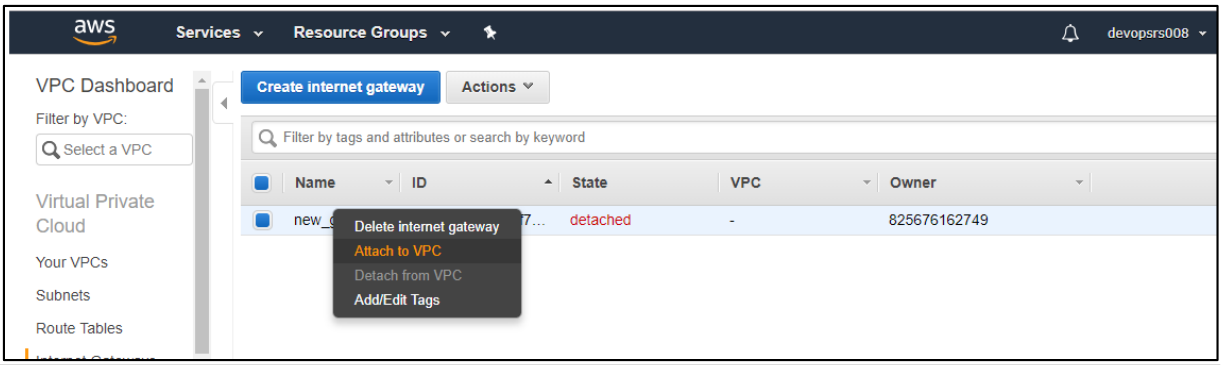
c. Create an internet gateway to access the server from outside:

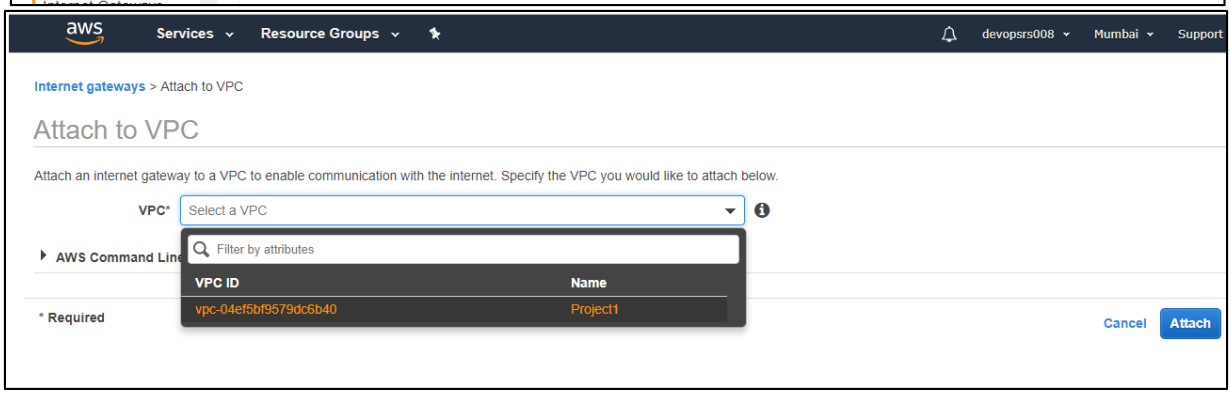


d. Click the **internet gateway** button and provide the name:

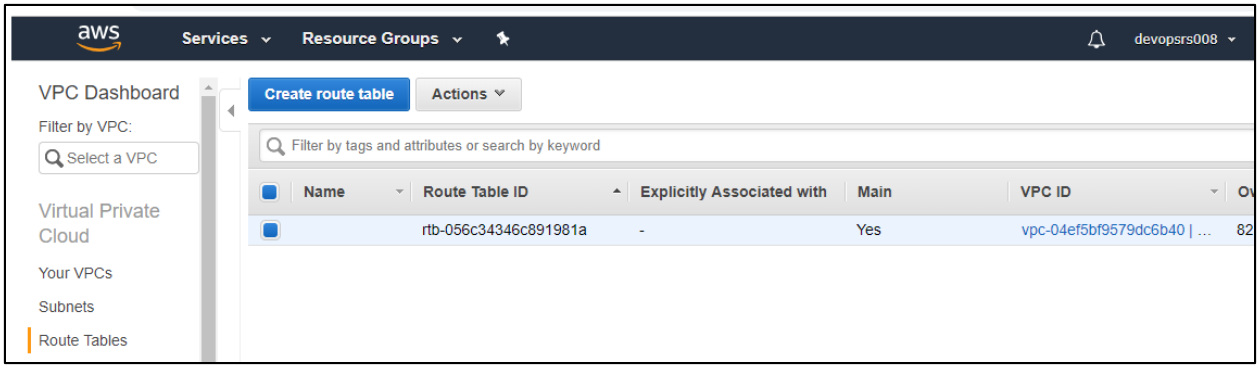


e. Once the Internet gateway is created, attach it in VPC.

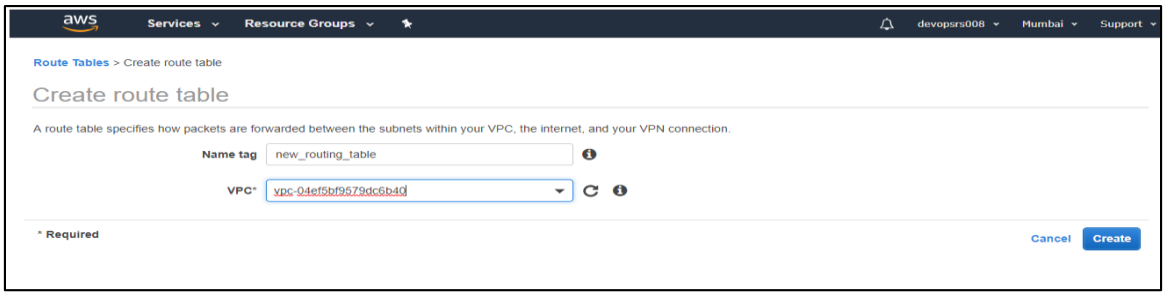


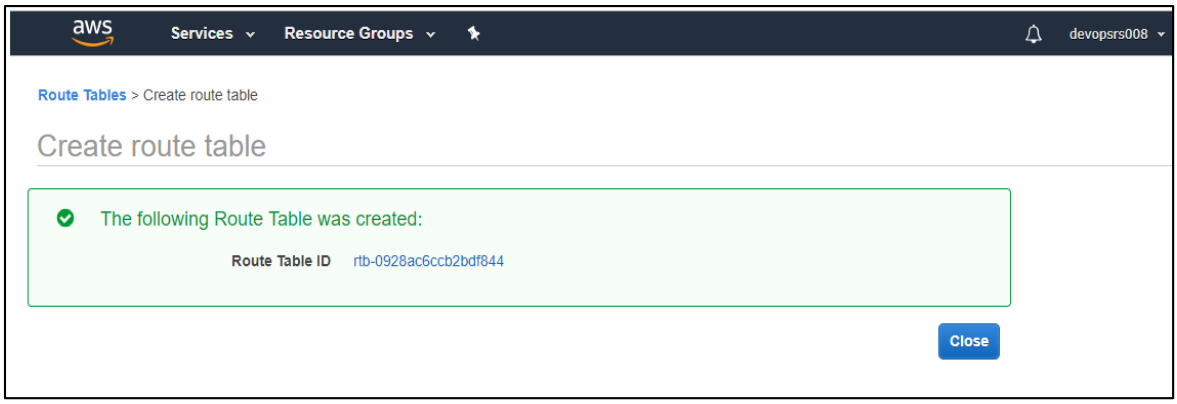


3. Route Table creation:

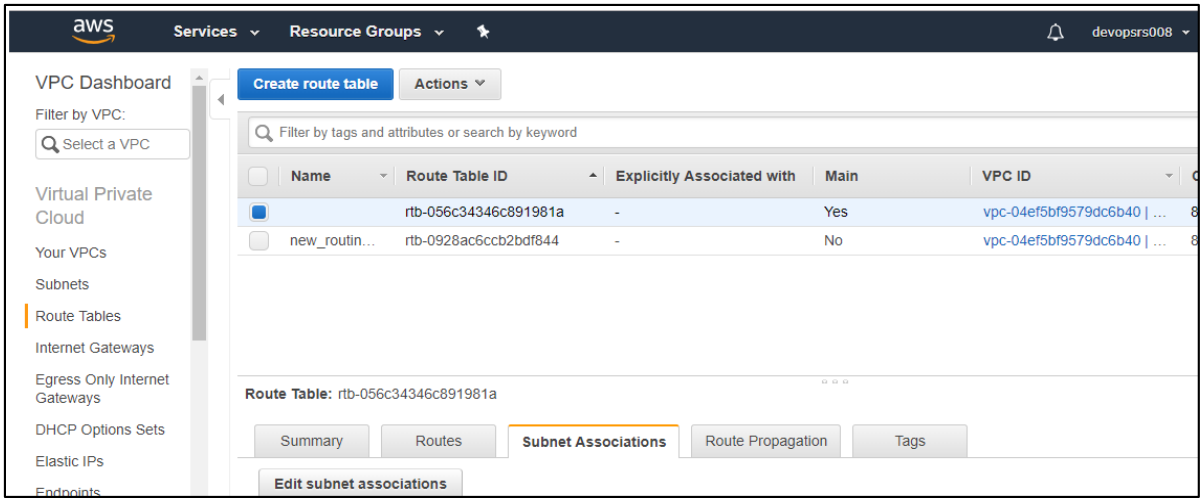


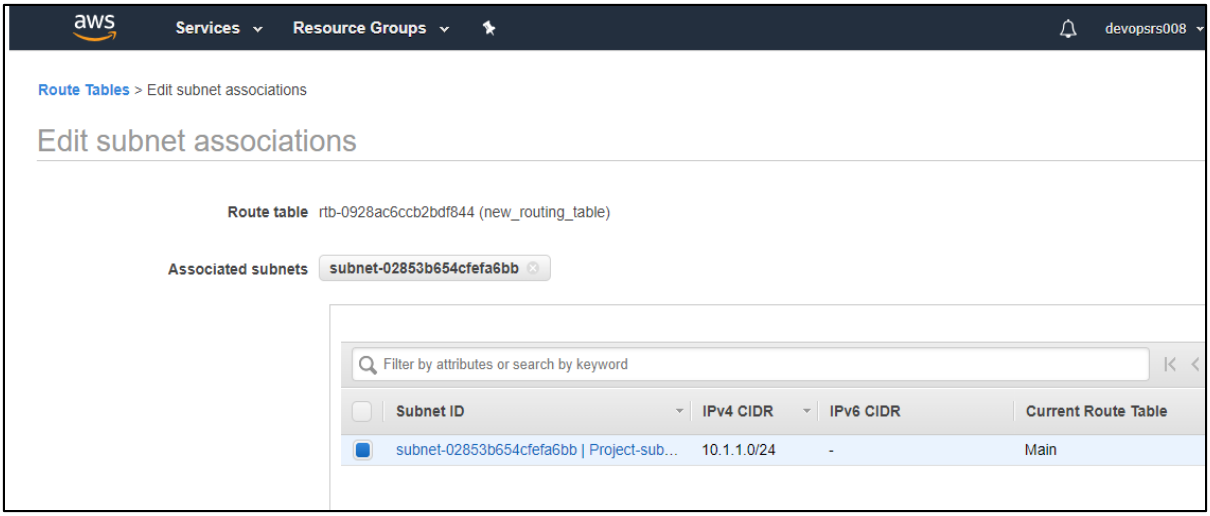
a. Click the **Create route table** button:



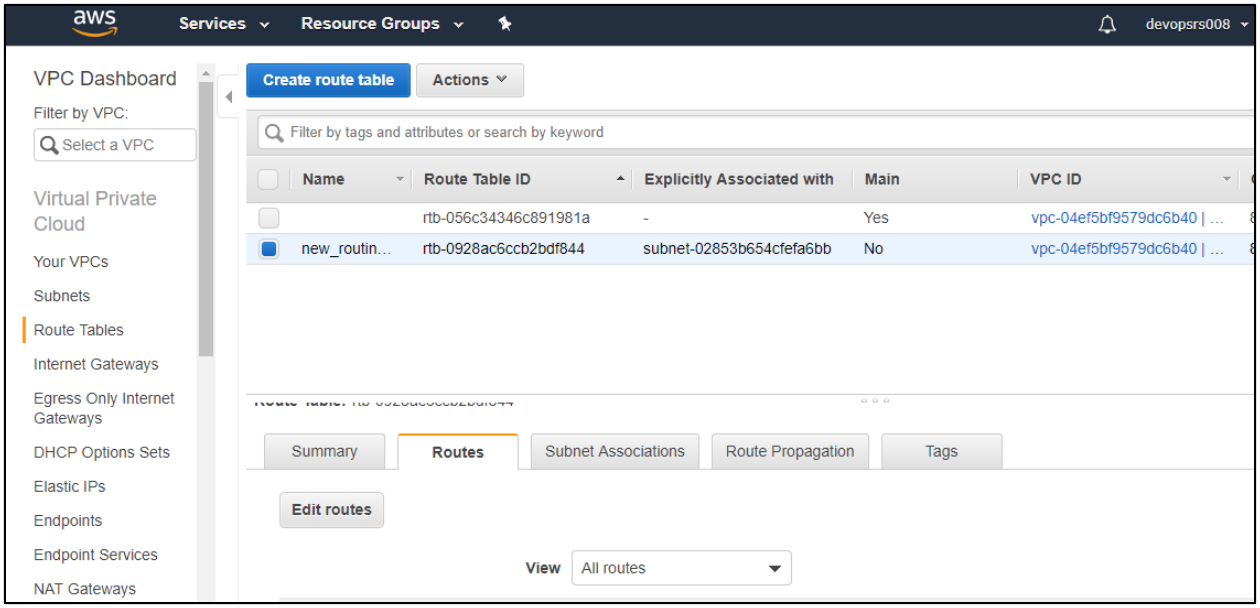


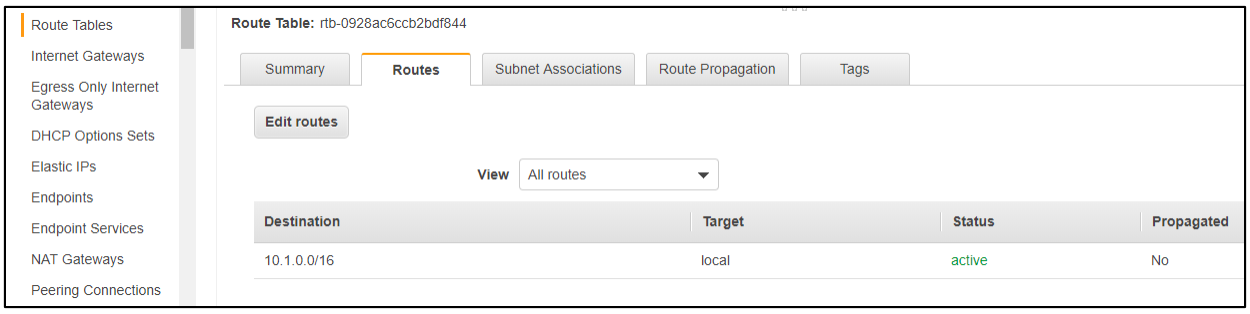
4. Subnet association:
   a. Select the routing table and select the subnet associations tab.





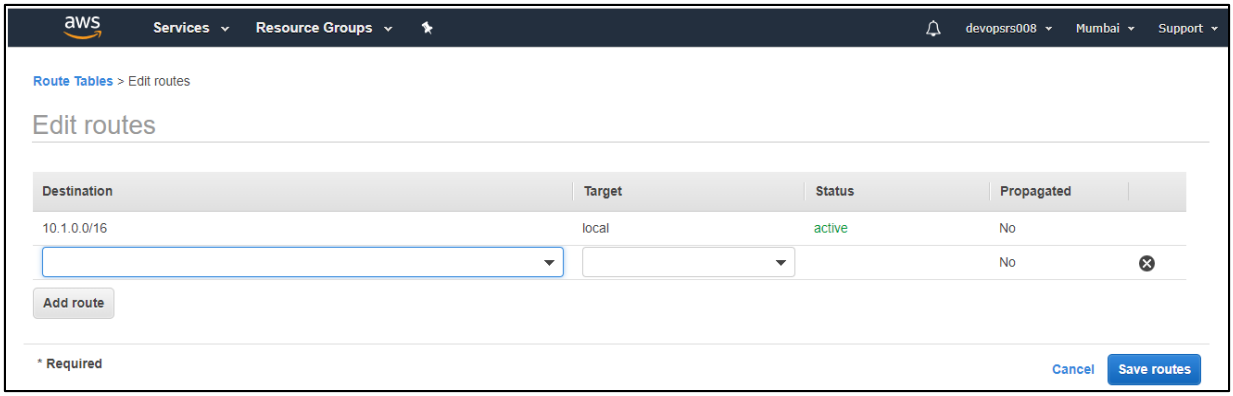
b. Check the routes tab:

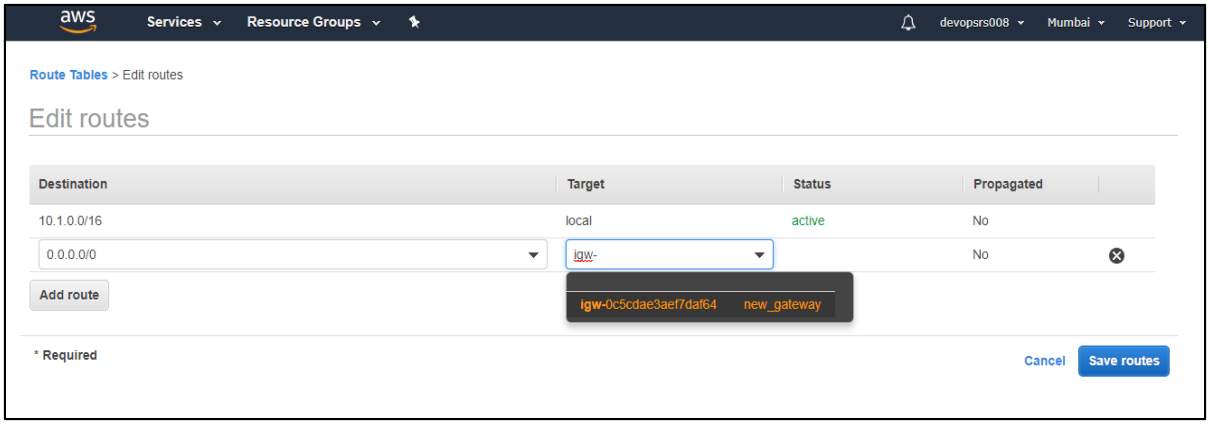


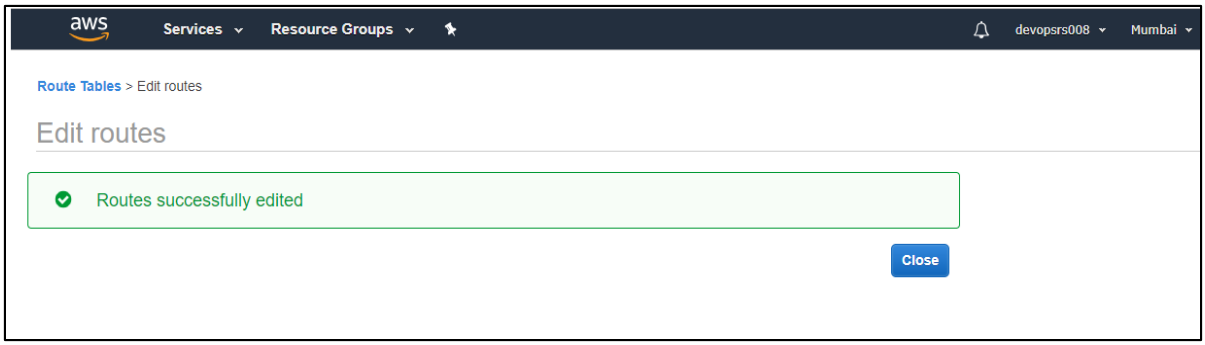


We have to redirect the traffic to different routing procedure: Within subnets we will use local routing table. For instance, if we want to browse something in the VM then it should not go and hit the local routing table so we are creating a new routing to hit the internet gateway.

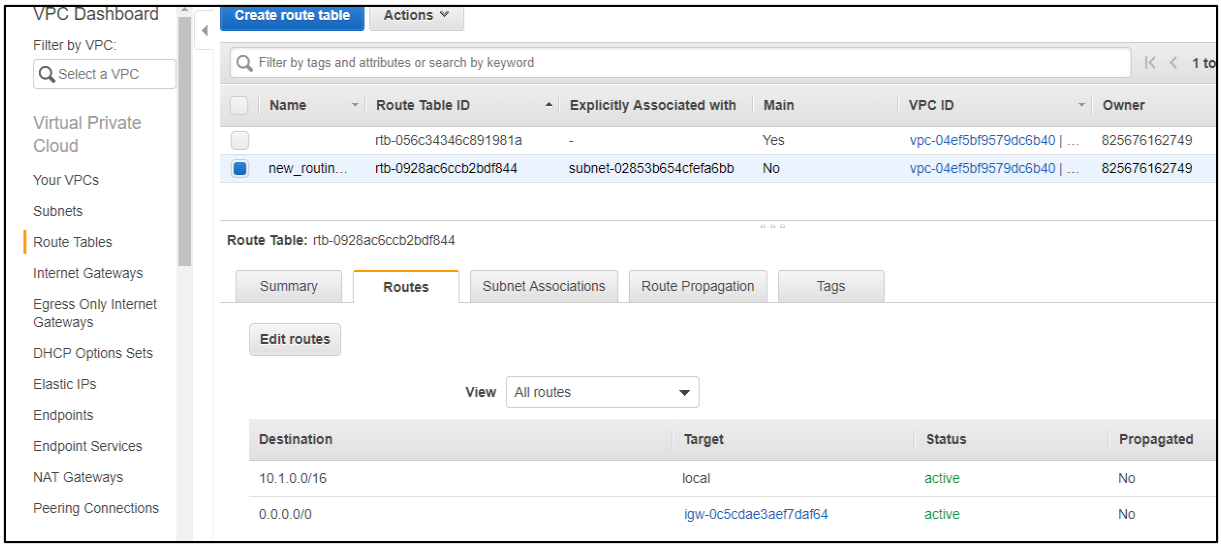
c. Click the edit routes button:



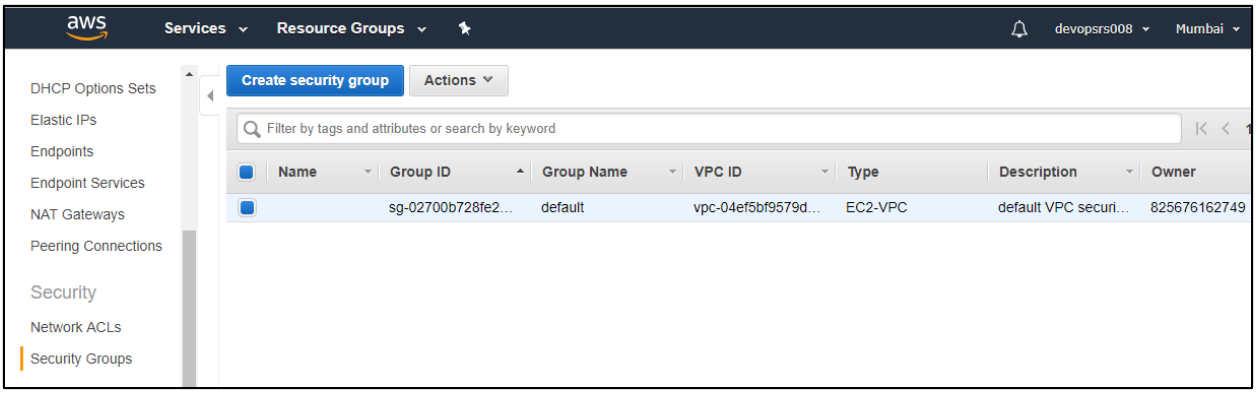


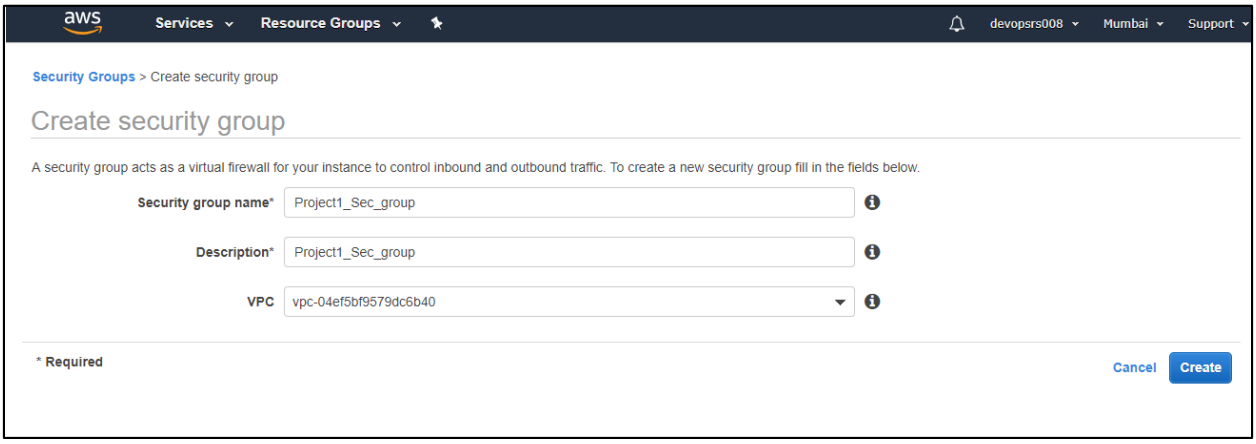


New routing is created and find the below screen.

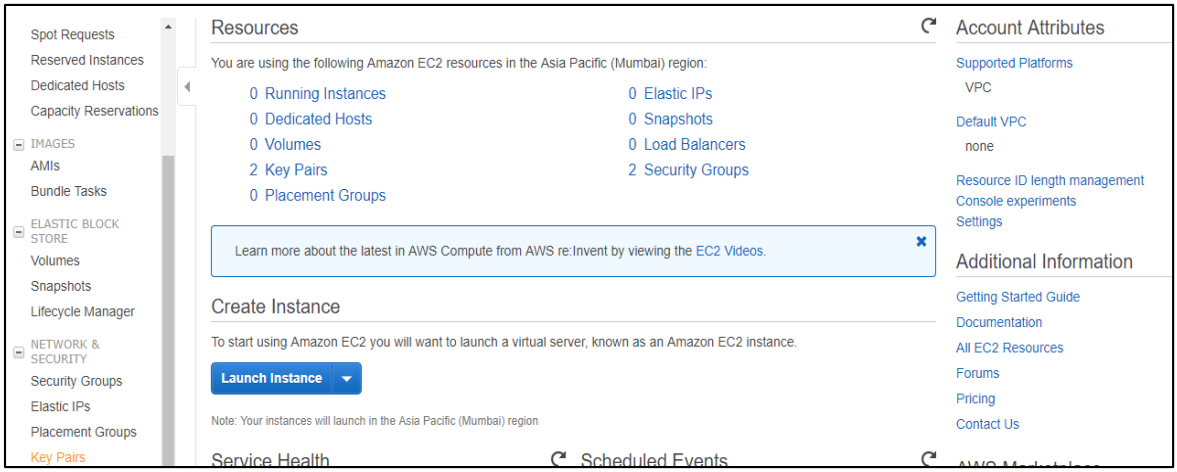


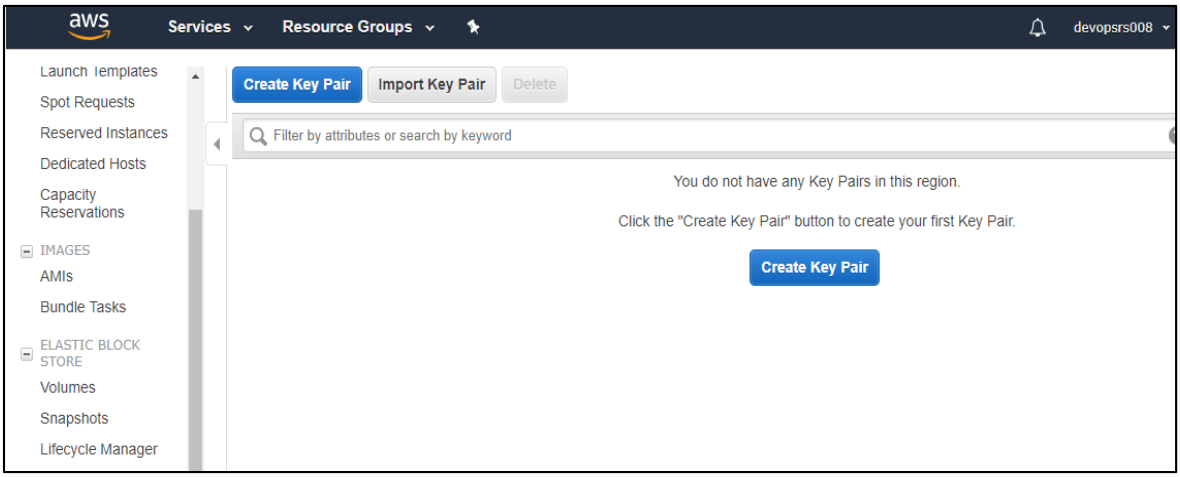
5. Create a security group:

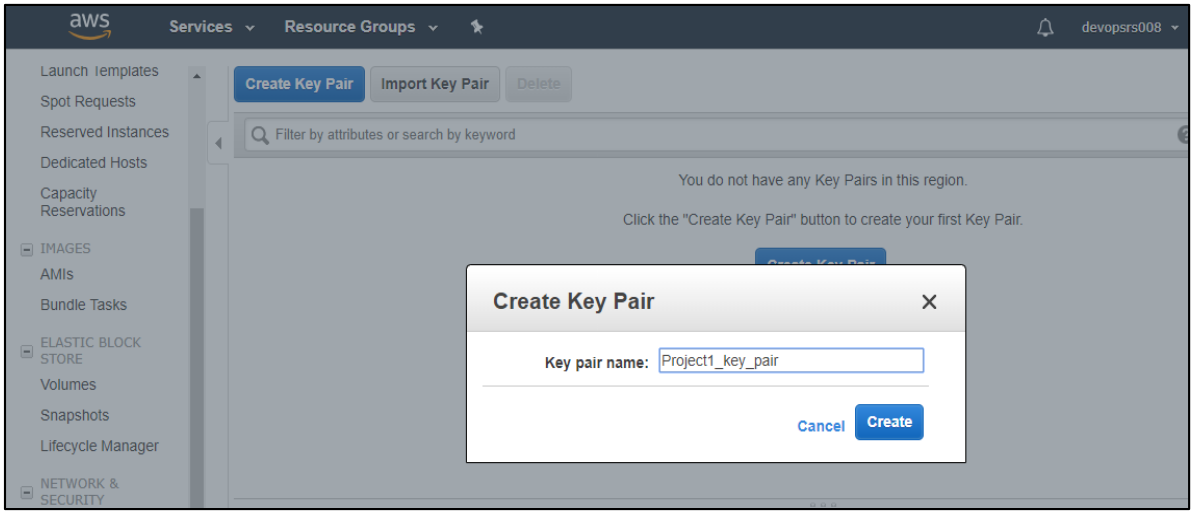




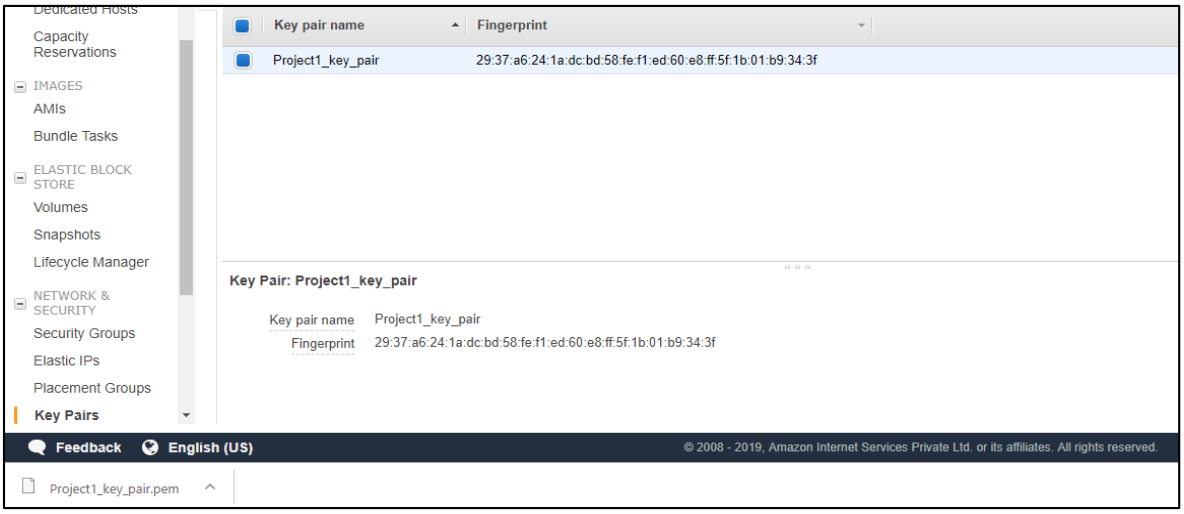
6. Create key-pairs:
   a. Select the key Pairs under Network security group.





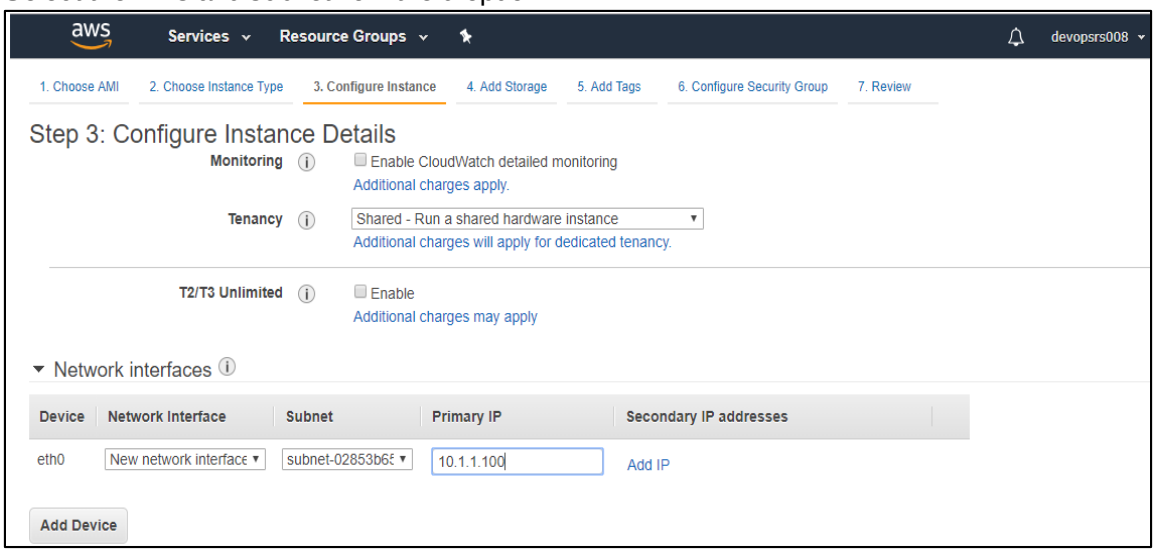


Note: PEM file will be downloaded in our download directory:

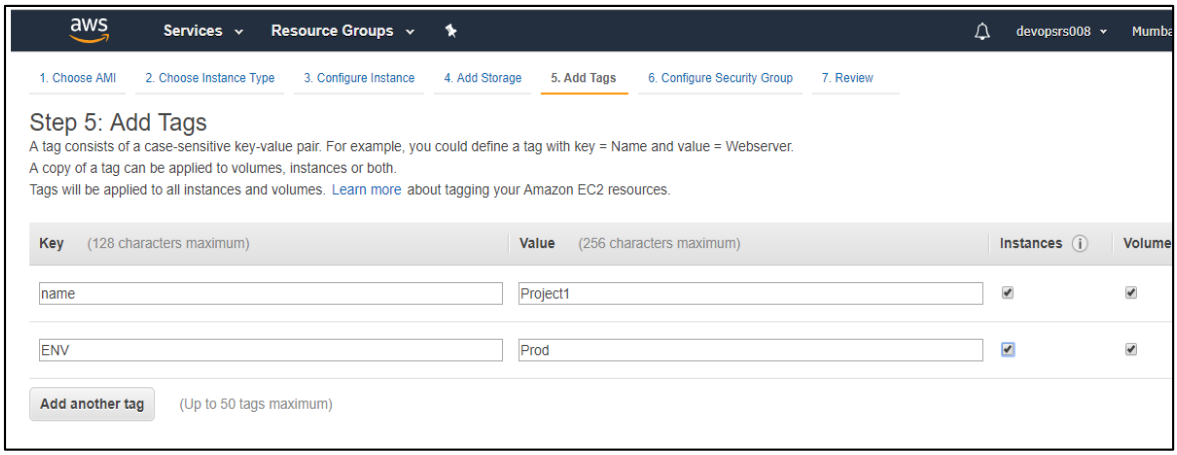


- Public and private key would be created.
- Private key comes to us.

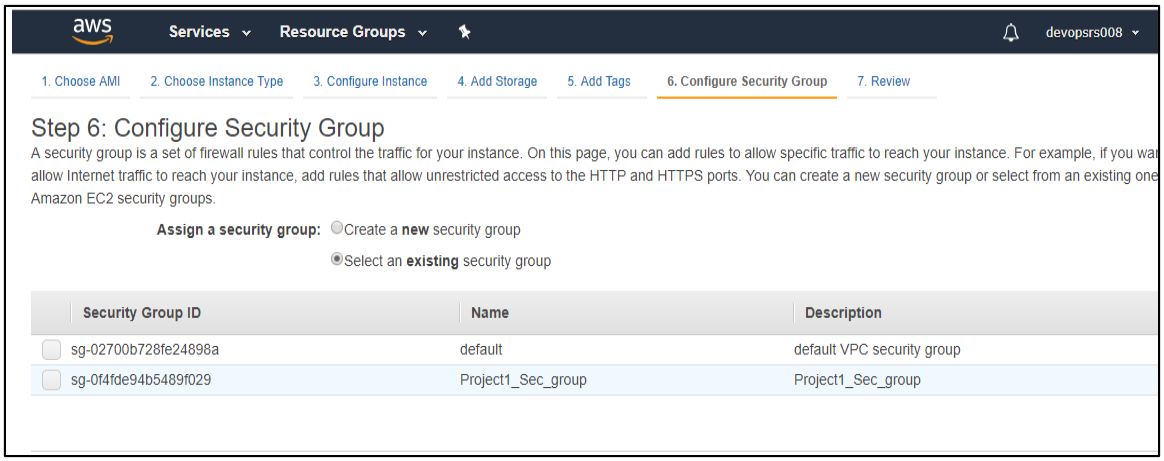
7. Create a new VM:
   a. Select the VPC and subnet from the dropdown.



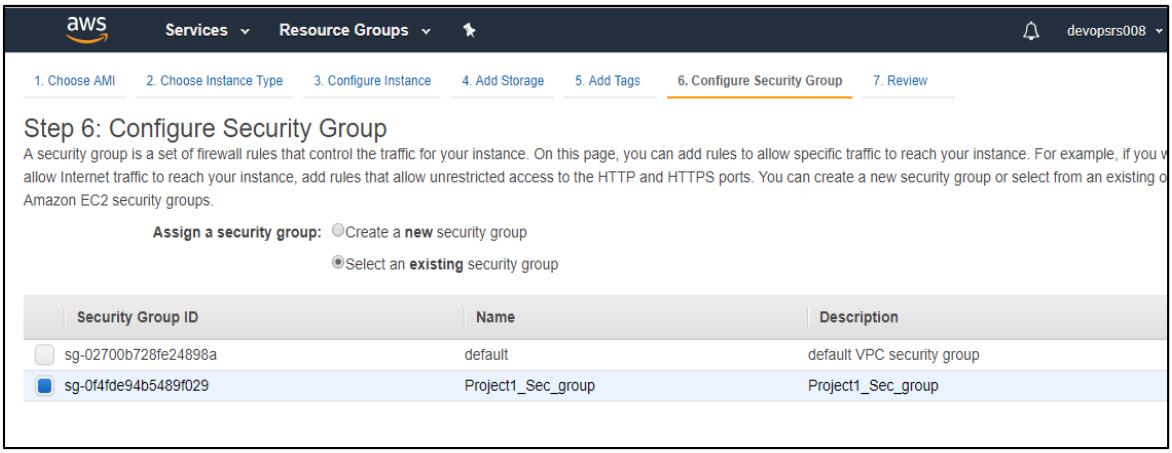
   b. Select the Tag if you want:



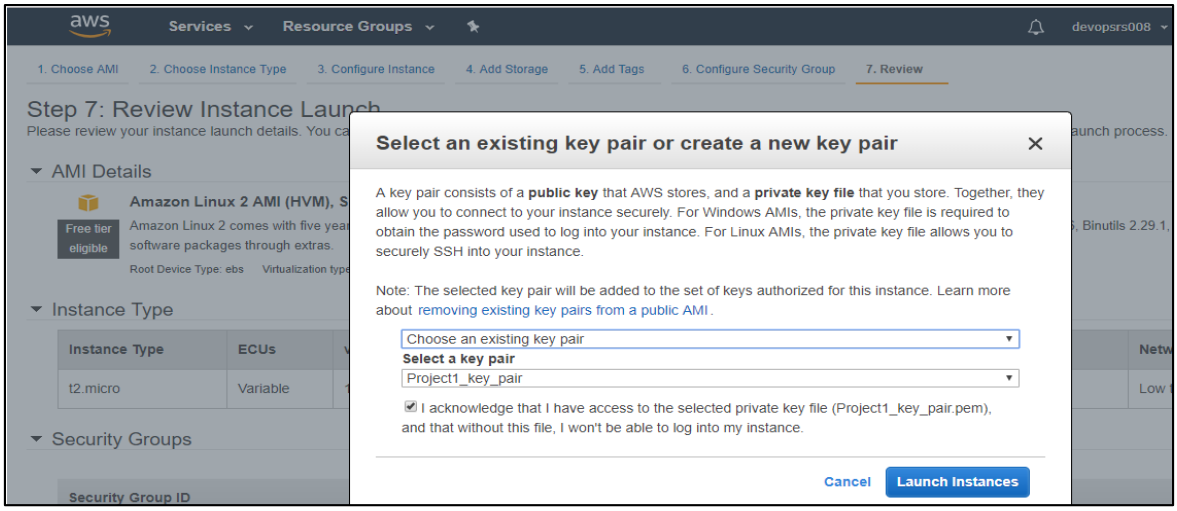
c. Select the newly created security group from the dropdown:



d. Select the security as mentioned in the below screenshot and click the review and launch.



e. Select the recently created key pair as mentioned in the below screenshot:



1.4 Troubleshooting

S. No	Problem	Solution
1	VPC is not working as expected.	Use the following command to check the VPC status. `show vpc`
2	The VPC peer ports or membership ports do not have identical configurations.	Use the following command to determine where the configuration mismatch occurs. `show vpc consistency-parameters interface`