VPC Tasks

1. Create VPC with 2 private and 2 public subnets.

To create a Virtual Private Cloud (VPC) with 2 private and 2 public subnets on AWS

* + - * AWS Management Console
      * Create VPC
        + Go to the **VPC Dashboard** in AWS.
        + Click on **Create VPC**.
        + Specify a **Name Tag** and a **CIDR block** (e.g., 198.168.0.0/24).
        + Leave **Tenancy** as default (unless you need dedicated instances).
        + Click **Create VPC**.
      * Create Subnets
        + **Public Subnet 1** and **Public Subnet 2**:

Go to **Subnets** on the VPC Dashboard.

Click **Create subnet**.

Select your VPC, name the subnet (e.g., Public-Subnet-1), and

specify an **Availability Zone** (e.g., ap-northeast-2).

Assign a **CIDR block** (e.g., 198.168.0.0/28 and 198.168.0.16/28).

Click **Create subnet**.

* + - * + Private Subnet 1 and Private Subnet 2:

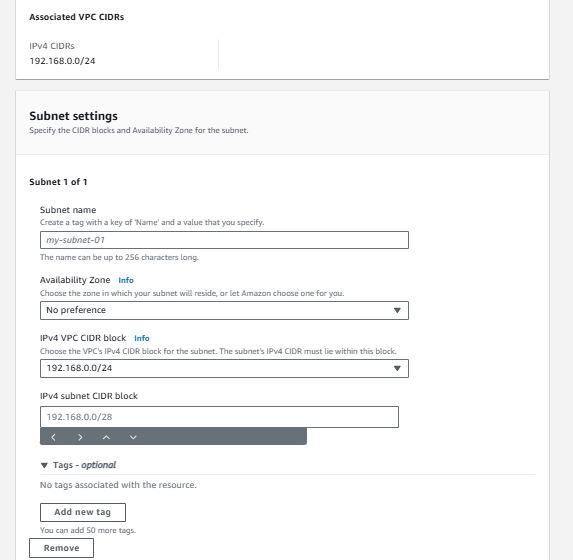
Go to **Subnets** on the VPC Dashboard.

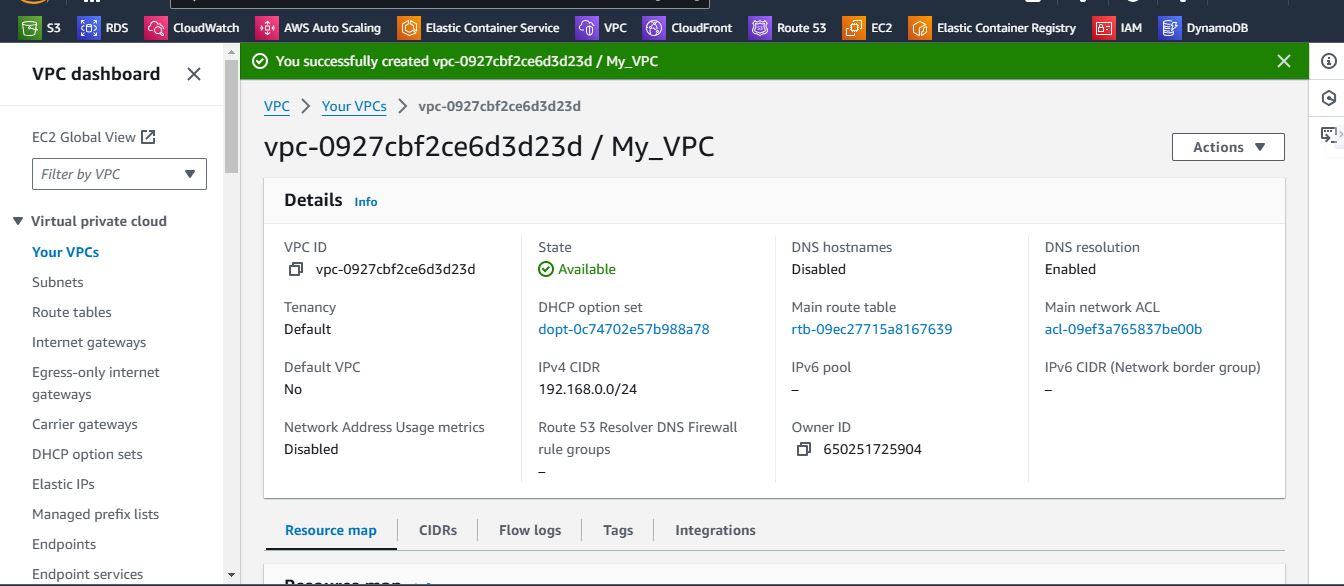
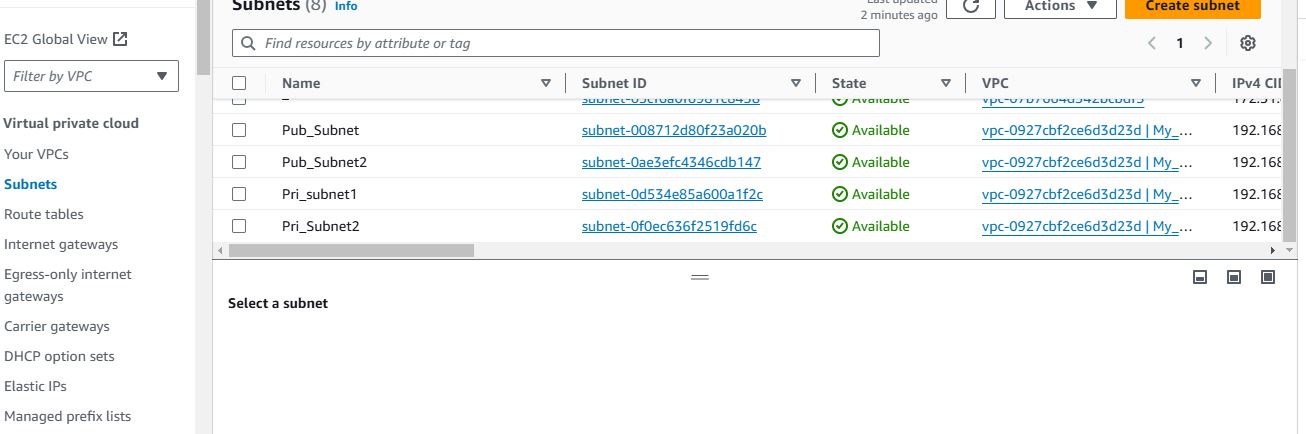
Click **Create subnet**.

Select your VPC, name the subnet (e.g., Public-Subnet-1), and

specify an **Availability Zone** (e.g., ap-northeast-2).

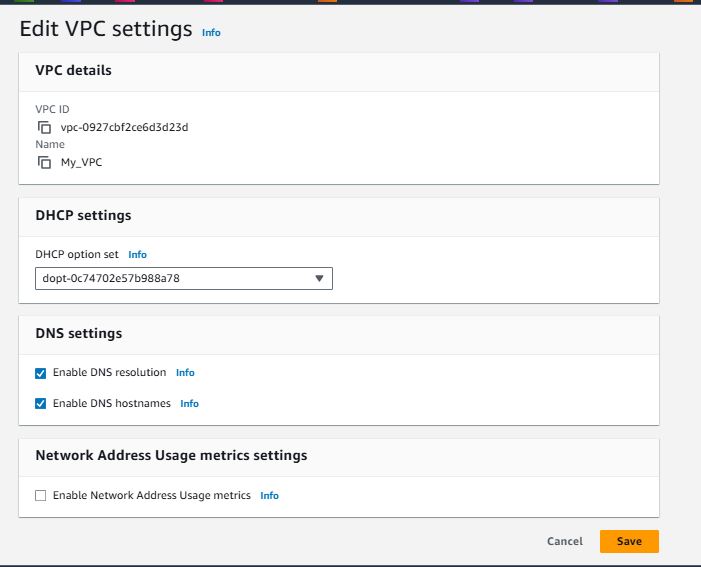
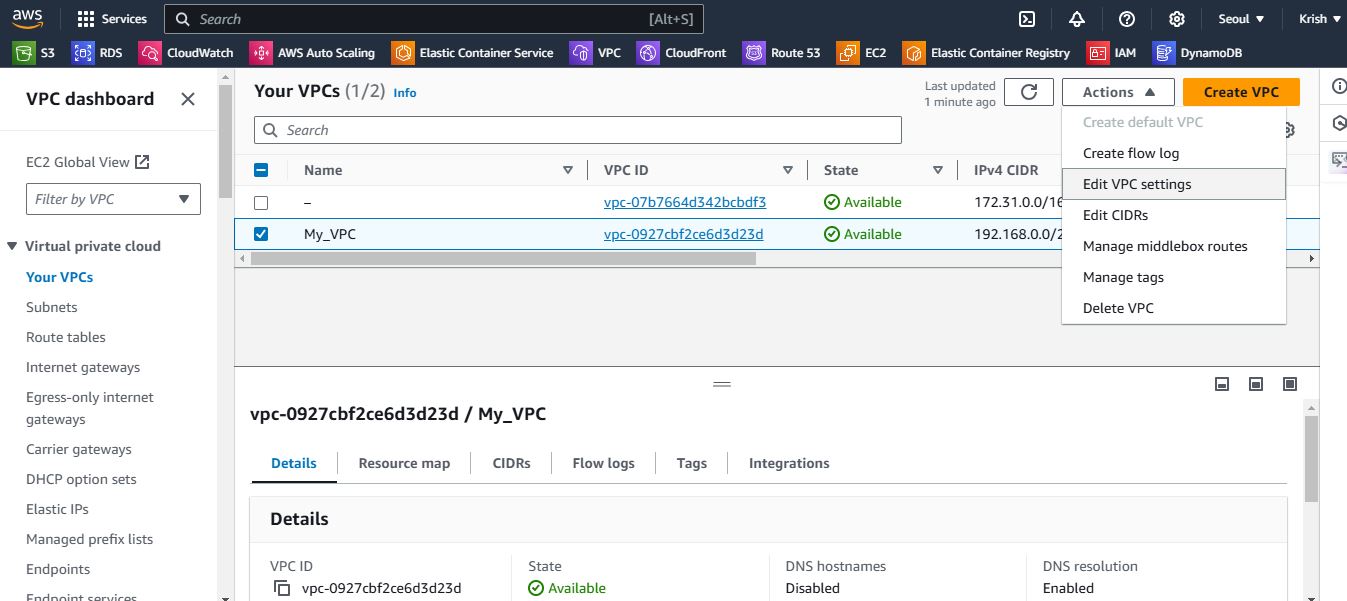
Assign a **CIDR block** (e.g., 198.168.0.32/28 and 198.168.0.64/28).

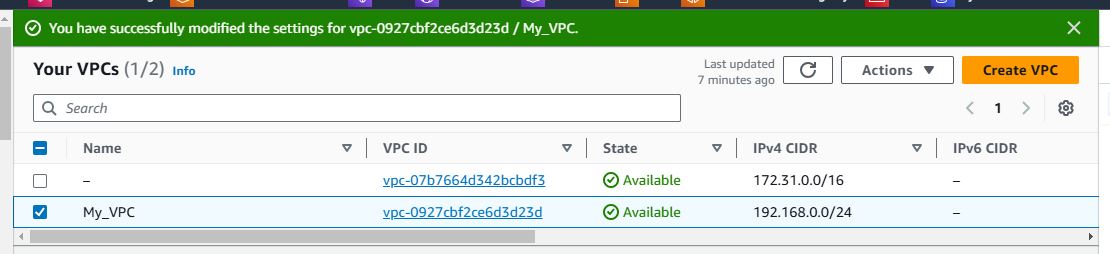
Click **Create subnet**.



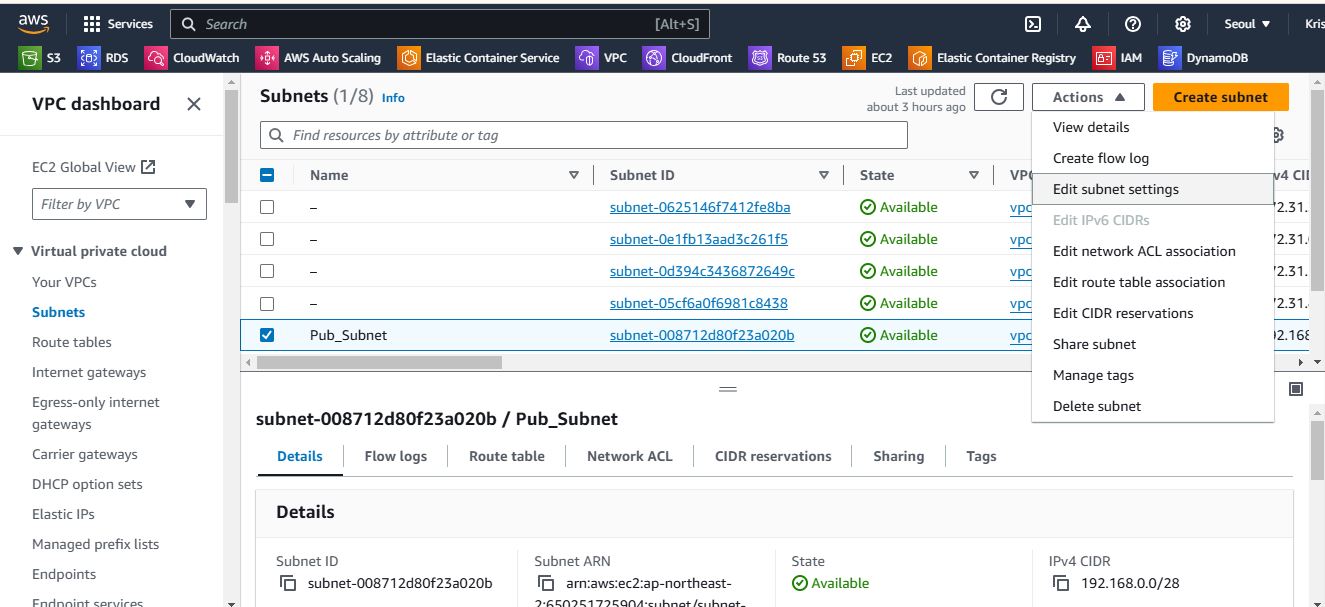
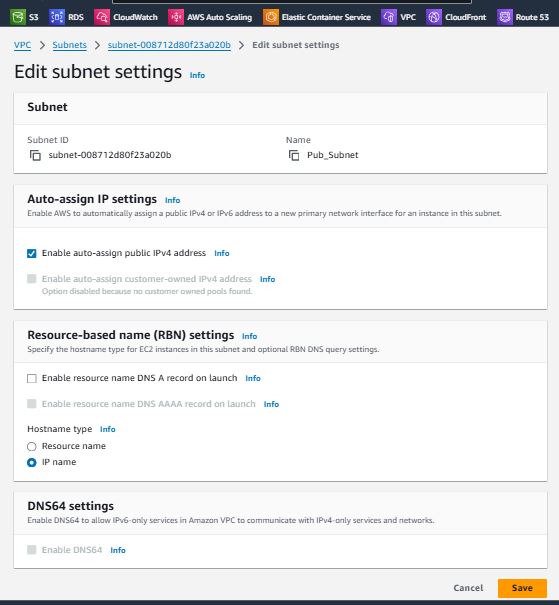
1. Enable DNS Hostname in VPC.

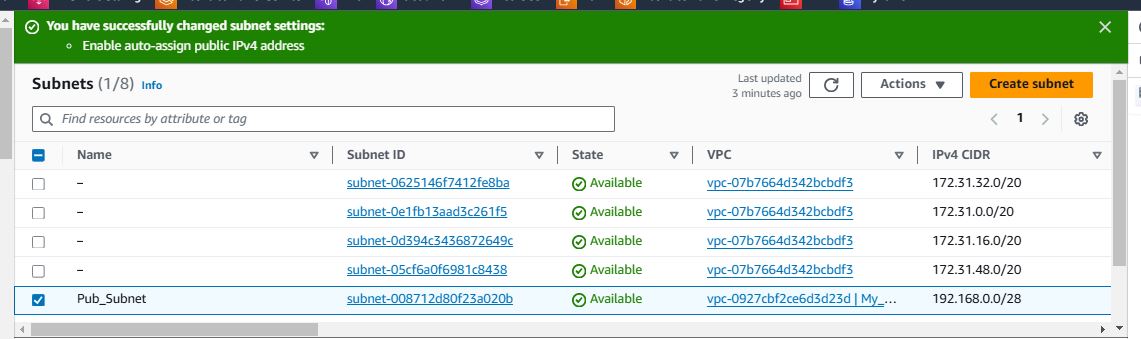
* Log in to the AWS Console and go to the VPC Dashboard.
* Select the VPC:
  + In the left-hand menu, select Your VPCs.
  + Find the VPC you want to enable DNS Hostnames for and select it.
* Modify DNS Settings:
  + With the VPC selected, click on **Actions** at the top, then choose **Edit VPC settings** or **Edit DNS Hostnames**, depending on the updated options.
* Save Changes.





1. Enable Auto Assign Public ip in 2 public subnets
   * + - Open the AWS VPC Dashboard
         * Log in to the AWS Management Console.
         * Navigate to VPC by typing "VPC" in the search bar and selecting VPC from the dropdown.
         * In the left-hand sidebar under **Subnets**, click on **Subnets** to see a list of your VPC’s subnets.
         * Click the Action dropdown button
         * Choose the Edit Subnet Setting
         * Look on Auto-assign IP settings and Enable auto-assign public IPv4 address.
       - Click **Save changes**.





1. Add 2 private subnets in private route table.
   * + - * **Open the AWS VPC Consol**

**Log in** to the **AWS Management Console**.

In the search bar, type **VPC** and select **VPC** from the dropdown.

* + - * + In the VPC Dashboard, under **Route Tables**, you can either **create a new route table**

To **create a new route table**, click **Create route table**, name it (e.g., PrivateRouteTable), and ensure it's in the correct VPC.

* + - * + Associate the Private Subnets with the Private Route Table

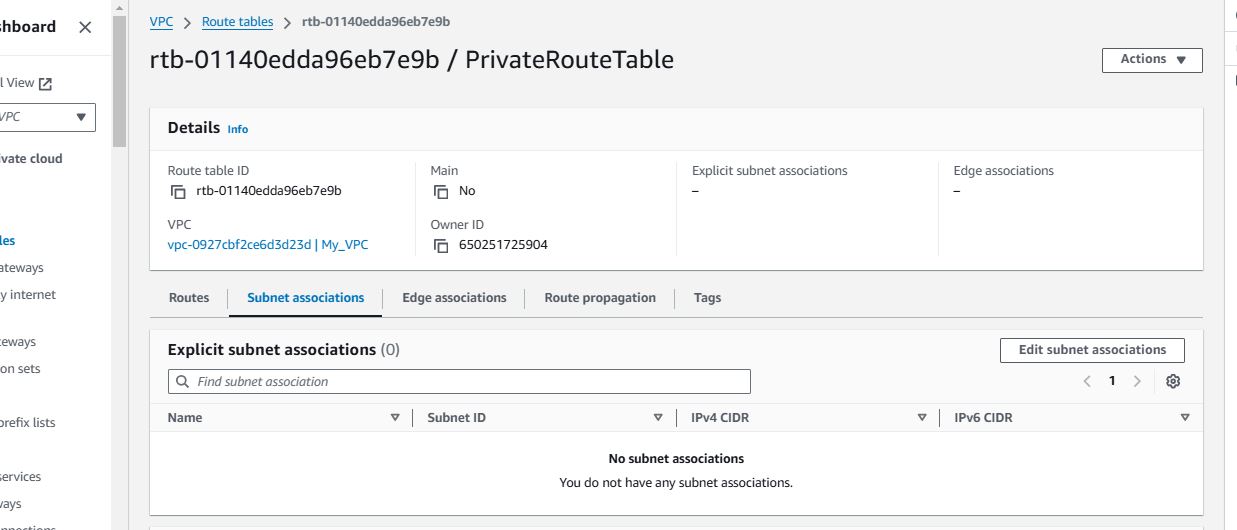
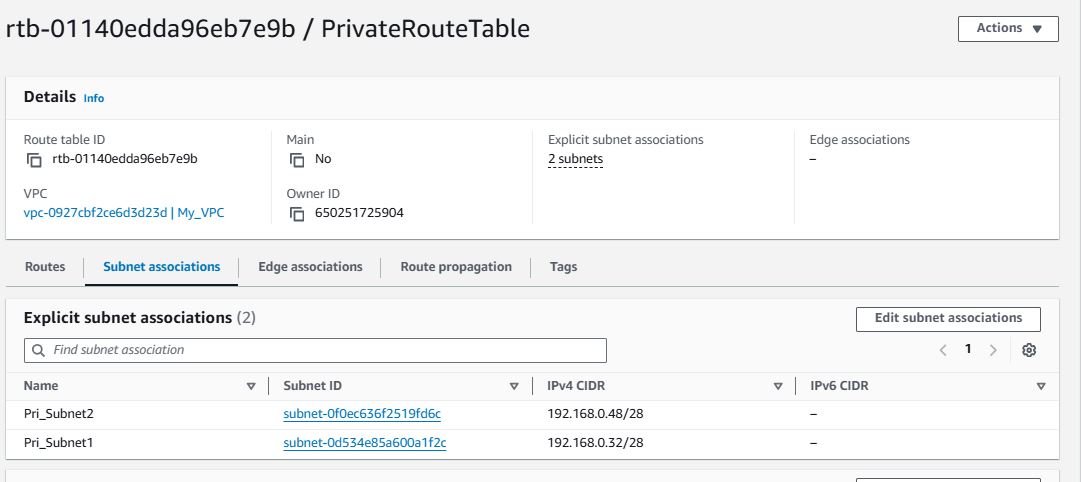
**Go to the Route Tables section** in the VPC Dashboard.

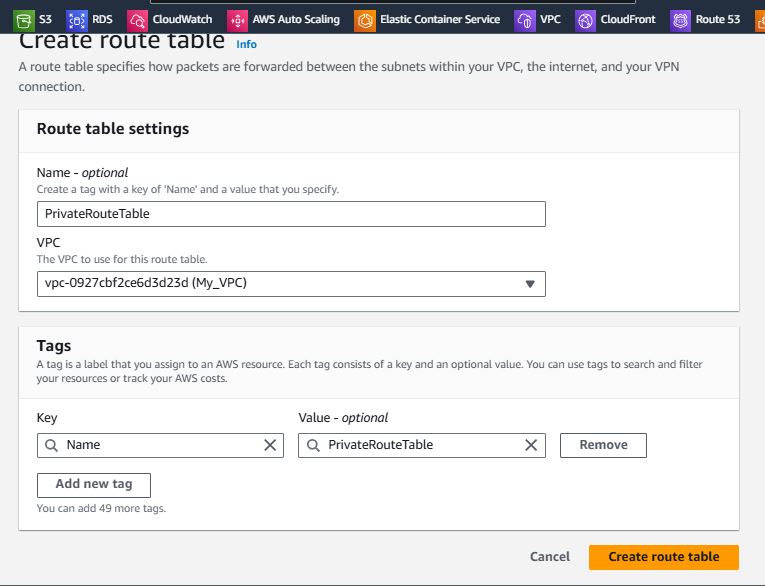
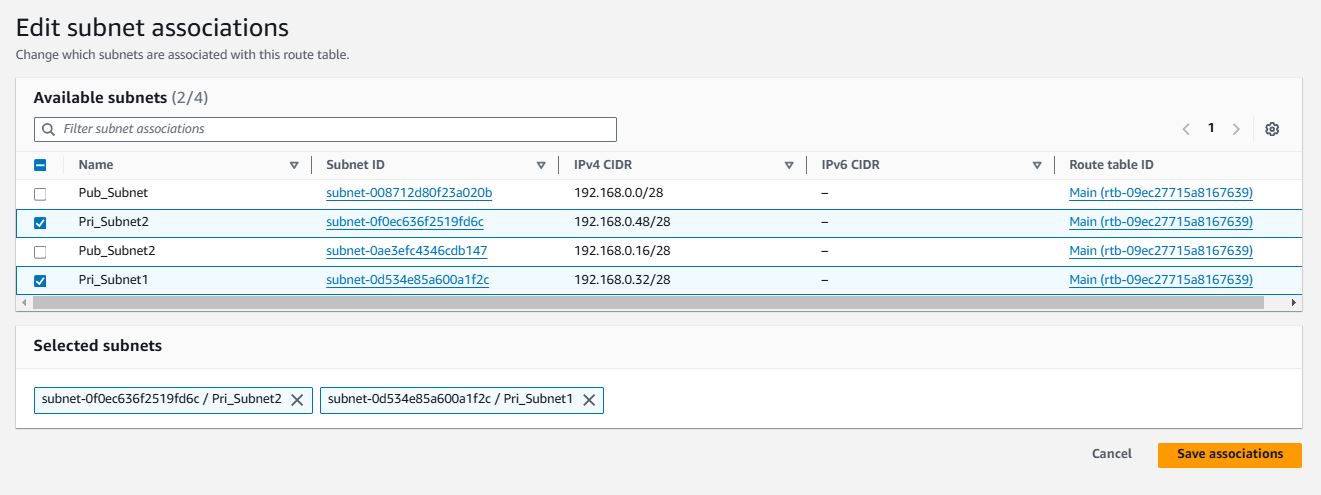
Select the **private route table** (either the new or existing one) that you want to associate with the two private subnets.

In the **Route Table Details**, select the **Subnet Associations** tab.

Click **Edit subnet associations**.

From the list of subnets, **select the two private subnets** that you want to associate with the private route table.

Click **Save associations**.



1. Add 2 public subnets in public route table

* + - * + **Open the AWS VPC Consol**

**Log in** to the **AWS Management Console**.

In the search bar, type **VPC** and select **VPC** from the dropdown.

* + - * + In the VPC Dashboard, under **Route Tables**, you can either **create a new route table**

To **create a new route table**, click **Create route table**, name it (e.g., PublicRouteTable), and ensure it's in the correct VPC.

* + - * + Associate the Private Subnets with the Private Route Table

**Go to the Route Tables section** in the VPC Dashboard.

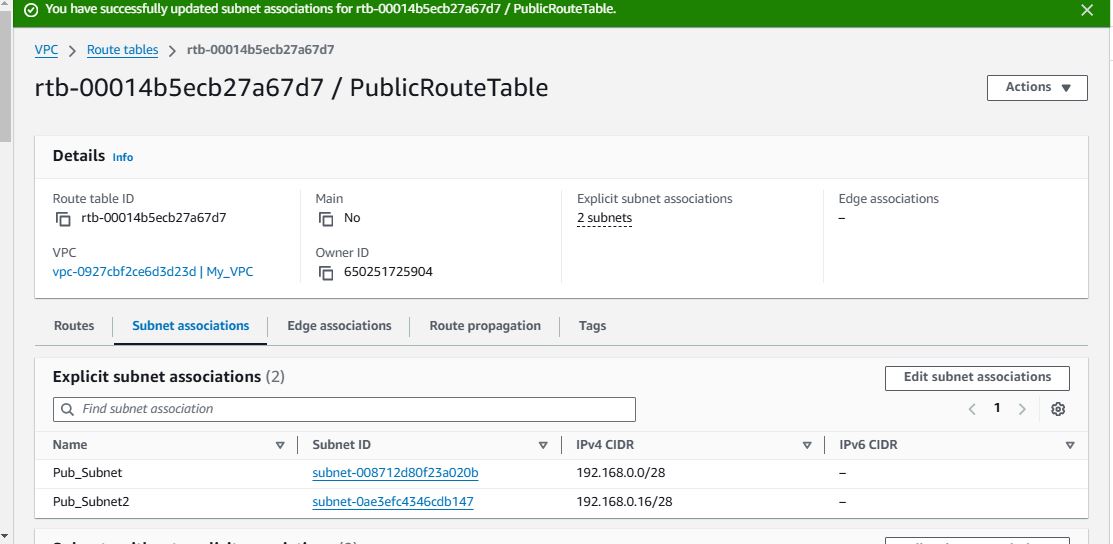
Select the **private route table** (either the new or existing one) that you want to associate with the two private subnets.

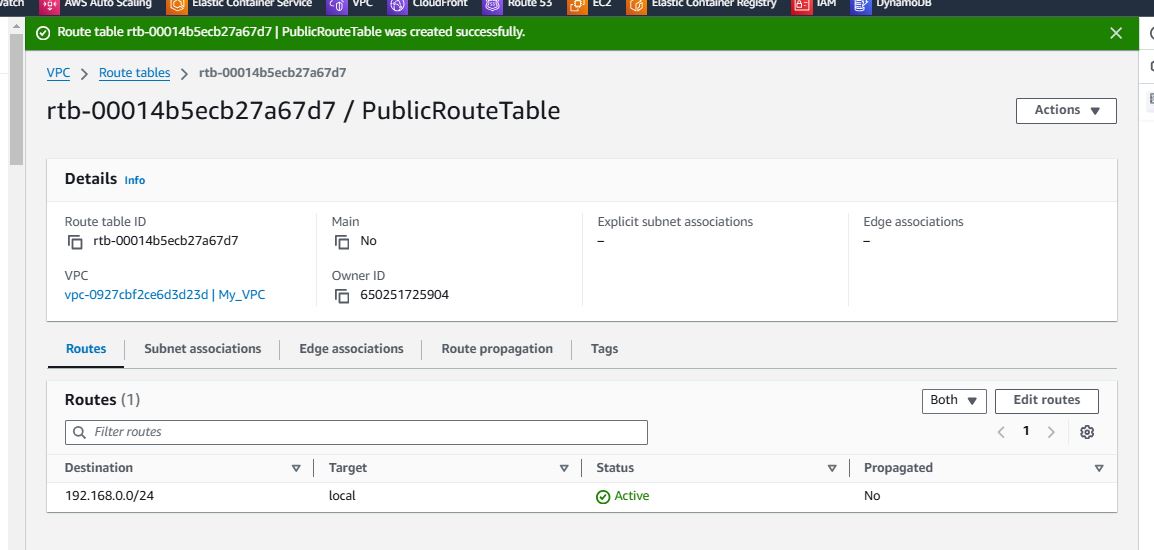
In the **Route Table Details**, select the **Subnet Associations** tab.

Click **Edit subnet associations**.

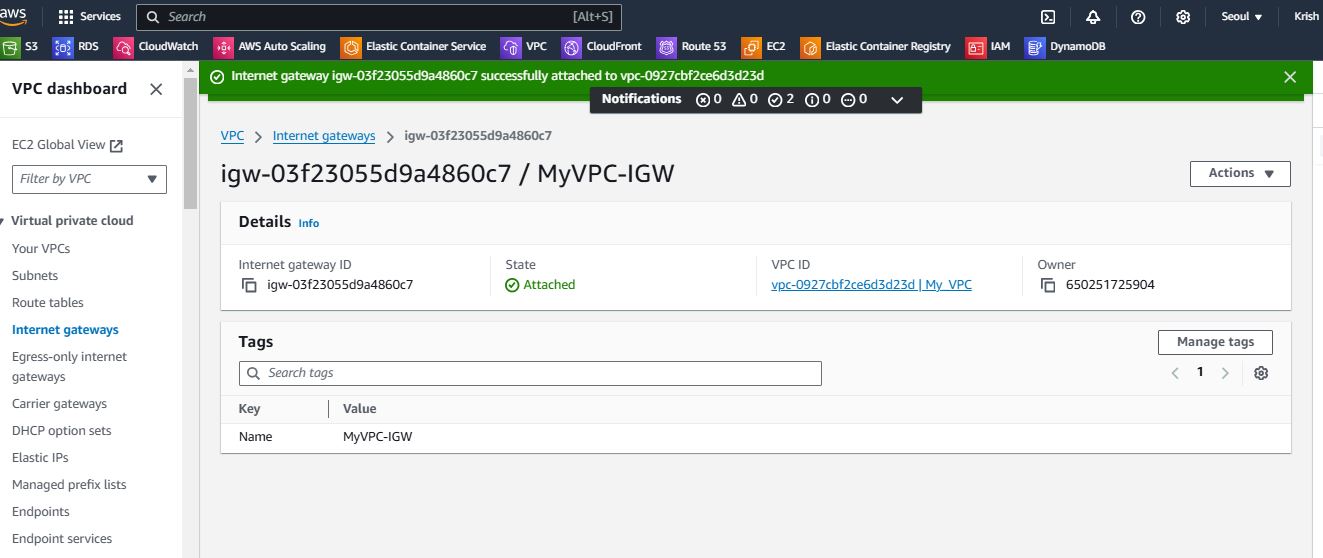
From the list of subnets, **select the two private subnets** that you want to associate with the private route table.

Click **Save associations**.

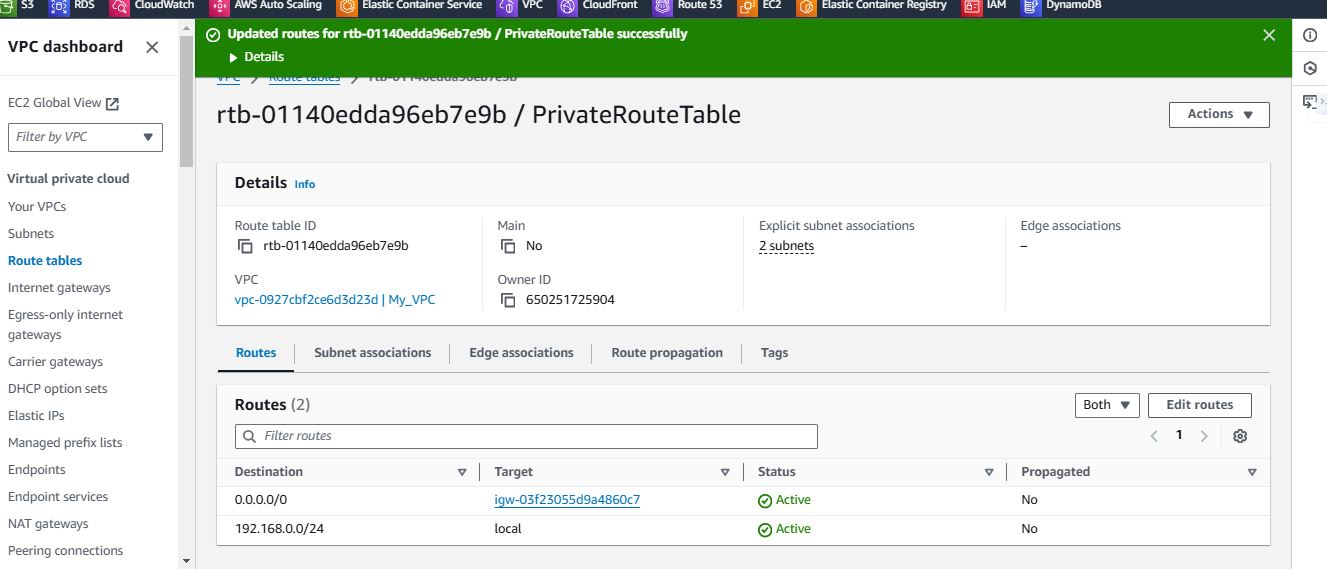


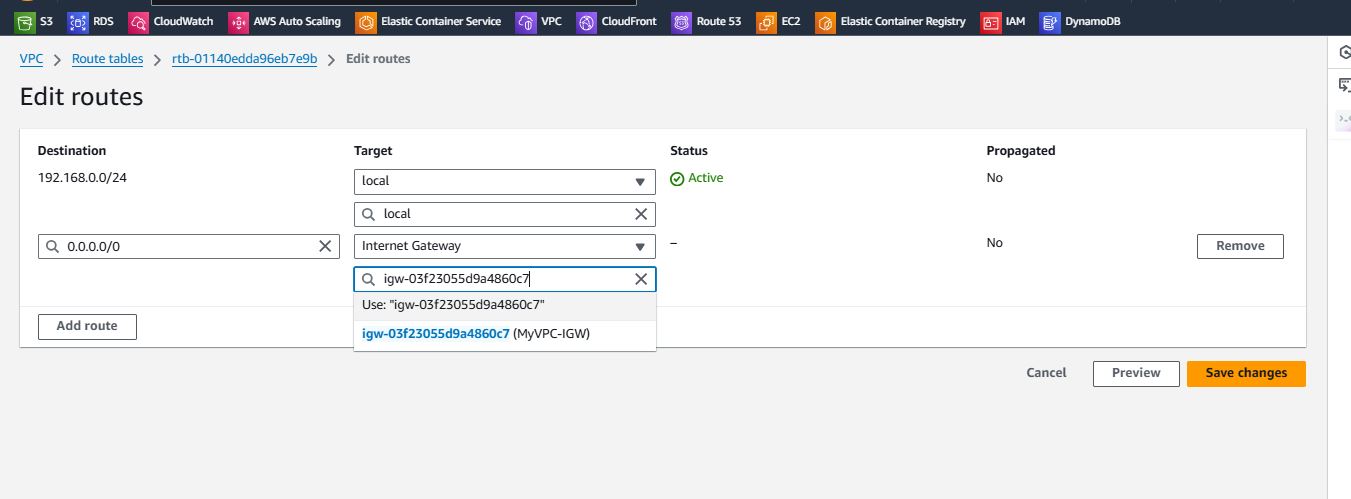


1. Public route table will have the routes to internet and local
   * + - **Local Route (for VPC Internal Traffic)**:
         * This route allows instances within the same VPC to communicate with each other.
         * AWS automatically creates this local route when the VPC is created.
       - **Internet Route (for Public Access)**:
         * This route directs outbound traffic to the internet, allowing instances in the public subnets to communicate with external networks.
         * This route requires an **Internet Gateway (IGW)** attached to the VPC.

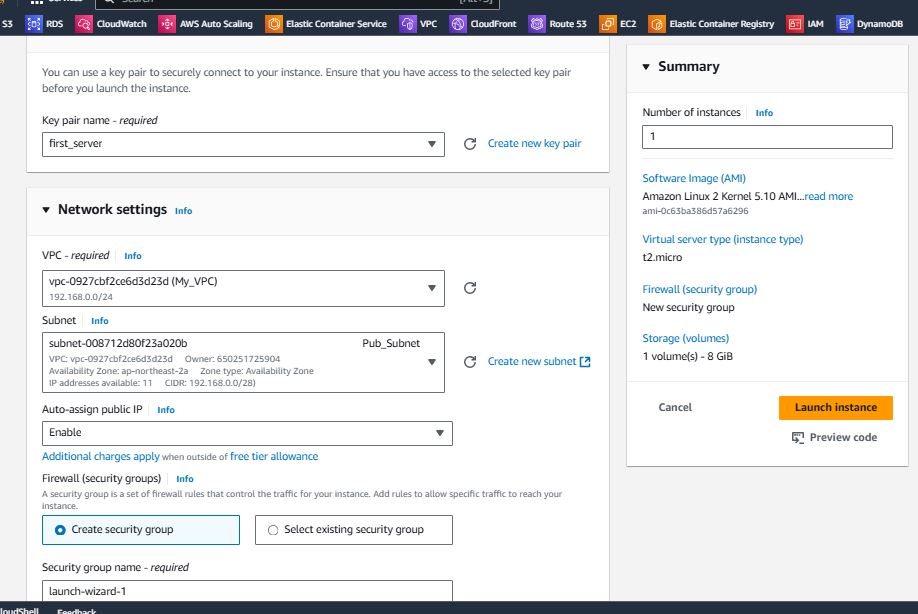
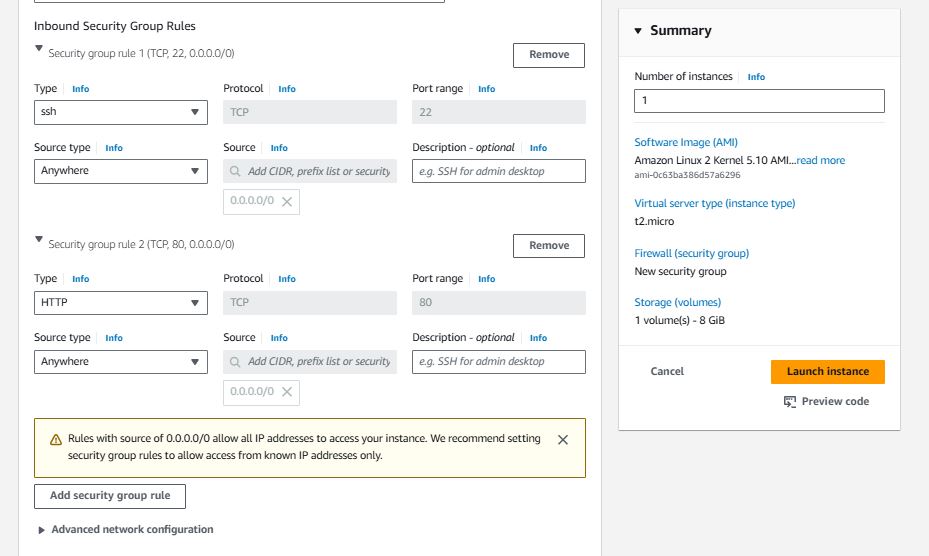


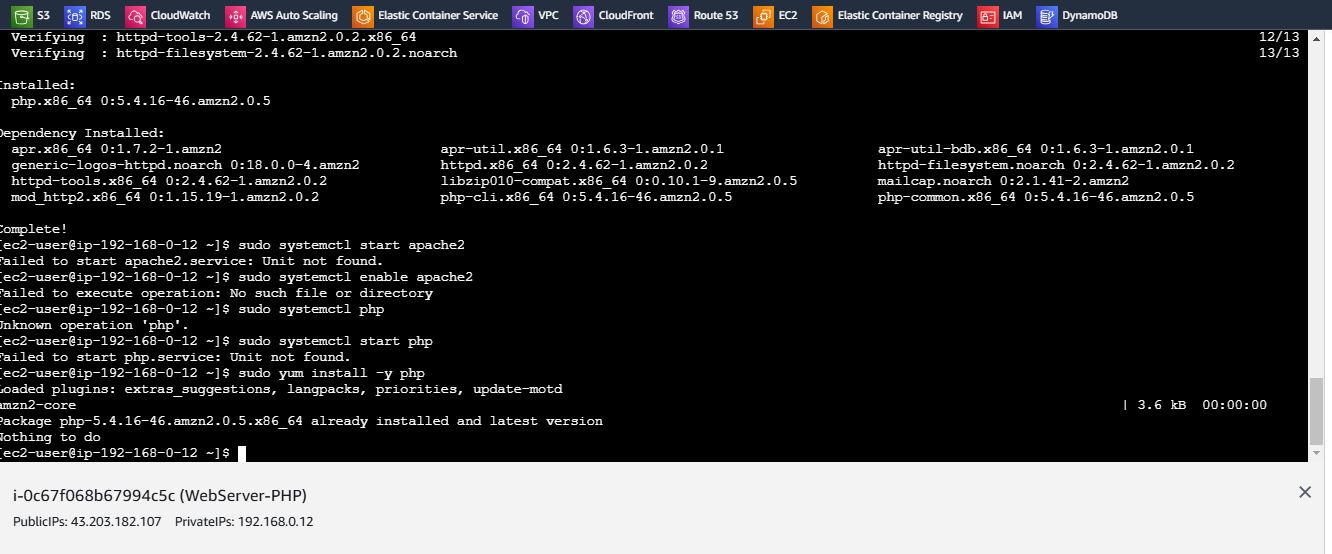
Internet Gateway



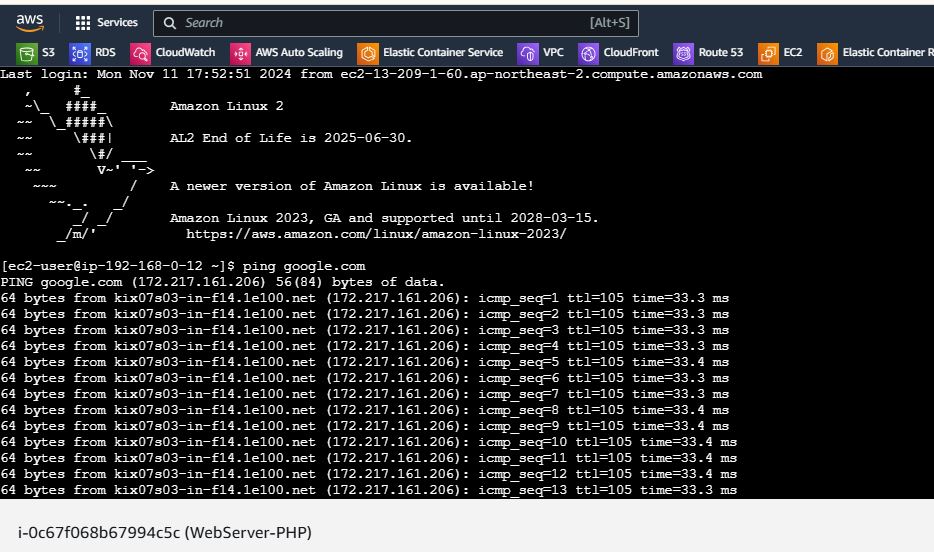


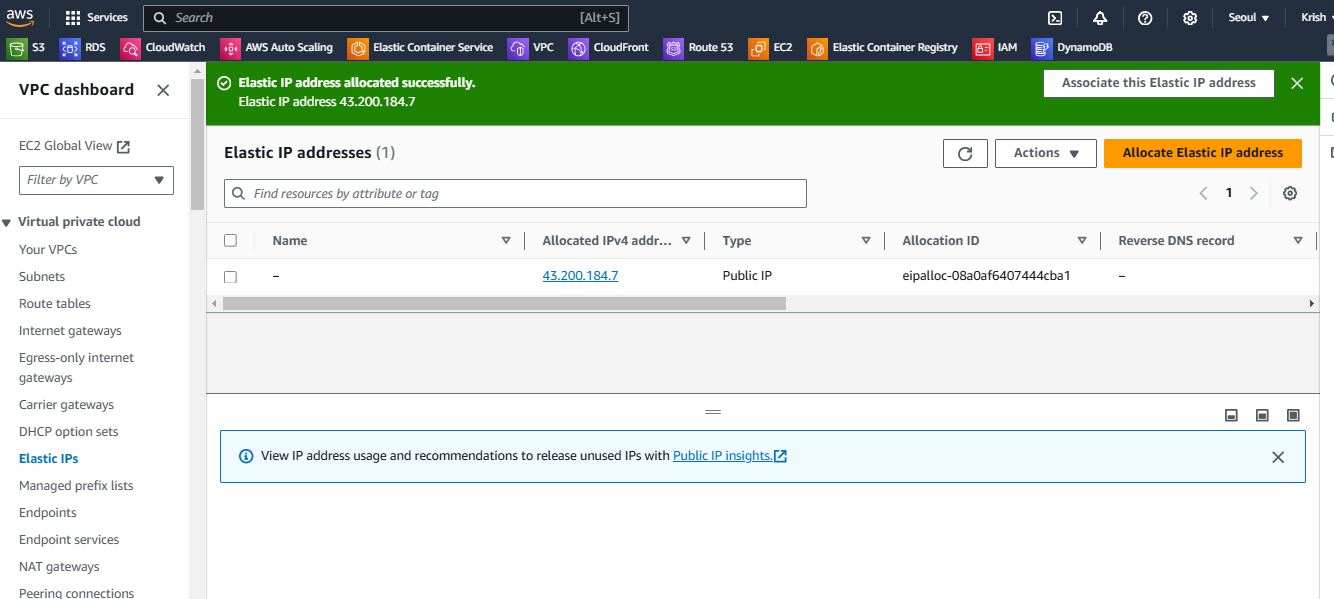
1. Create Ec2 in public subnet with t2micro and install php.

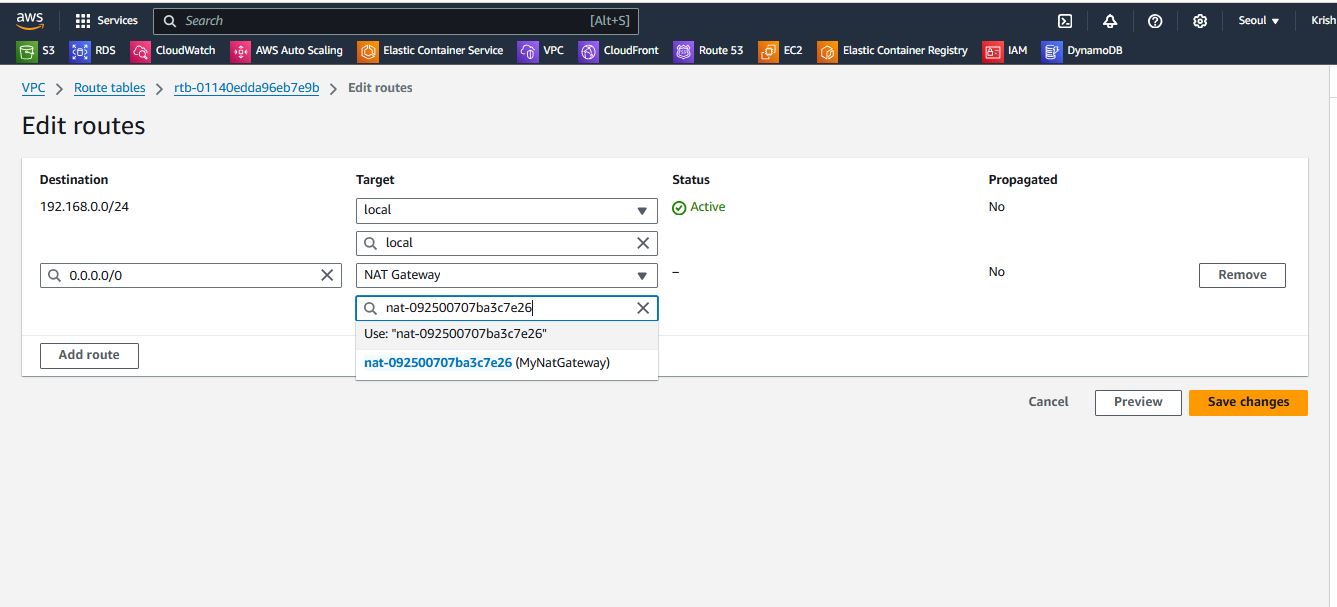
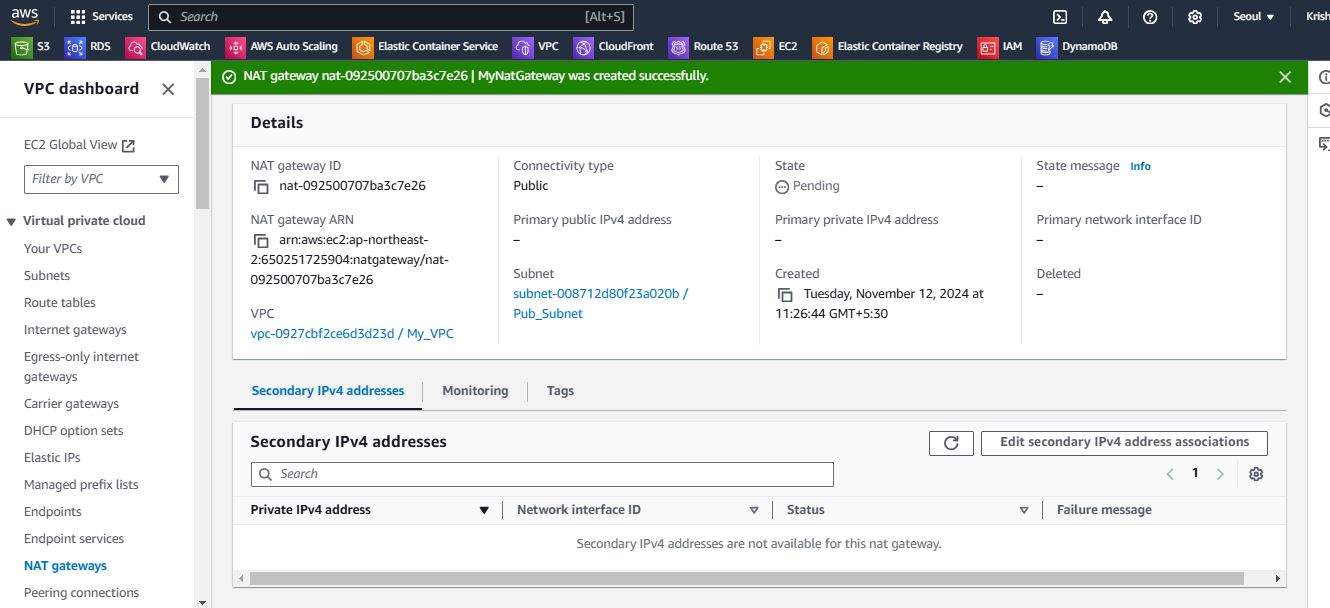


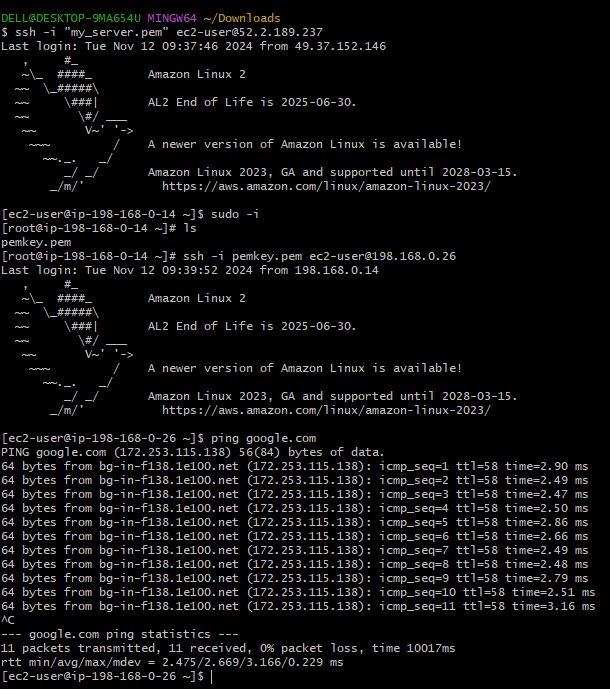


1. COnfigure Nat gateway in public subnet and connect to private Instance

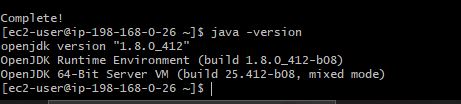


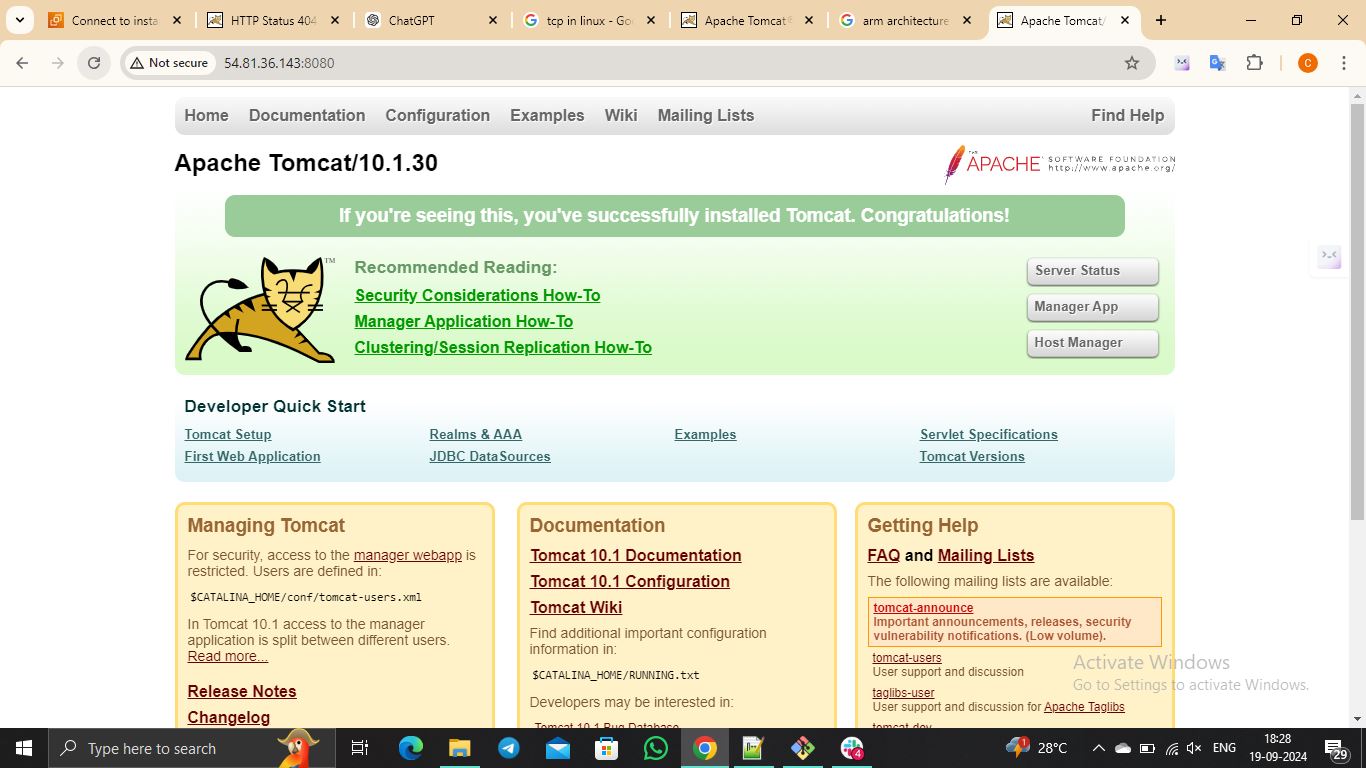


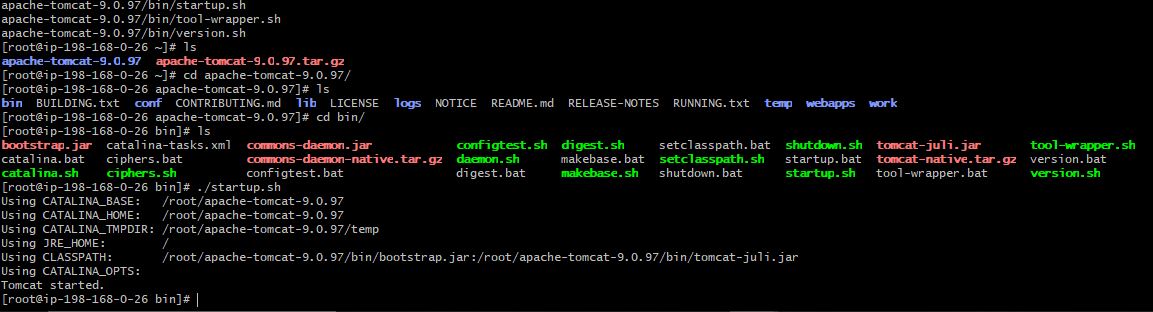




1. Install Apache Tomcat in private ec2 and deploy a sample app.







1. Configure VPC flow logs and store the logs in s3 and cloudwatch.

