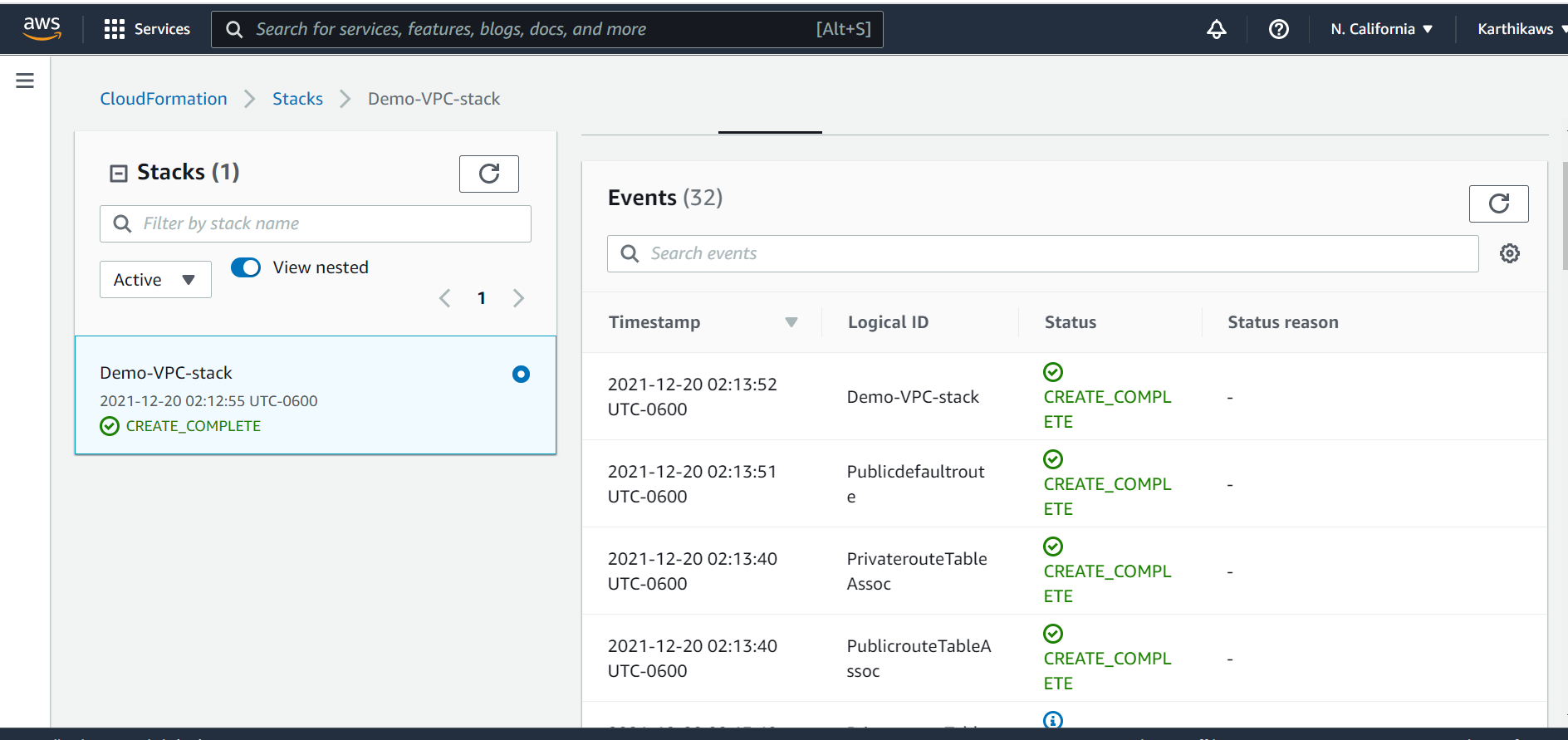
**Employee Name: Karthikraja Venkatesh**

**Employee Id: 662056**

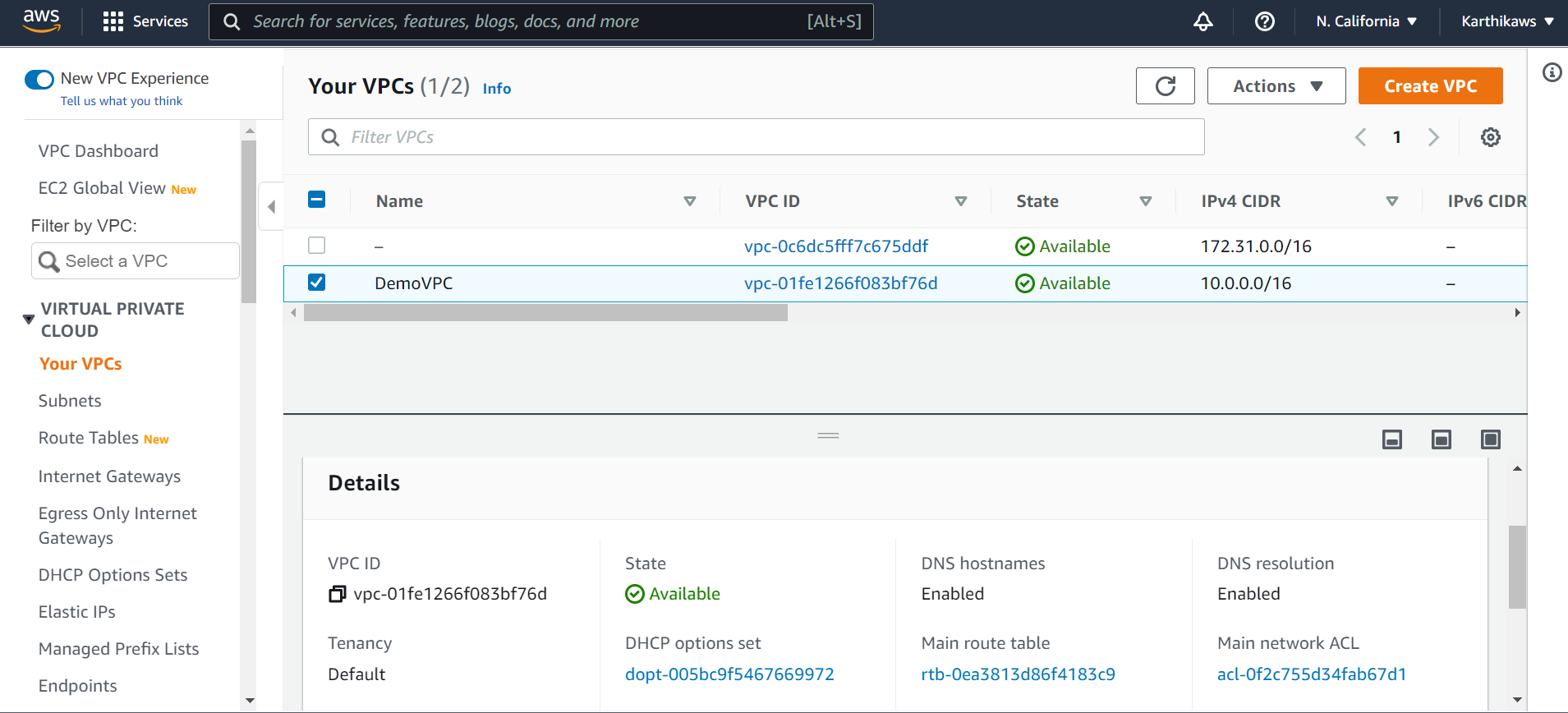
**Step1: Cloud Formation**

Creating VPC with one public subnet and one private subnet, routing table and internet gateways

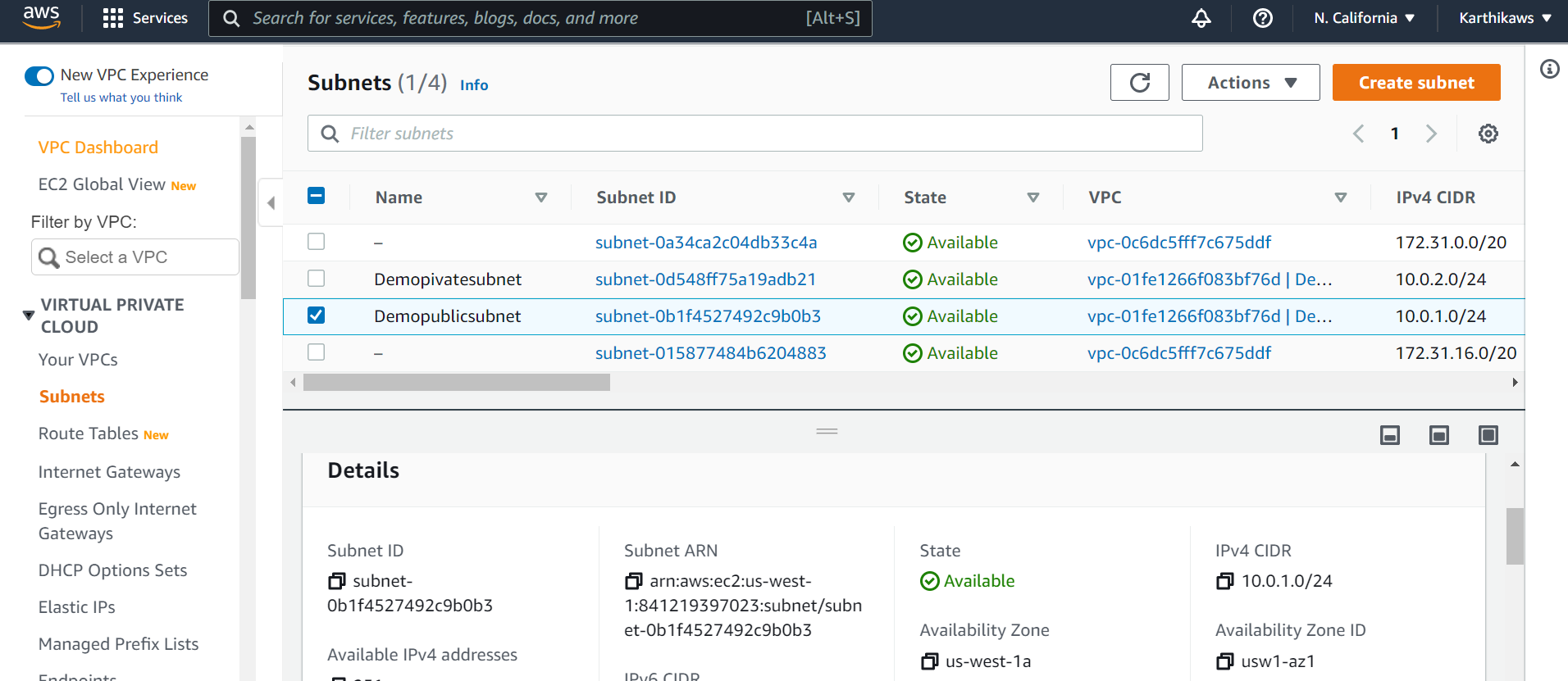
Creation of stack with template:



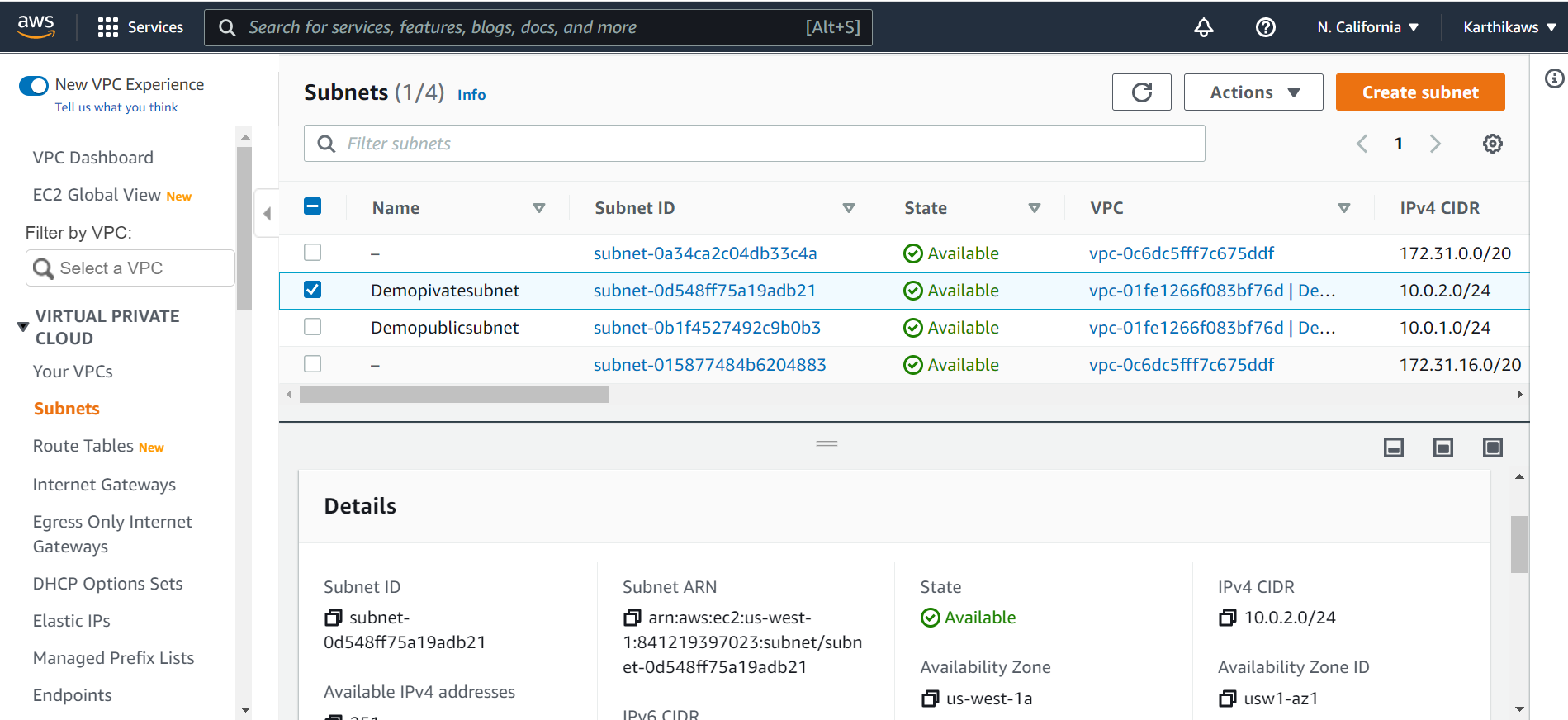
VPC:



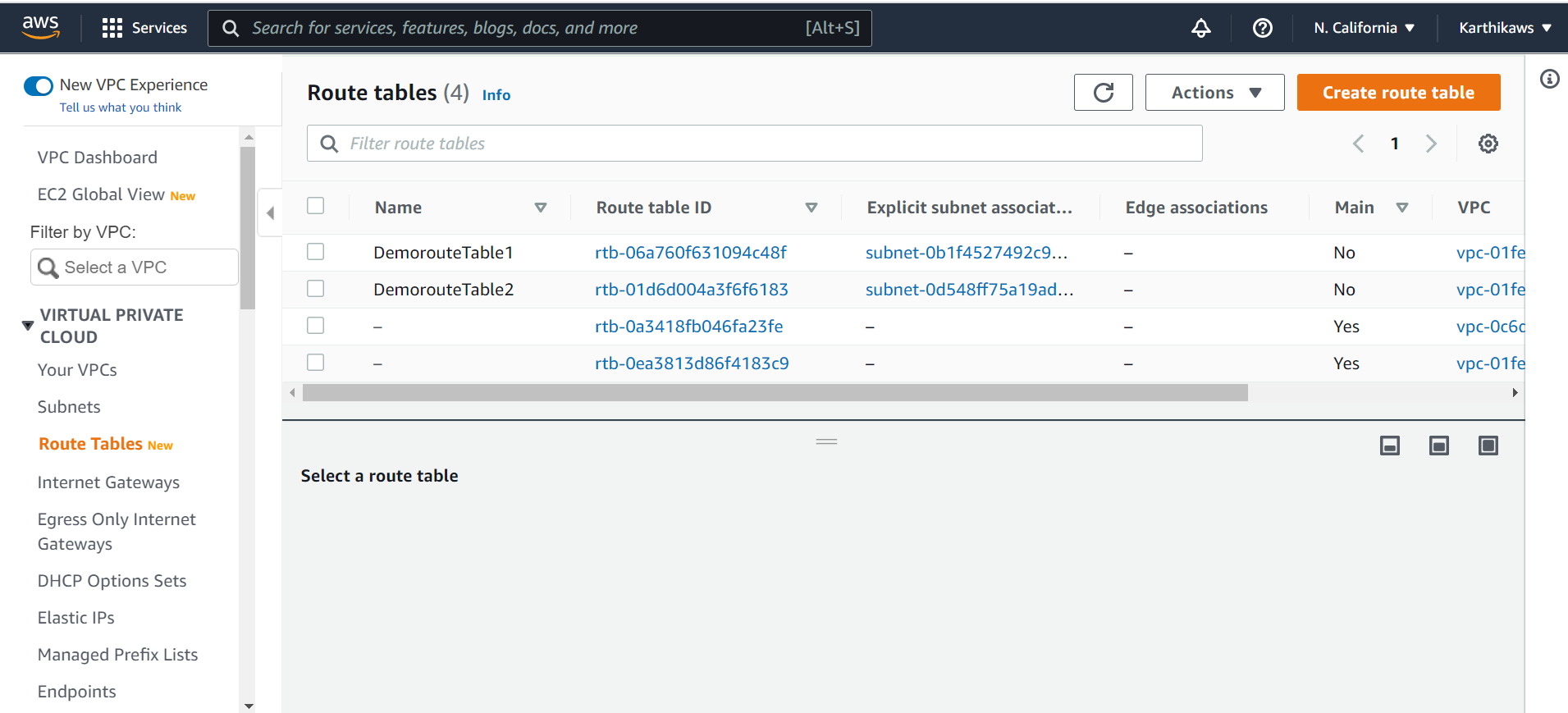
Public Subnet:



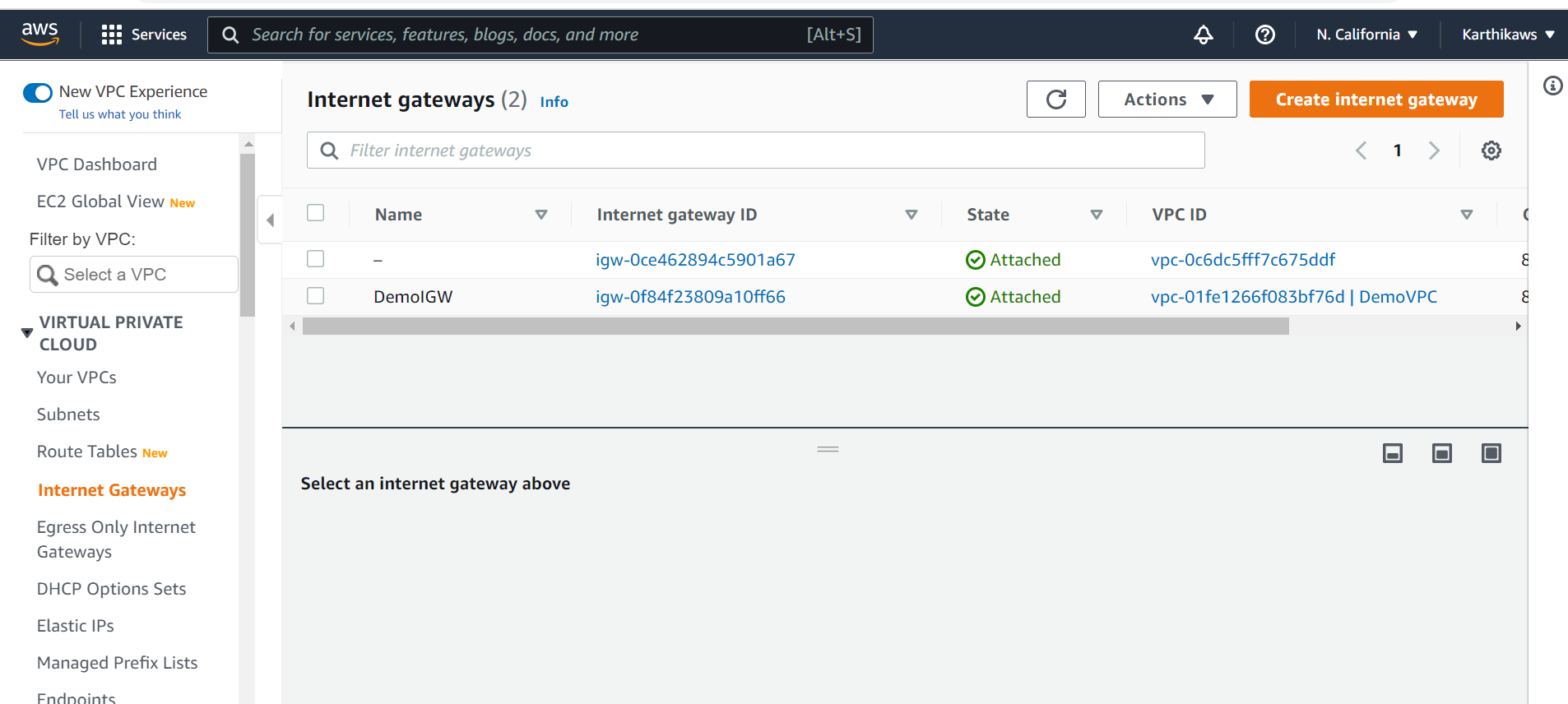
Private Subnet:



Routing Table:

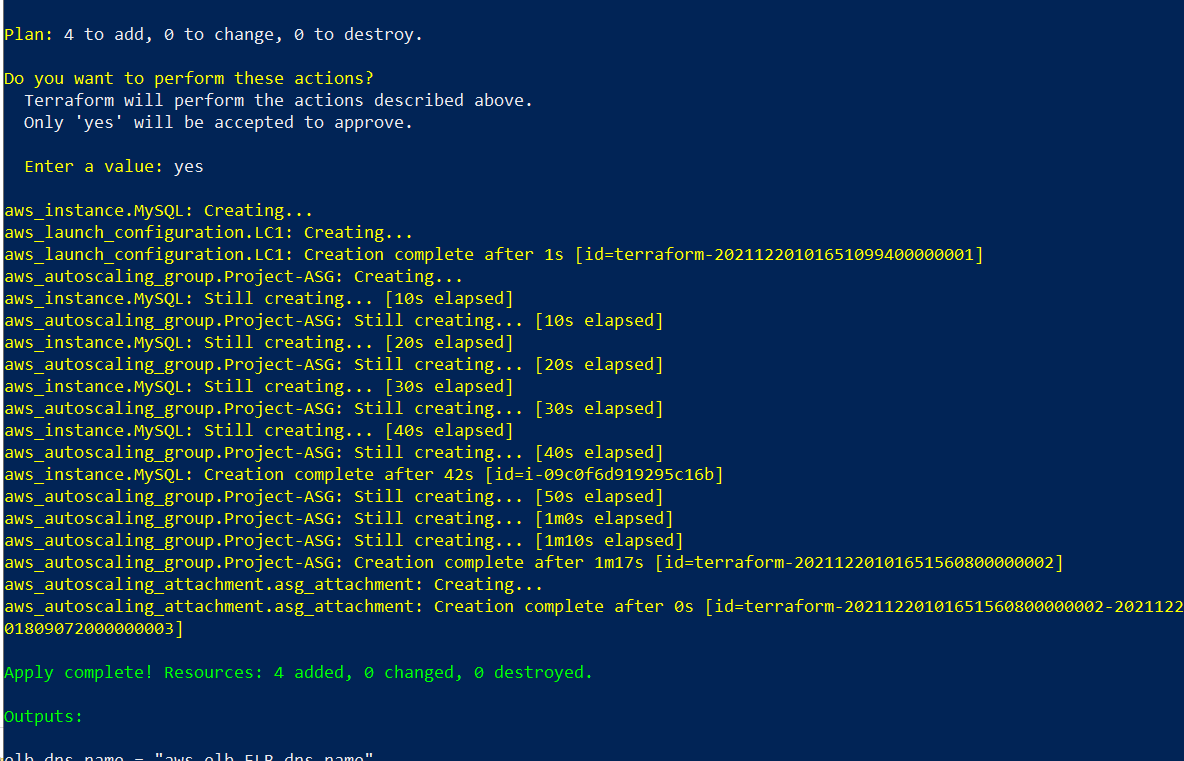


Internet Gateway:

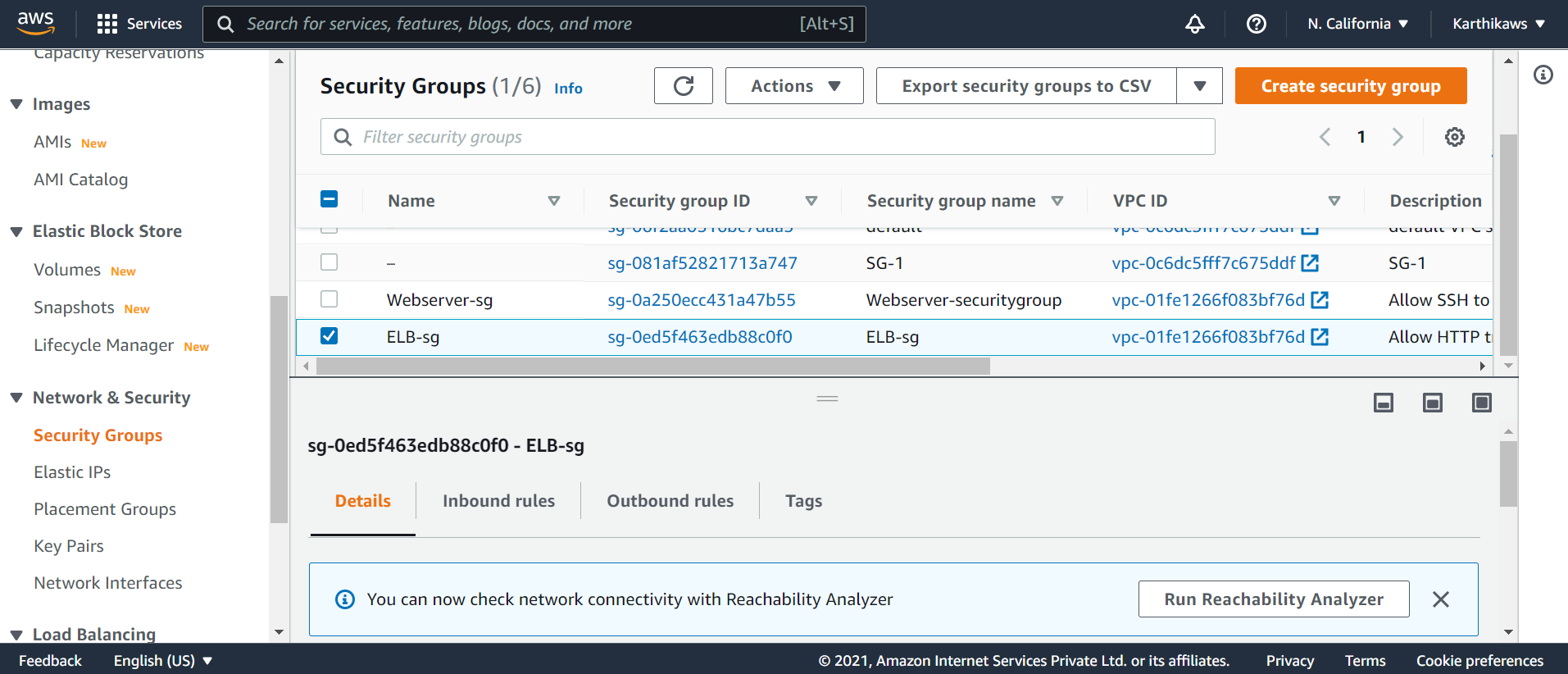


**Step2: Terraform**

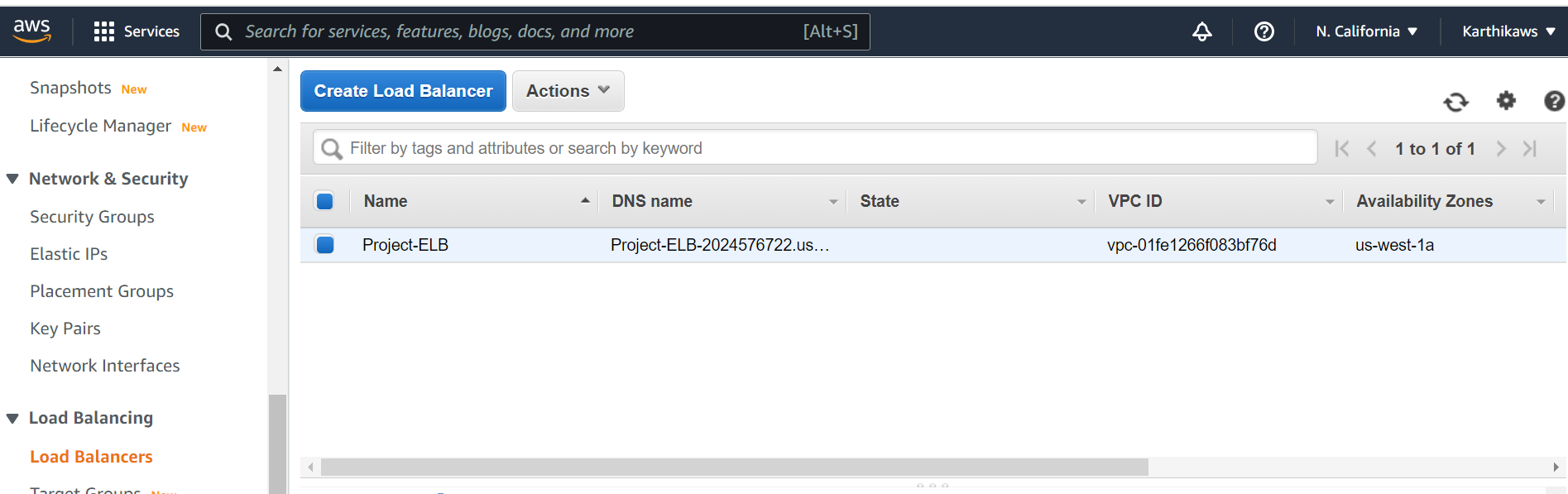
Create application infrastructure using auto scaling group along with Layer 4 load balancer



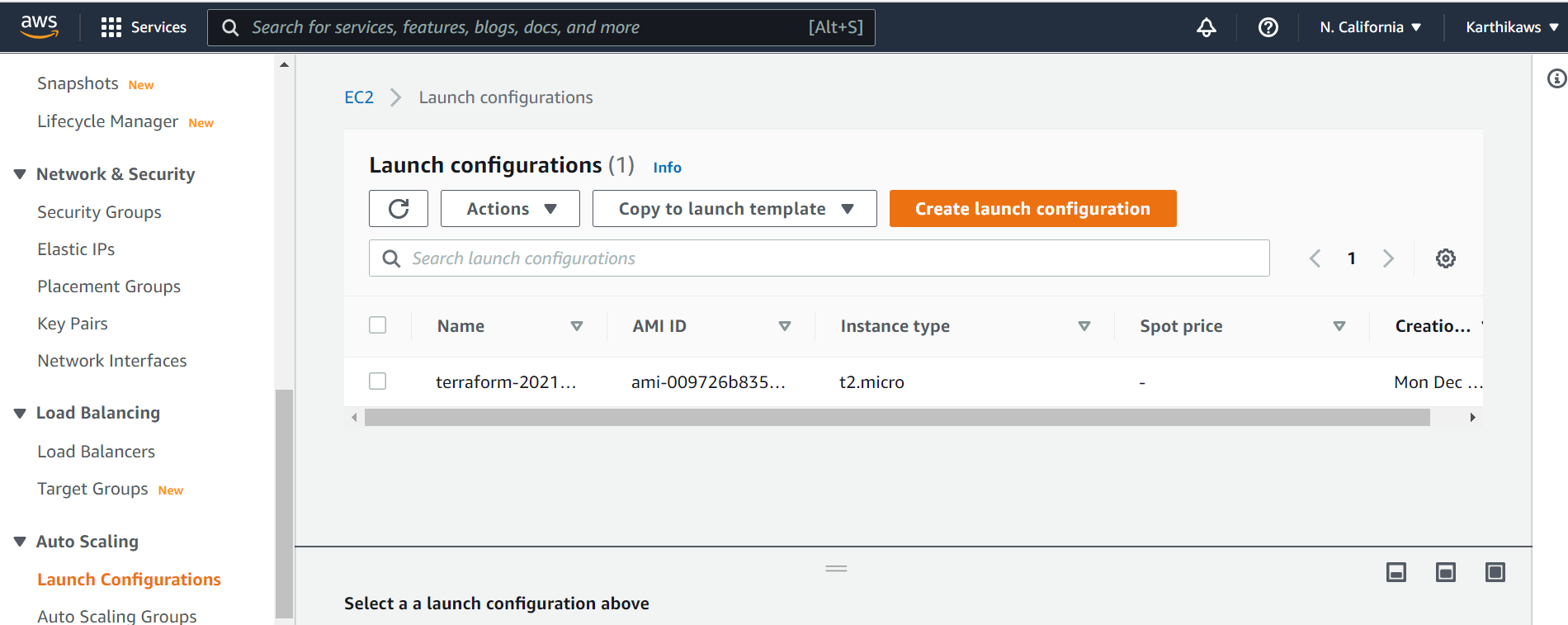
Security Groups:



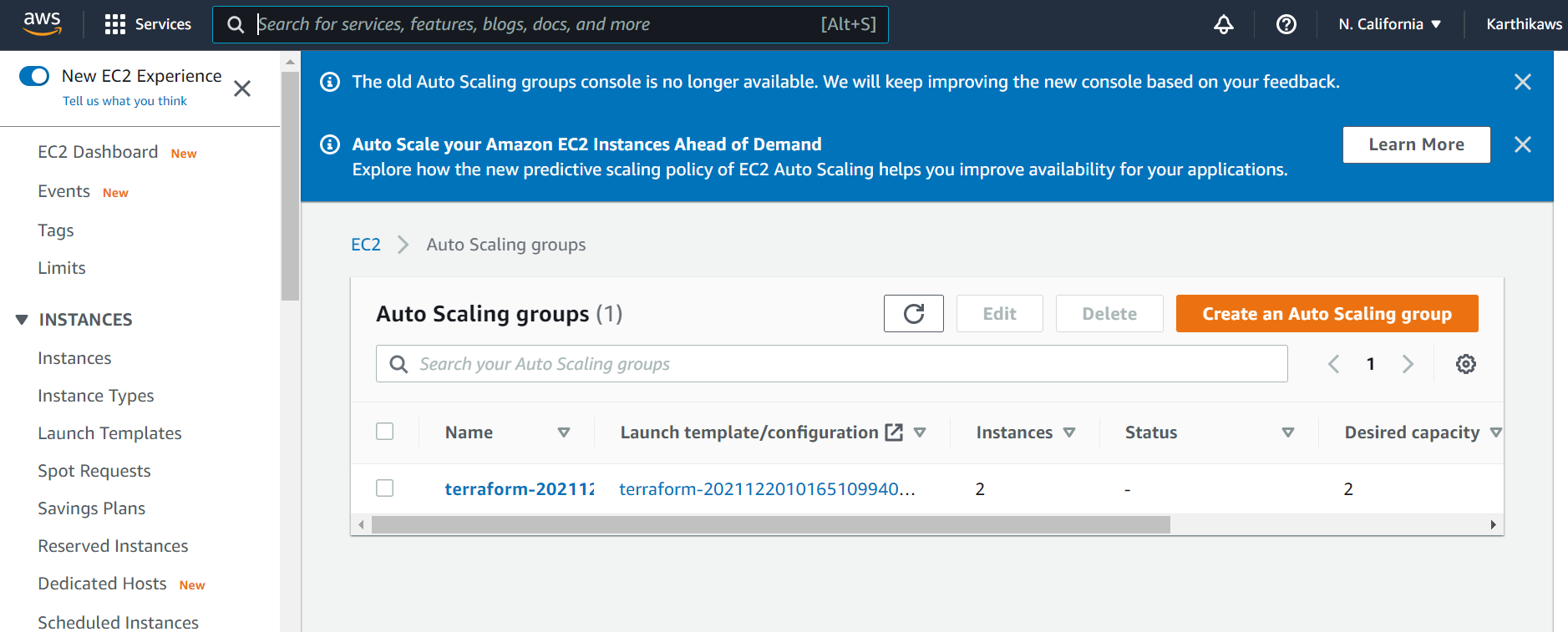
Load Balancer:



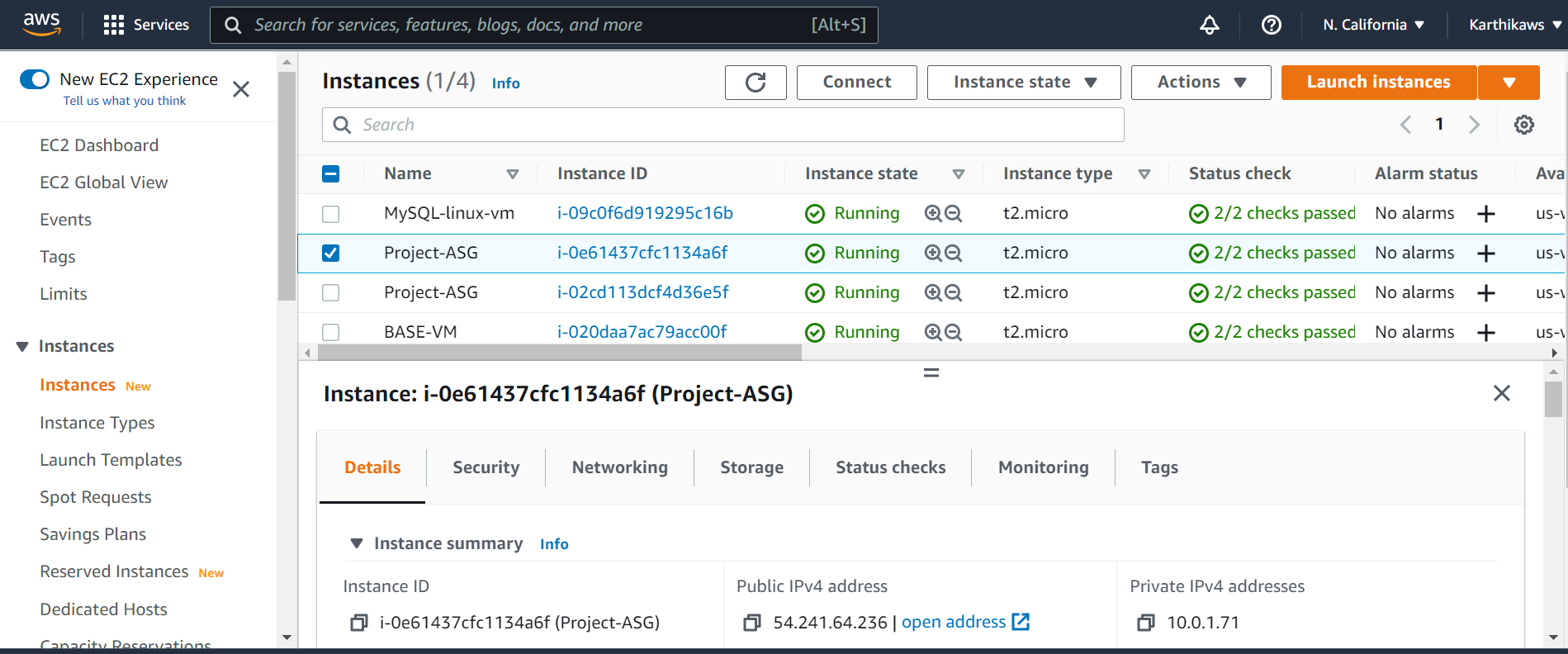
Launch Configuration:



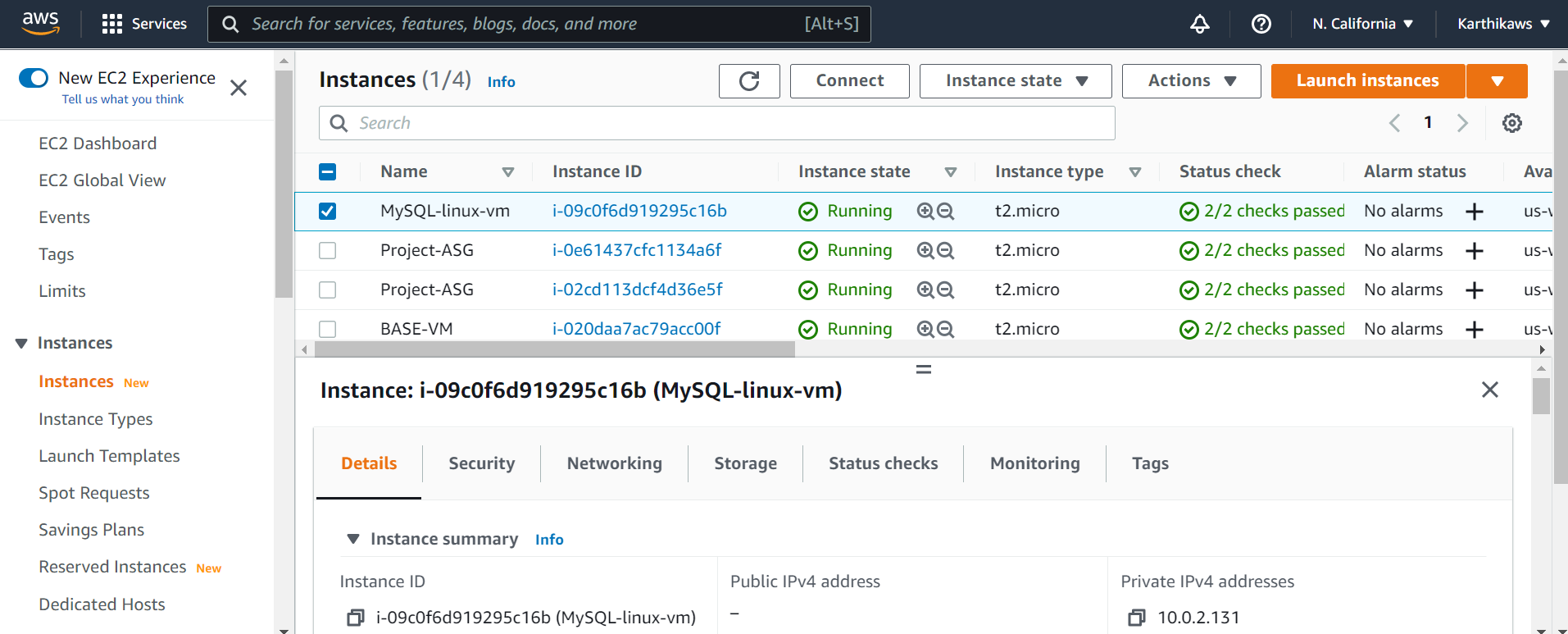
Auto scaling group:



Webcluster instances

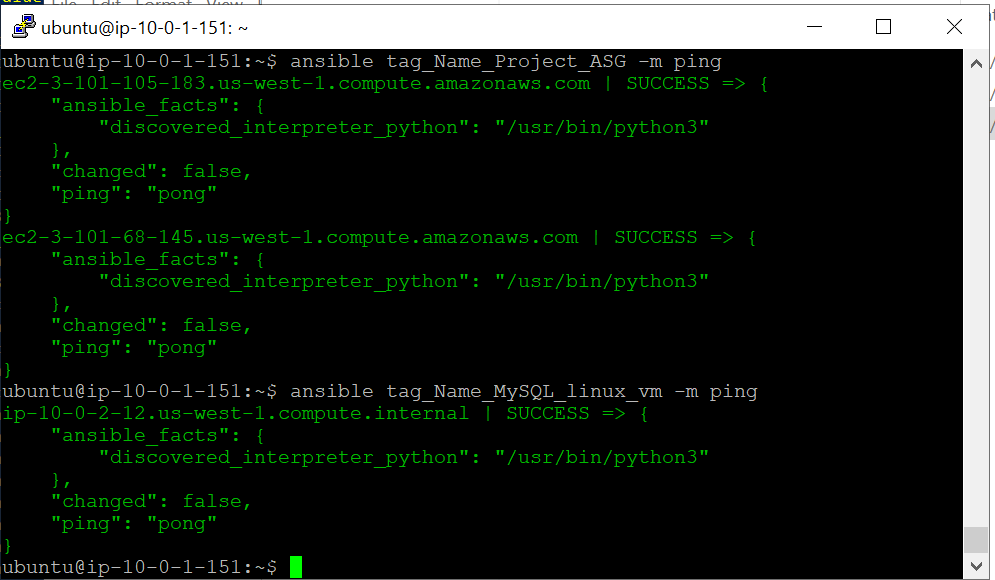


MYSQL instances

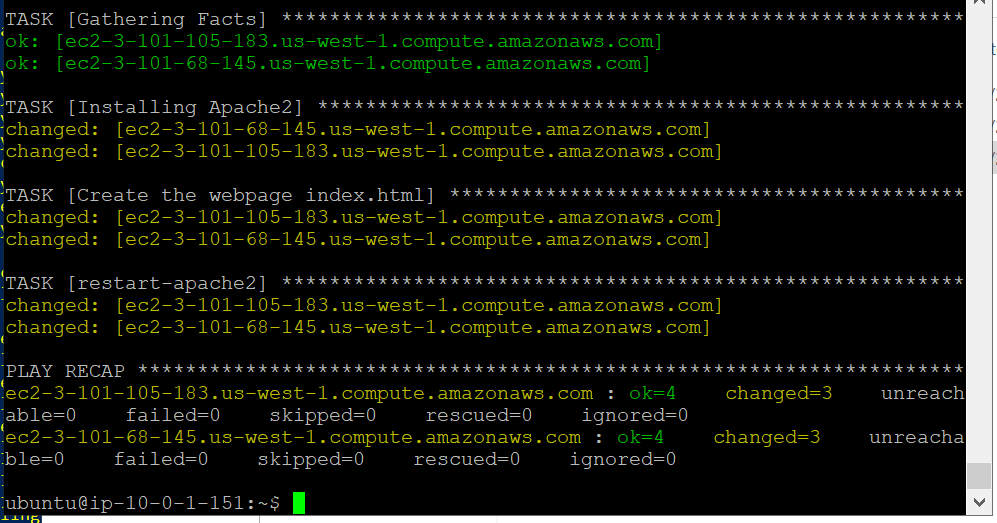


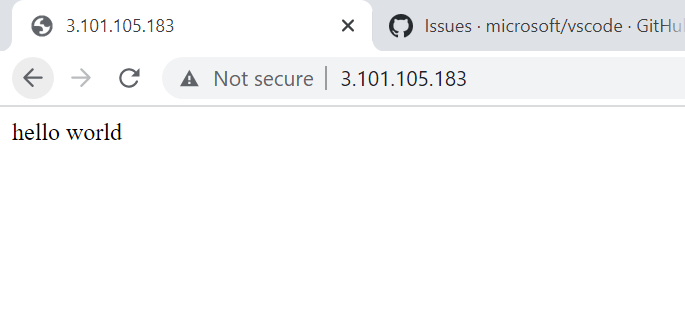
**Step3: Ansible**

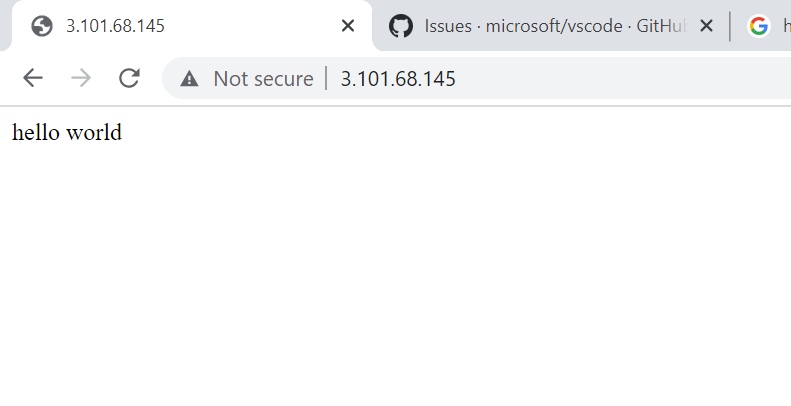
ASG EC2 instance and private instance set as remote nodes in ansible server



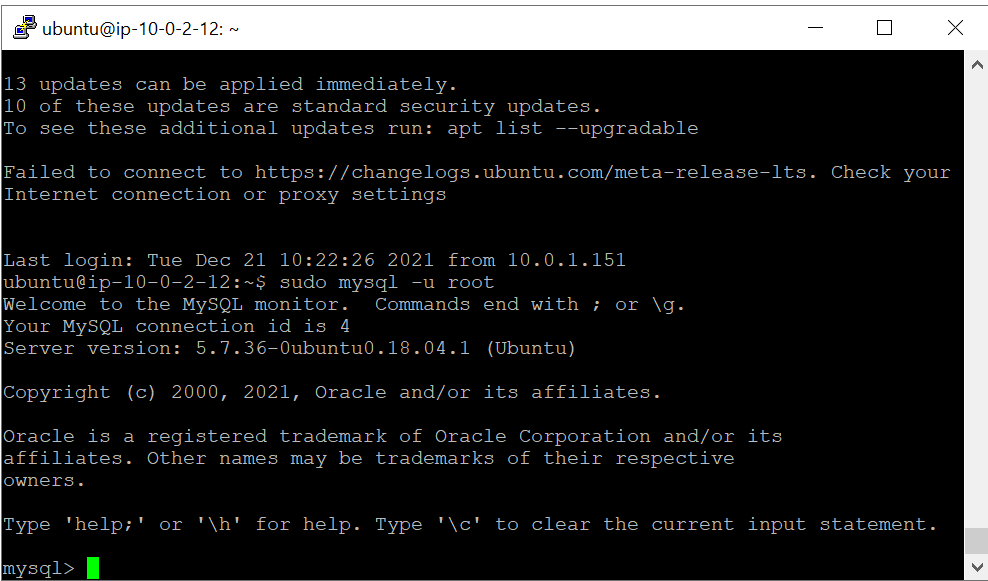
Apache server installed





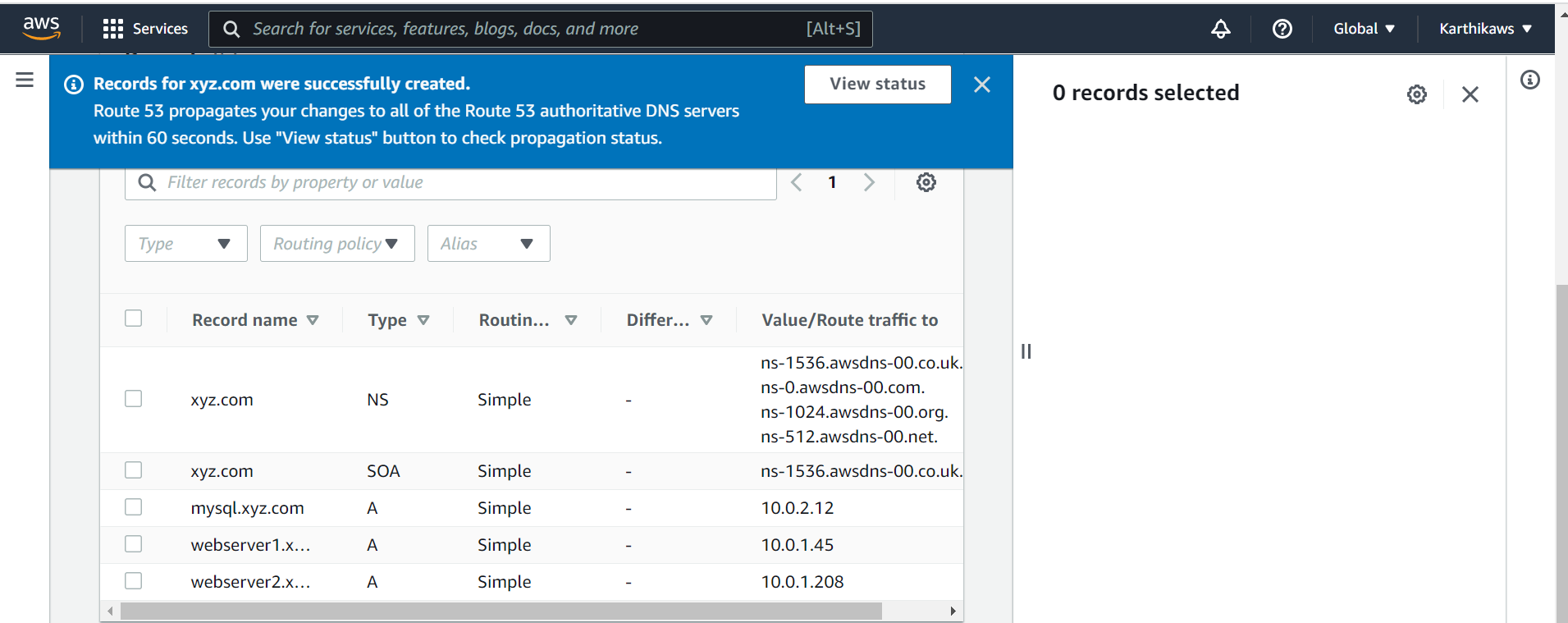


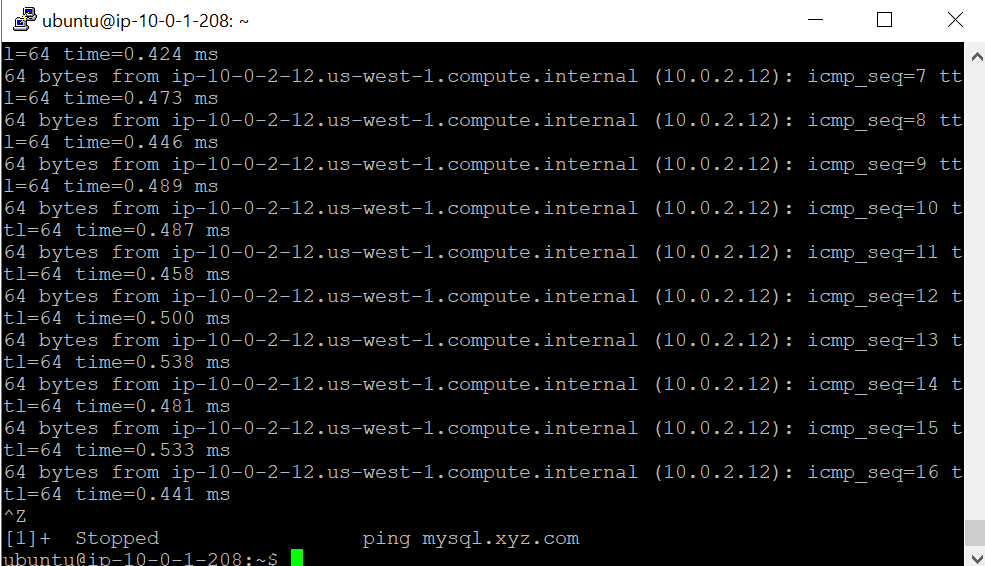
Installed MySQL



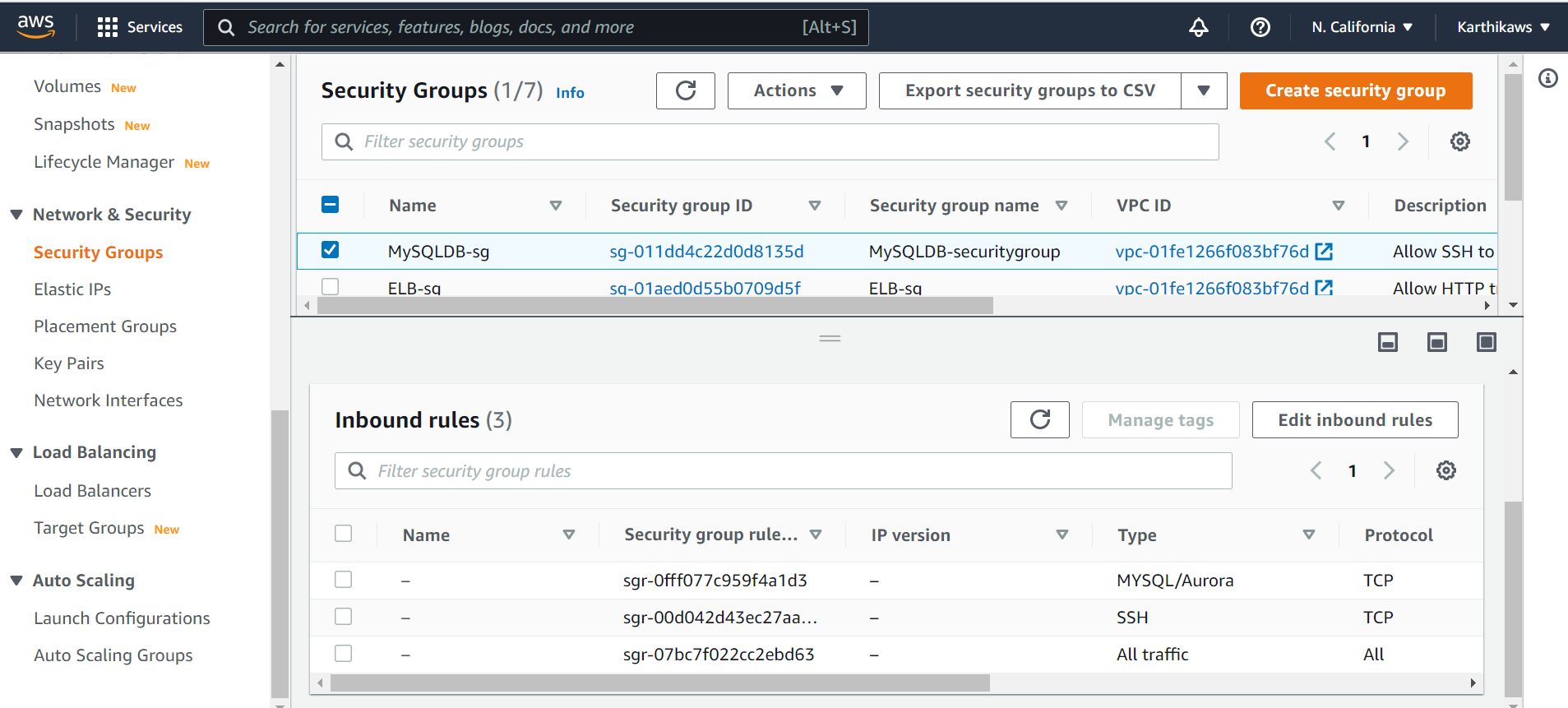
**PART-2 Secure Communication and Security Management for Application (Manual Configuration)**

Route53 Configuration for VPC internal communication:

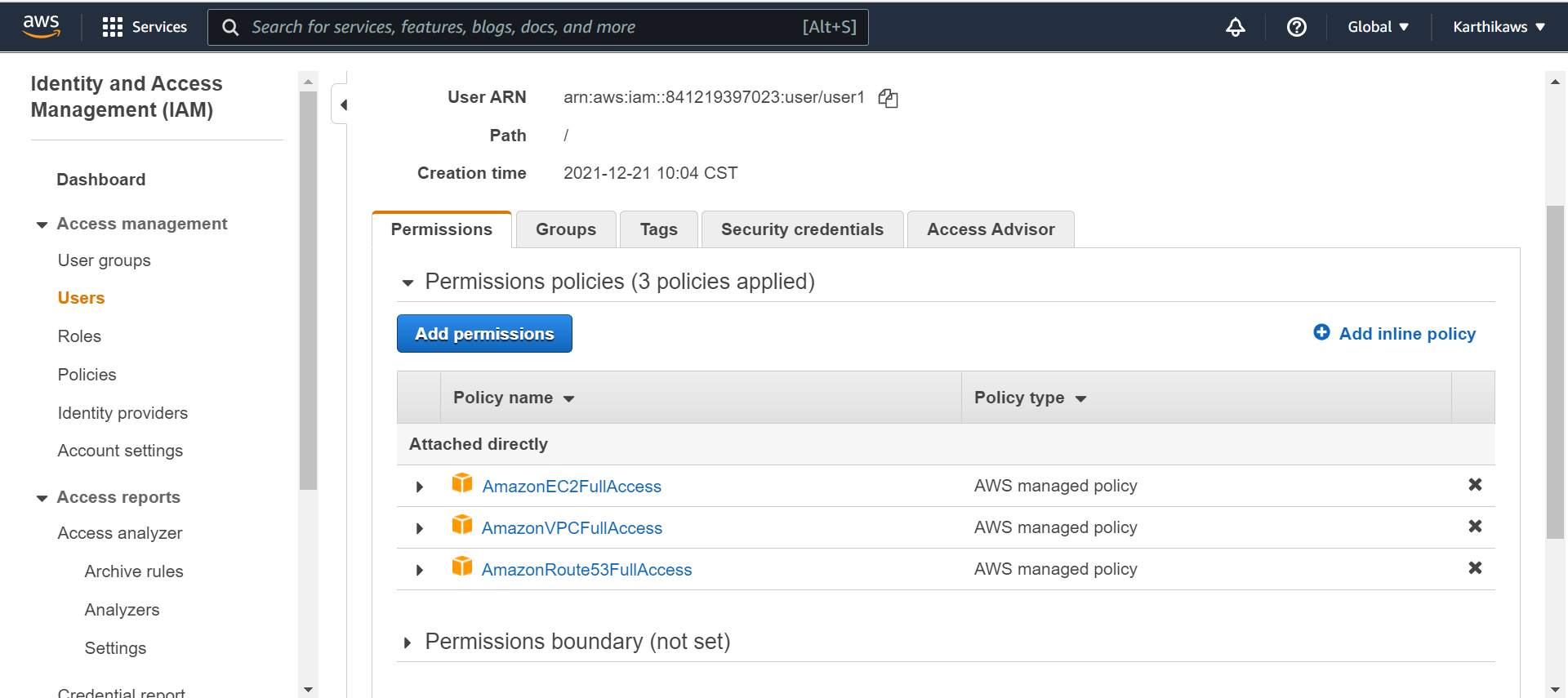




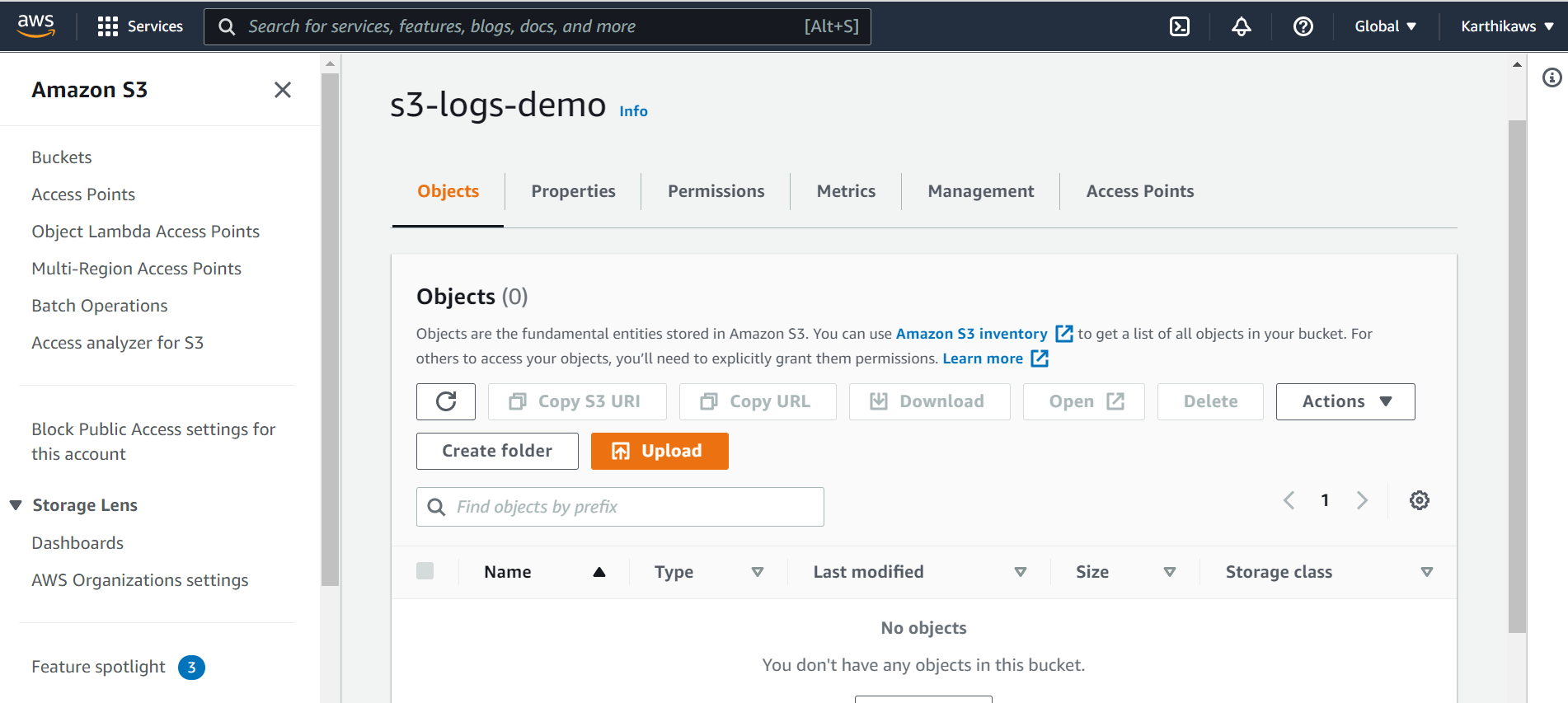
MySQL DB (EC2) should be accessible only from ASG (Webserver Cluster) Hint: Security Groups



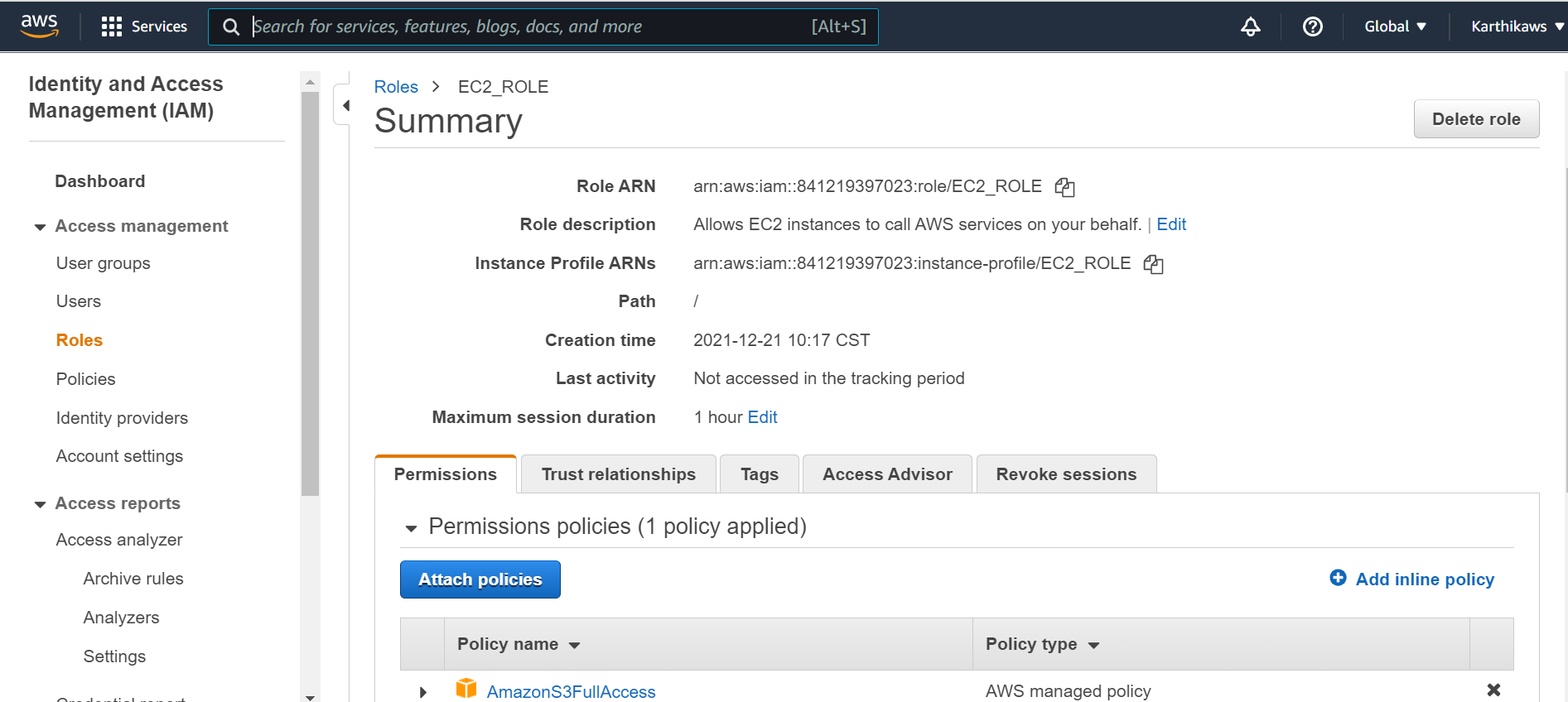
Create user to manage application infrastructure (VPC, Route53, EC2) Hint: IAM Policies

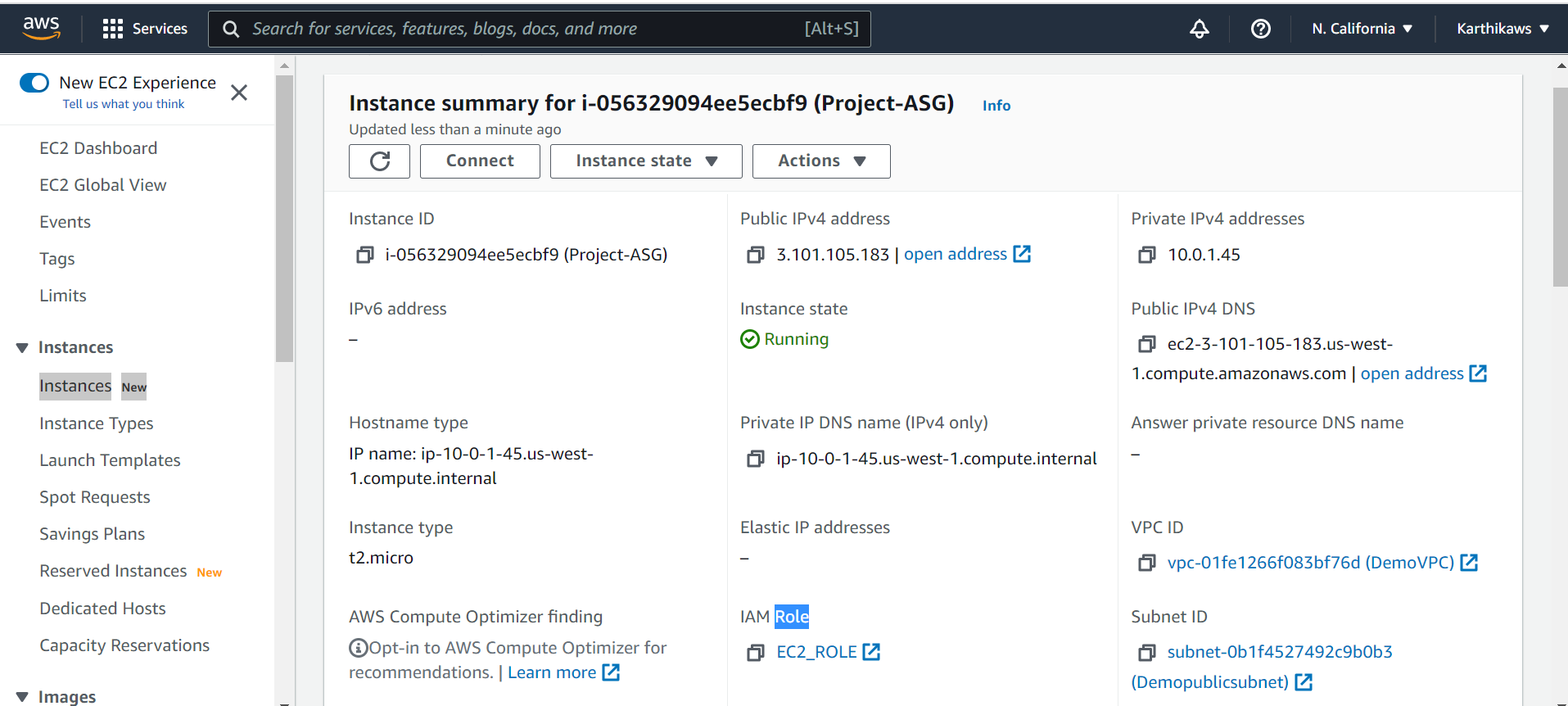


S3 Bucket to store log files:



All EC2 instances should have permission to access S3 Hints: IAM Roles





**PART-3 Monitoring App and App Infrastructure (Manual Configuration)**

**Create a CloudWatch Dashboard to monitor:**

