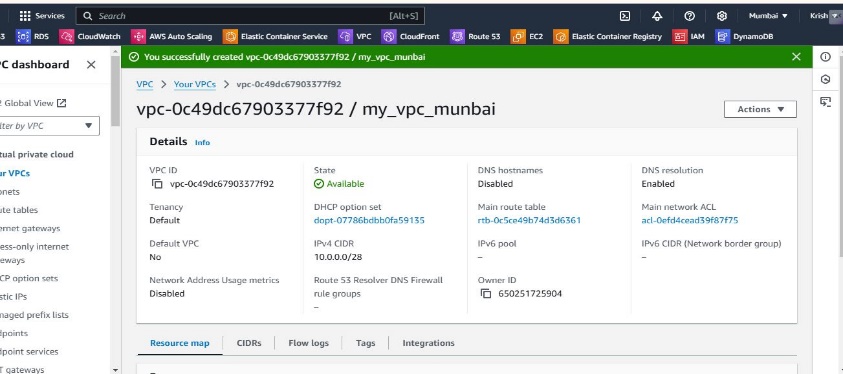
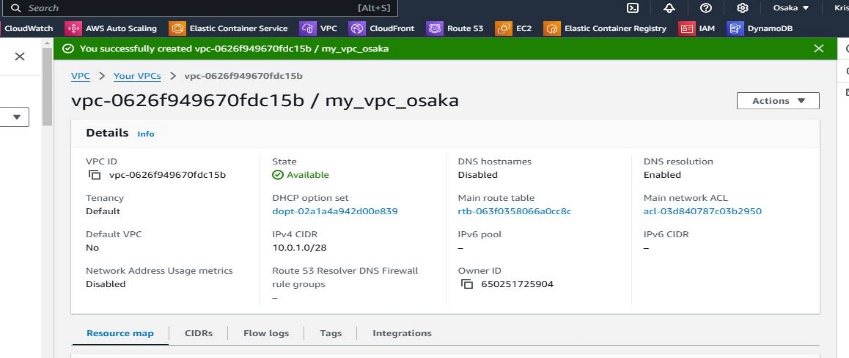
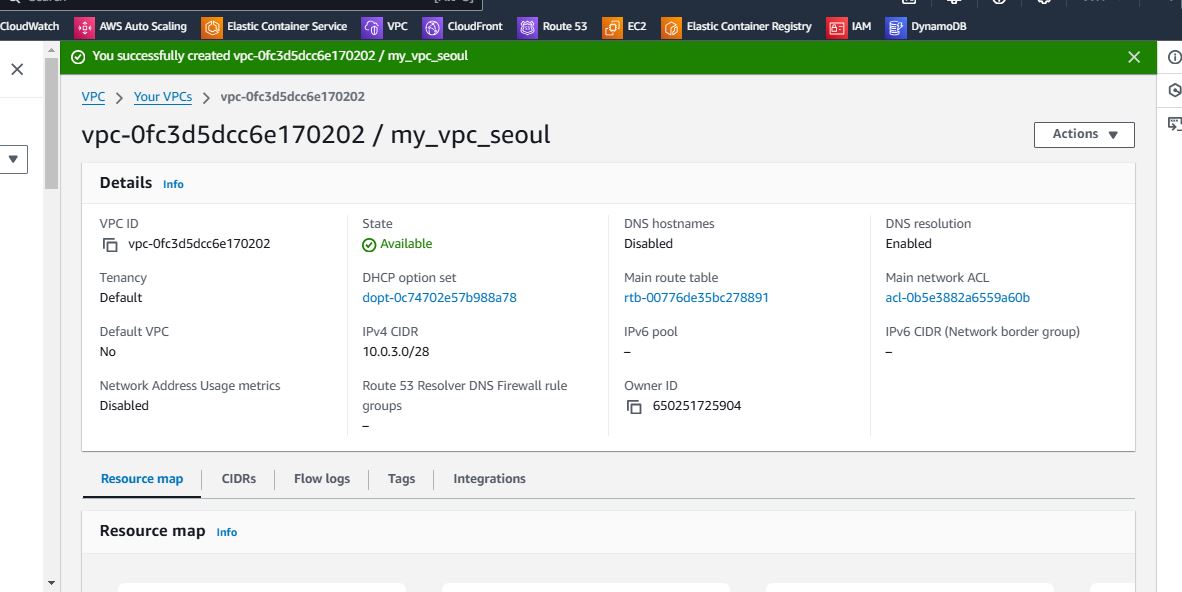
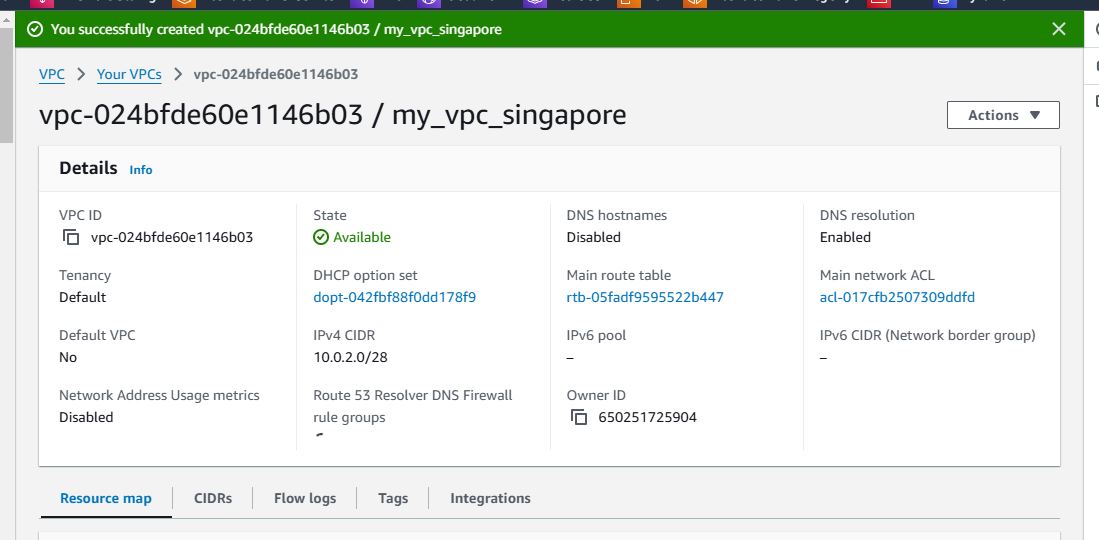
Design and deploy a scalable network architecture using AWS Transit Gateway to simplify network connectivity between multiple VPCs. Configure VPC endpoints to securely access AWS services without internet gateways or NAT gateways, ensuring data privacy and minimizing exposure to external threats.

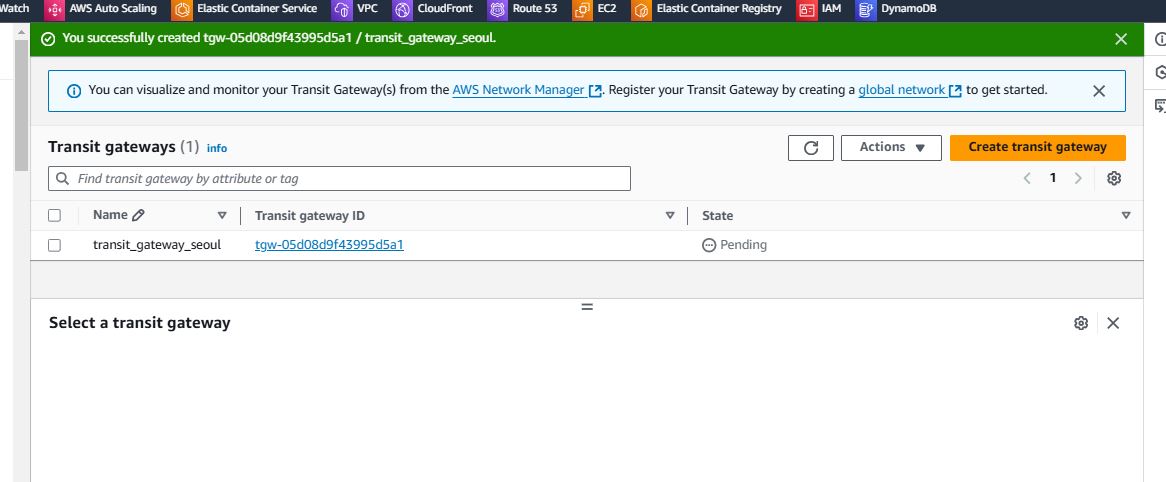
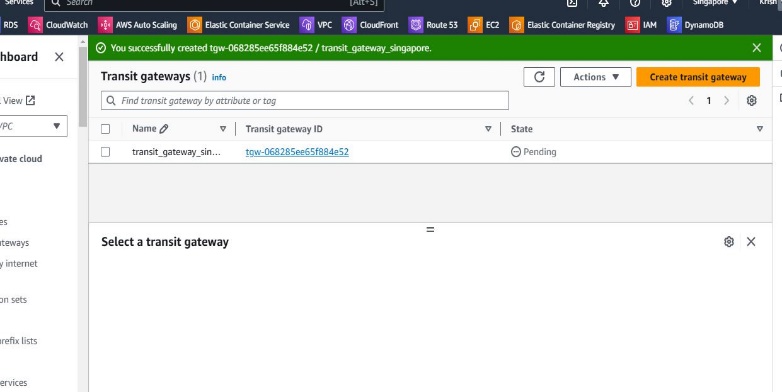
### Create Custom VPCs in Each Region

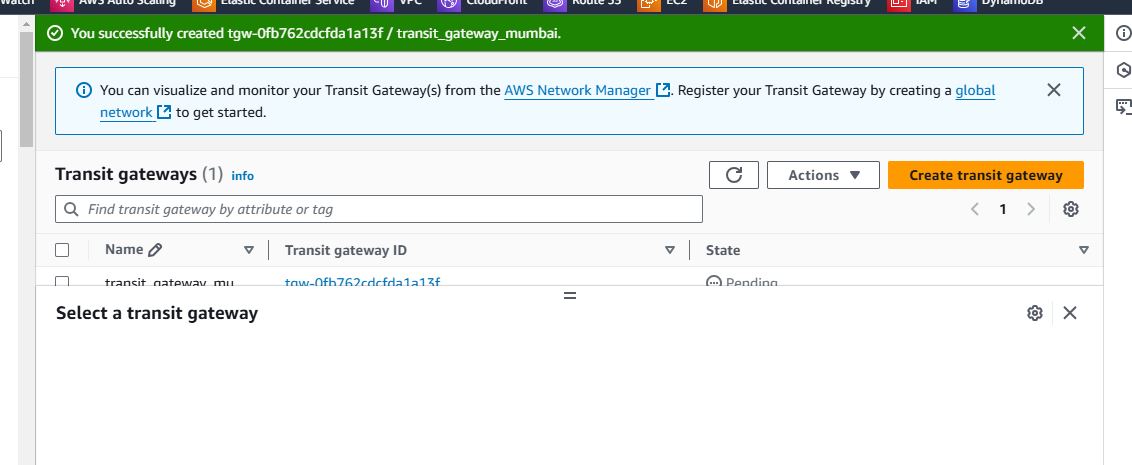
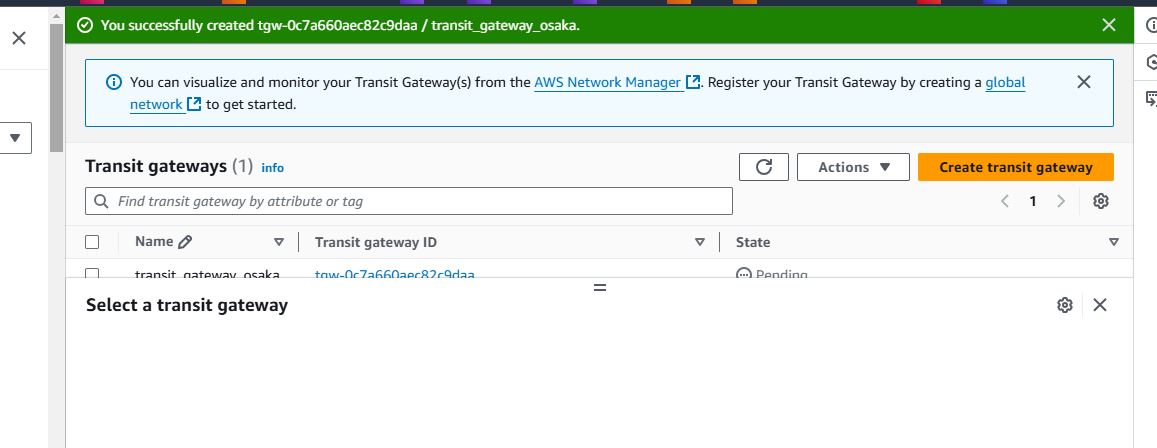
* We will create 4 custom VPCs in 4 different AWS regions. These VPCs will host the resources for different departments or applications.
* VPC CIDR Blocks (example):
  + **My-VPC 1**: 10.0.0.0/16 (Region 1)
  + **My-VPC 2**: 10.1.0.0/16 (Region 2)
  + **My-VPC 3**: 10.2.0.0/16 (Region 3)
  + **My-VPC 4**: 10.3.0.0/16 (Region 4)



Create Transit Gateway

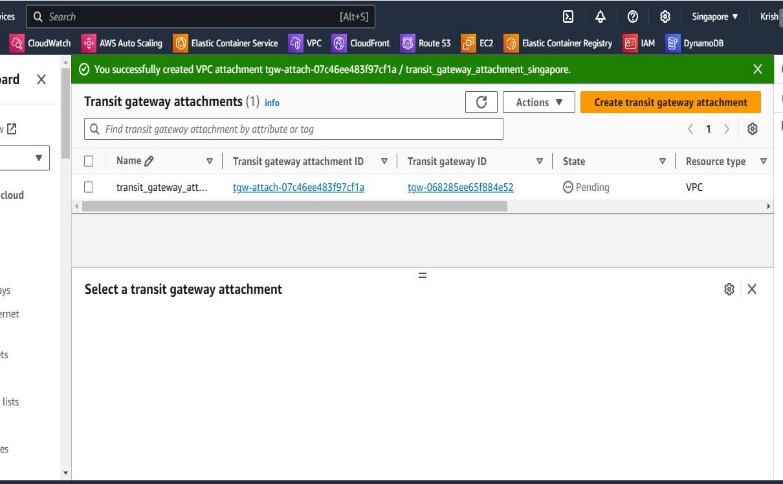
The Transit Gateway will be deployed in one region (typically in Region 1) and then peered with VPCs in the other regions.

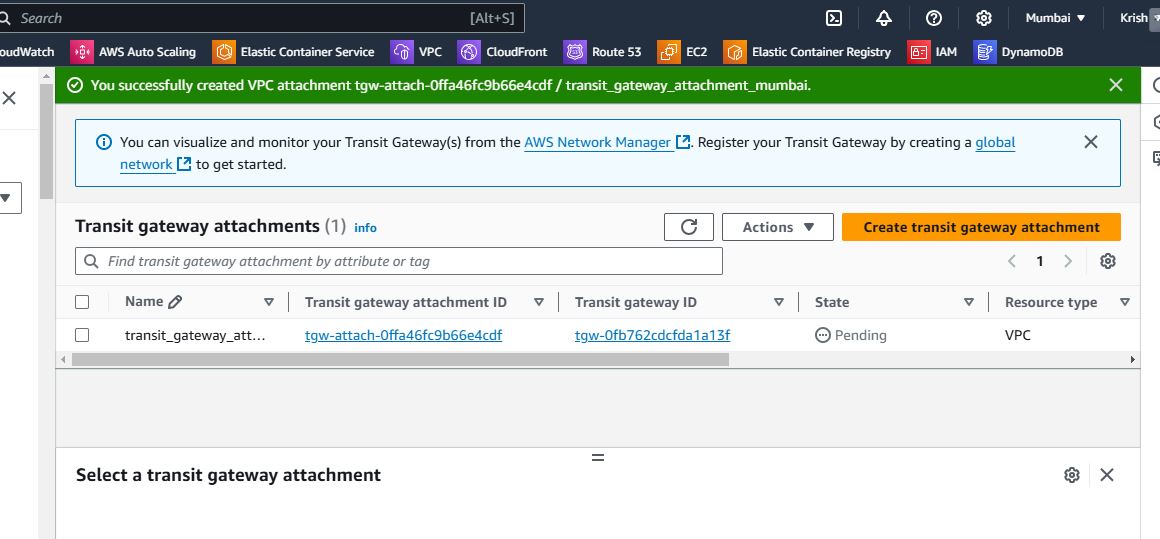
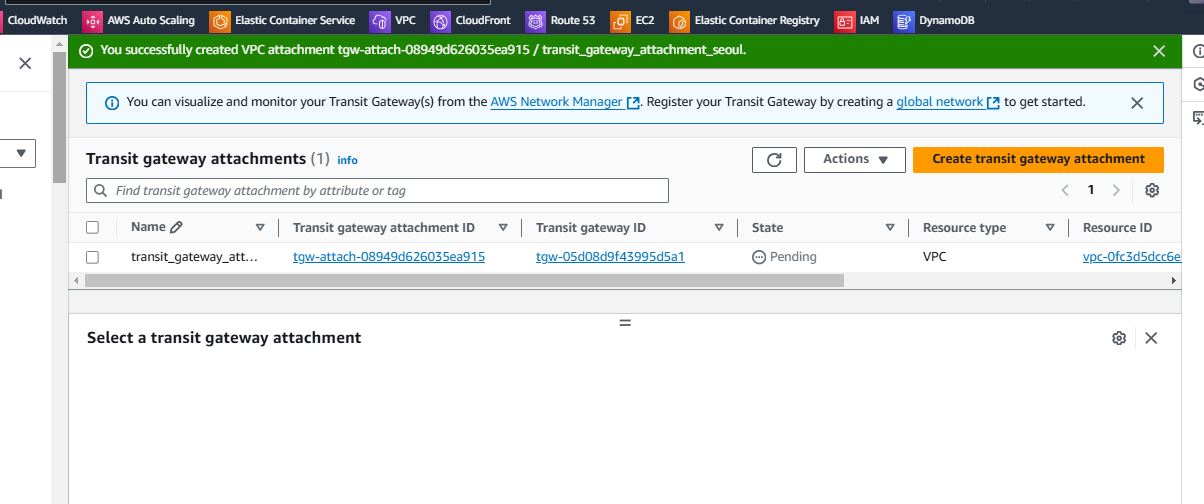
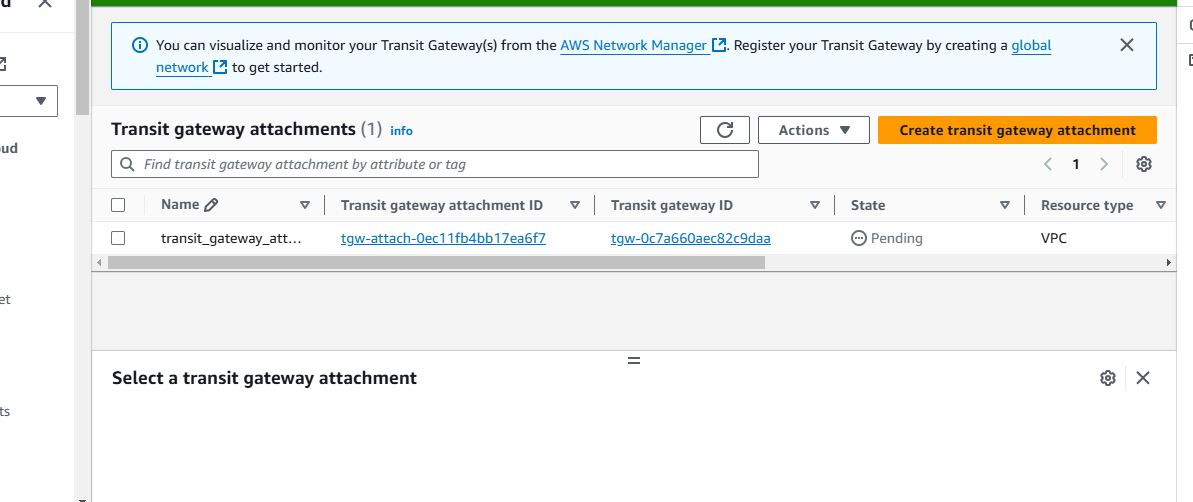


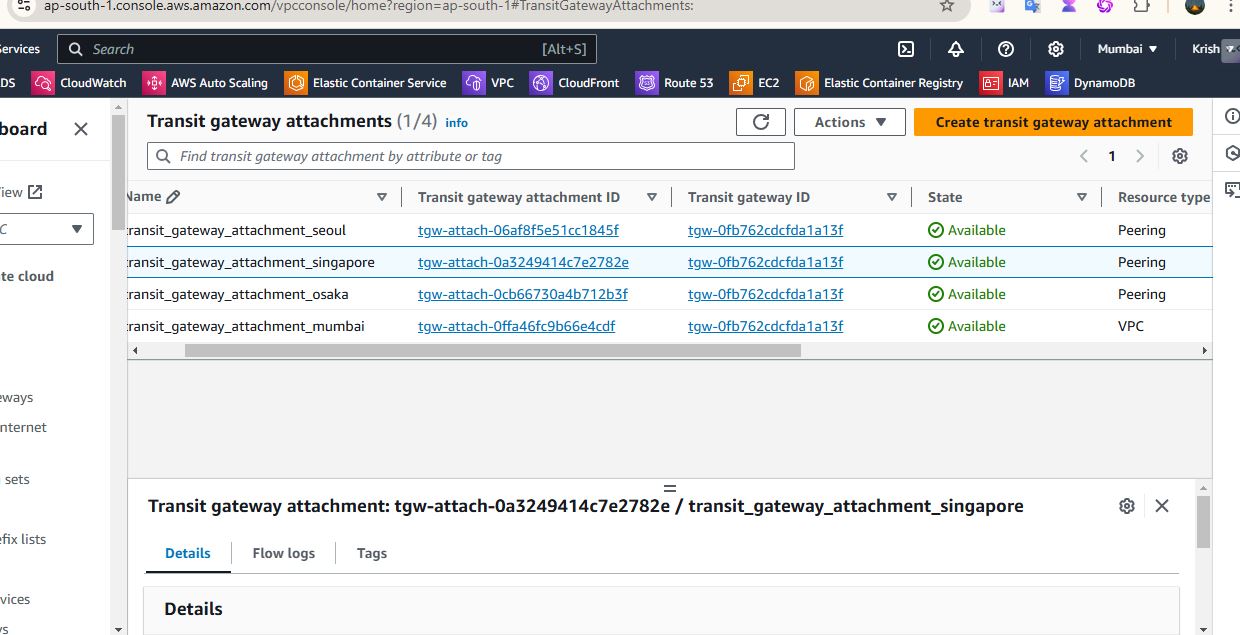


**Attach VPCs to Transit Gateway**

Attach each VPC to the Transit Gateway via VPC attachments.



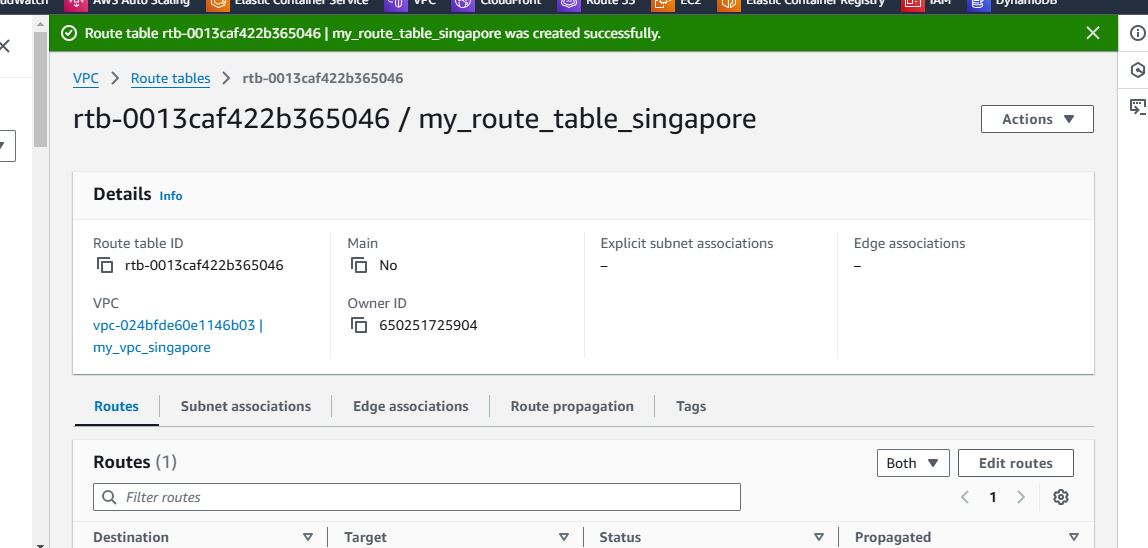
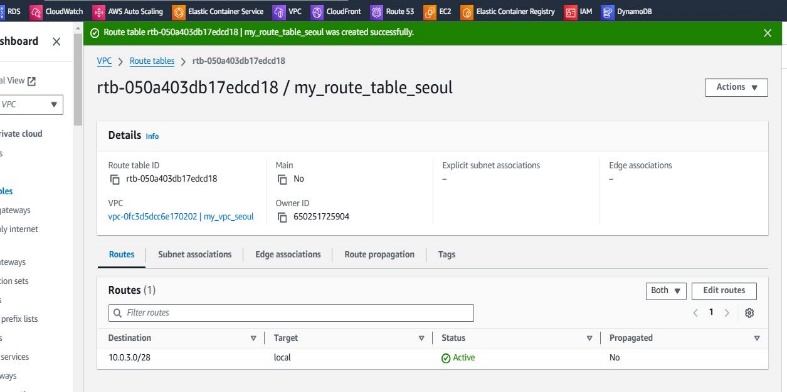


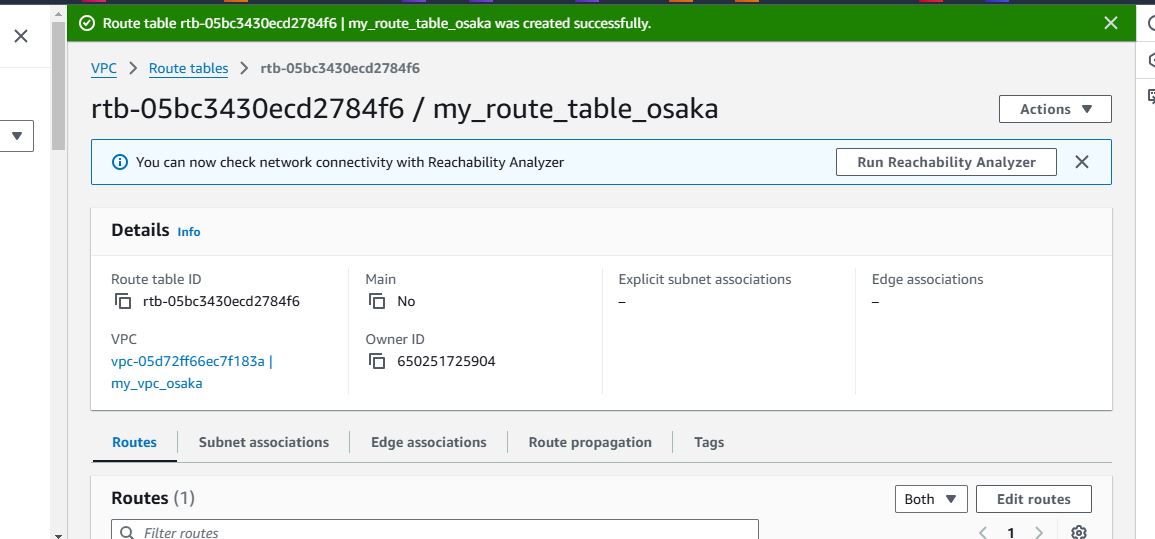
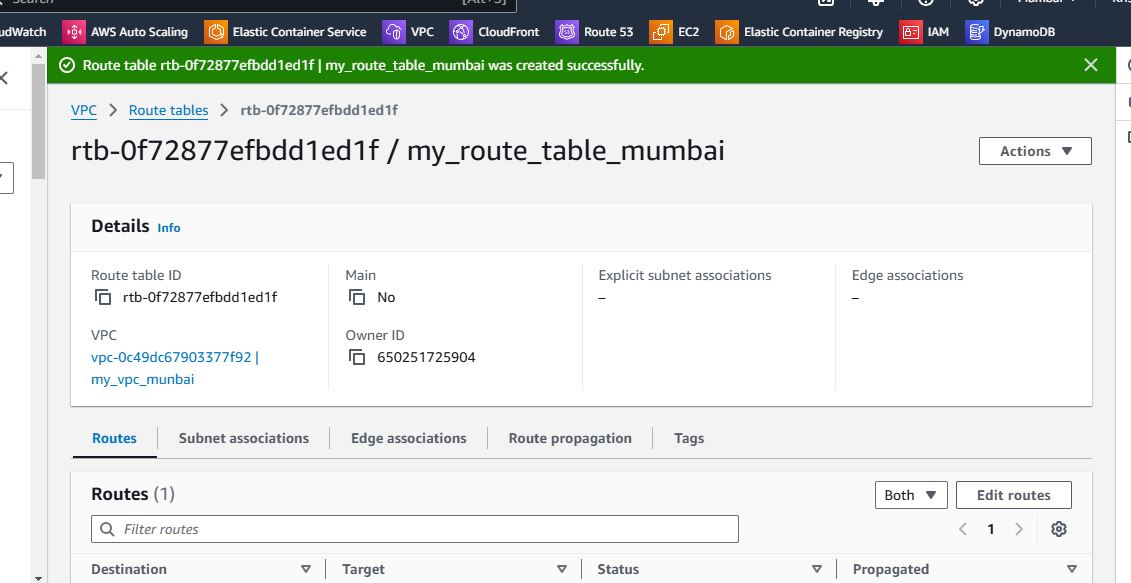


Transit Gateway all attachments in one region

**Set Up Routing Between VPCs and AWS Transit Gateway**

Configure route tables in each VPC and the Transit Gateway to ensure proper routing of traffic between VPCs





Test the Connectivity

Launch EC2 instances in each VPC and test the connectivity between them using ping or other network tools.

