Autoscaling_Practical(10-09-24) **Savita Nalawade**

- Autoscaling helps to maintain availability of your application

❖ Create template launch configuration

a. Select AMI as Amazon Linux

b. Select instance type “t2.micro”

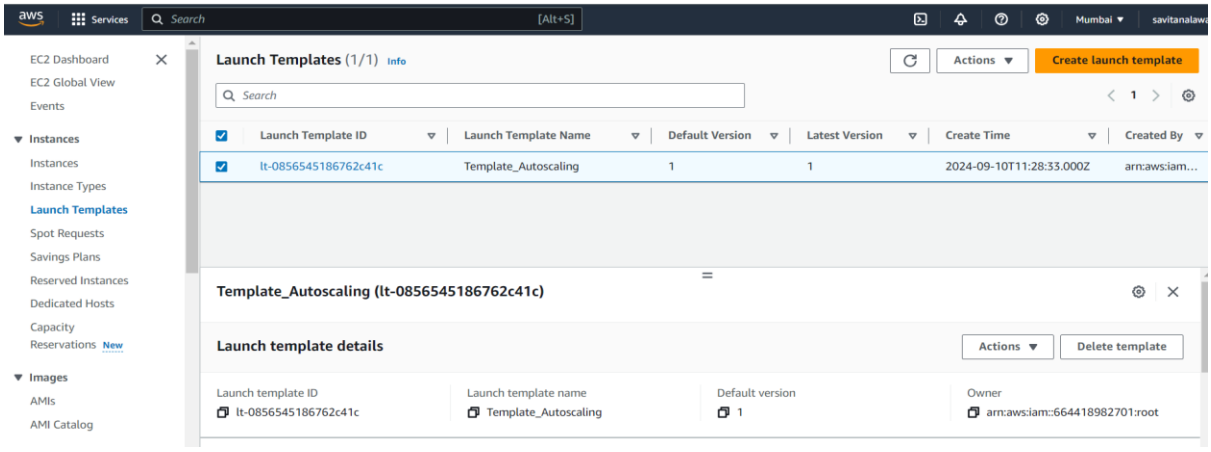
c. Select key-pair

d. Select Security group

e. Select role with s3FullAccess

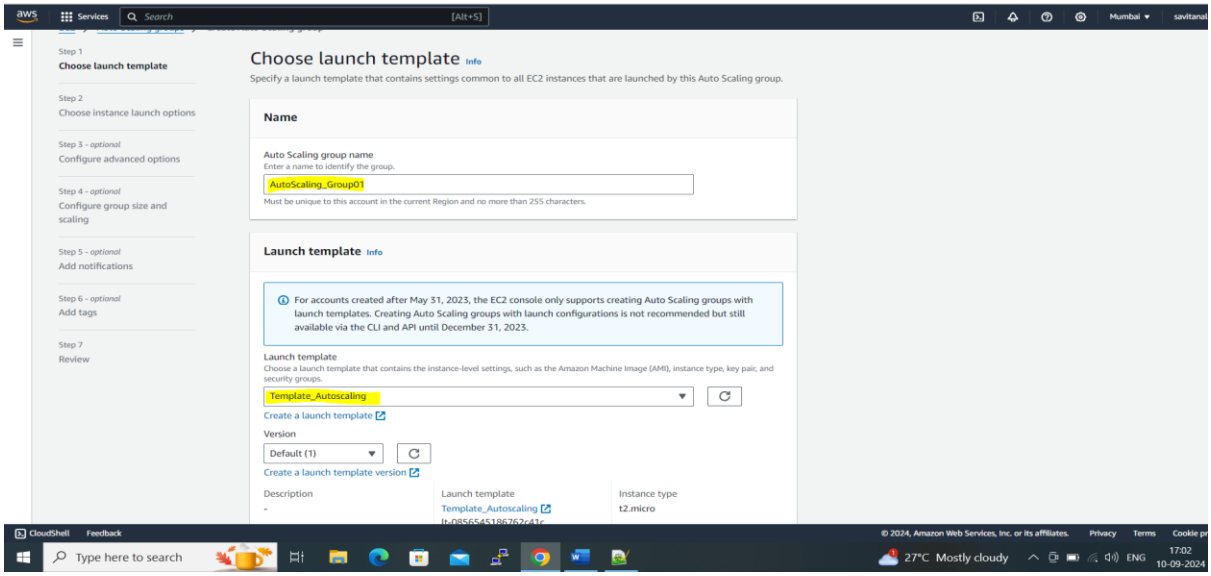
f. Advance setting -> add bootstrap script

g. Create launch template

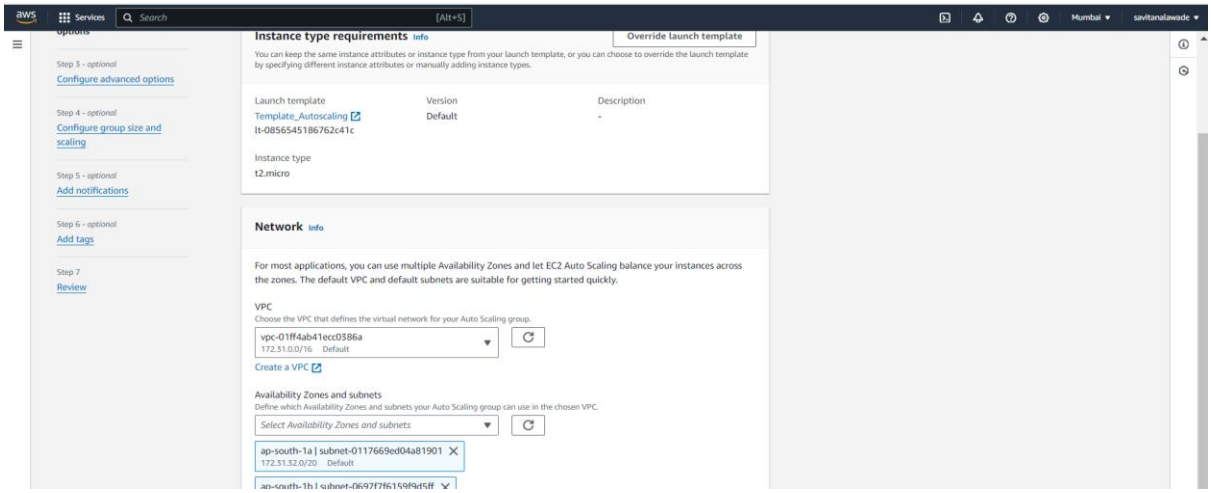


❖ Create Autoscaling Group

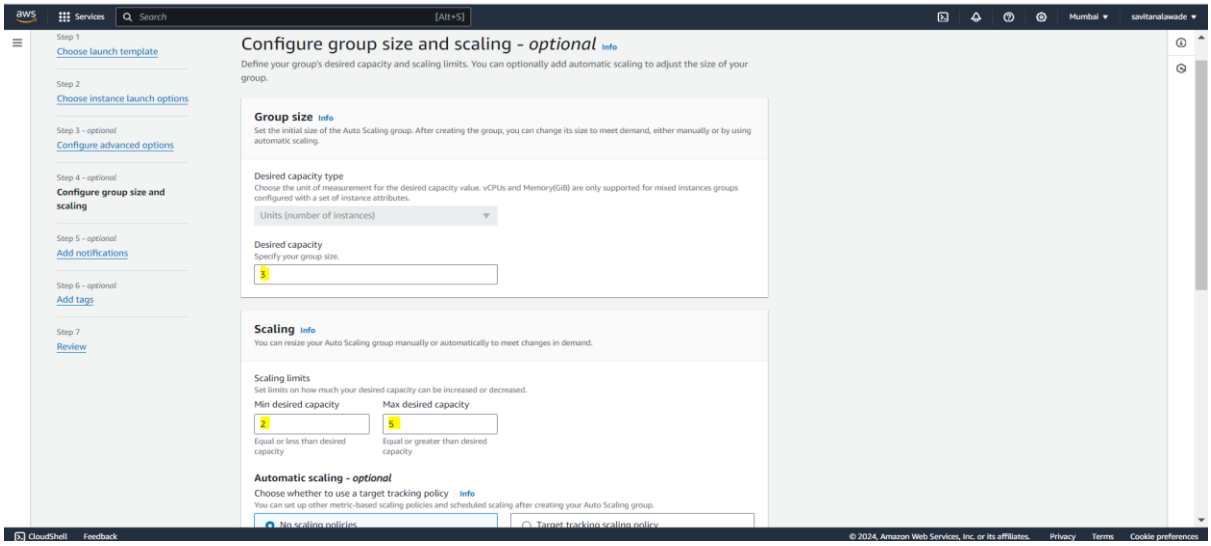
a. Provide group name and select template which is created



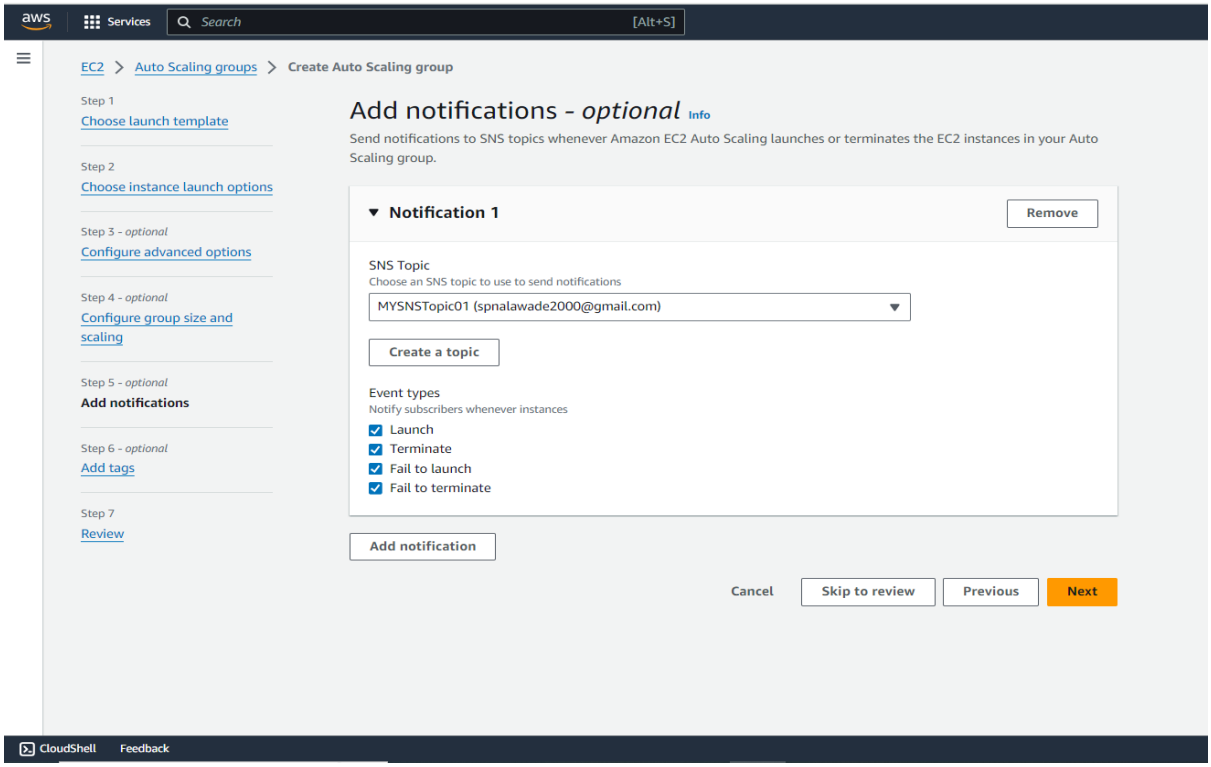
b. Select Availability Zone which you want to create instance



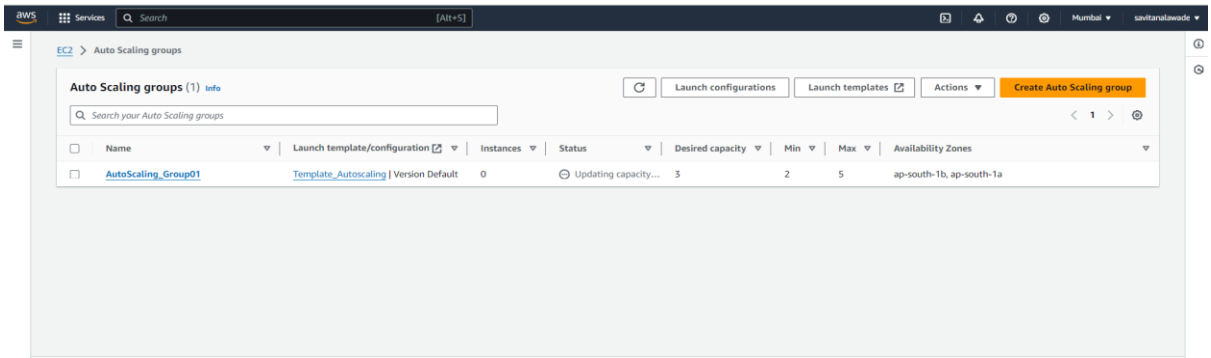
c. Configured group size and scaling



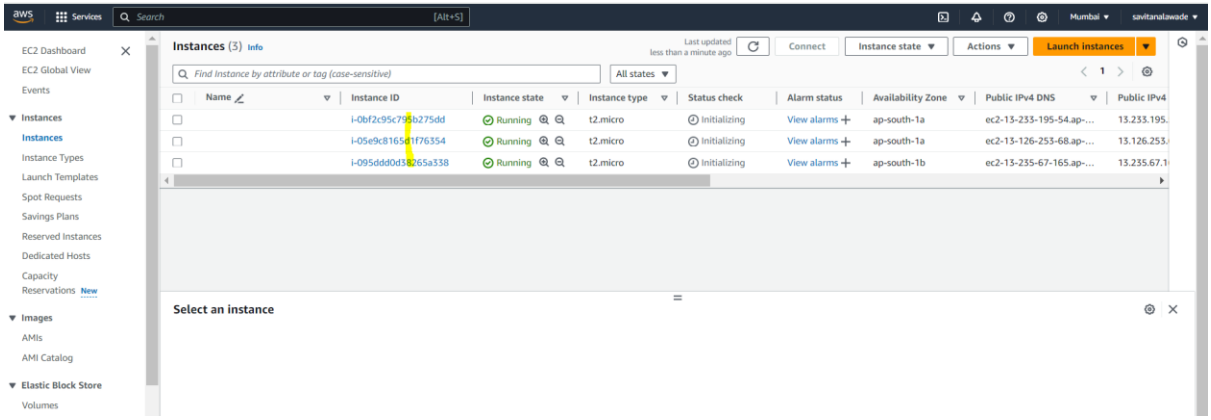
d. Add Notification



e. Auto scaling group have been created

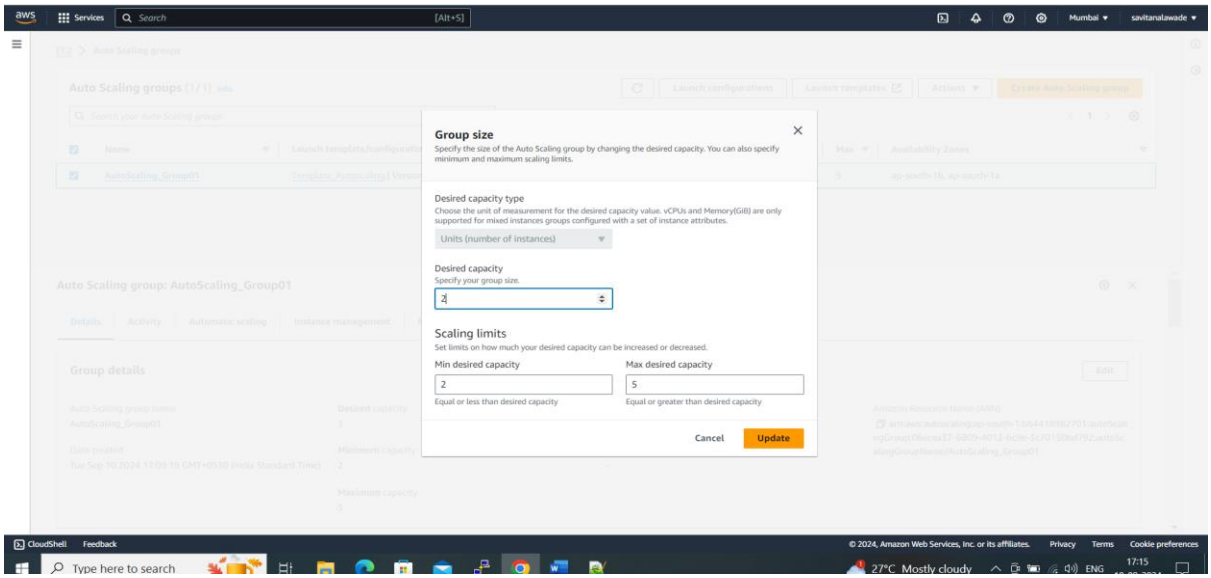


f. AutoScaling group have been created 3 Instance as we mentioned in desire “3” count

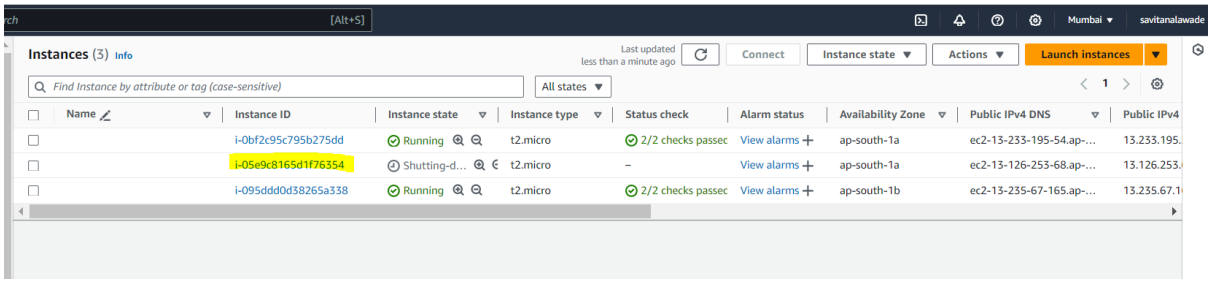


❖ Manual Scaling

If I change desired value manually as “2” so AutoScaling group should remove 1 instance.



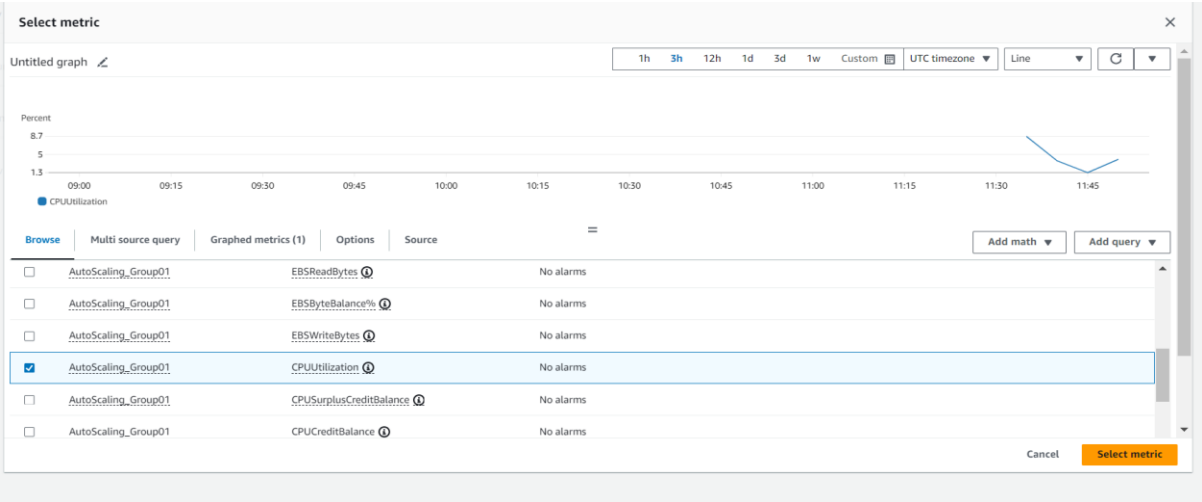
We can see one instance is going to terminate



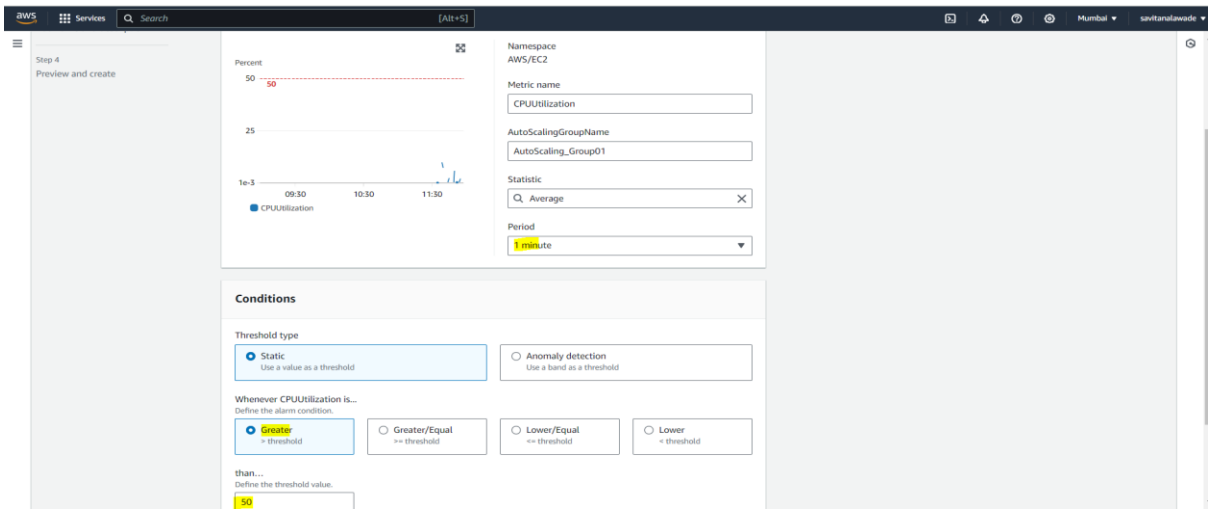
❖ Automatic Scaling

To create Automatic scaling we required alarm to set Create dynamic scaling policies.

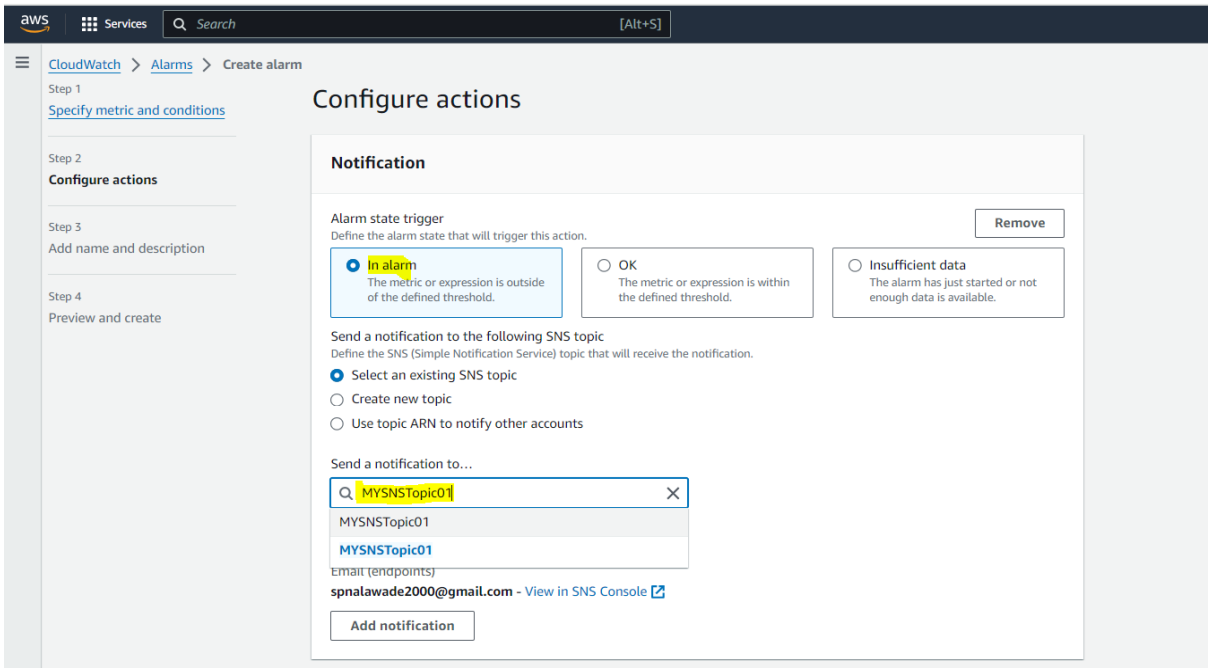
1. Create alarm for Autoscaling group
   a. Select metric(CPU-Utilization)



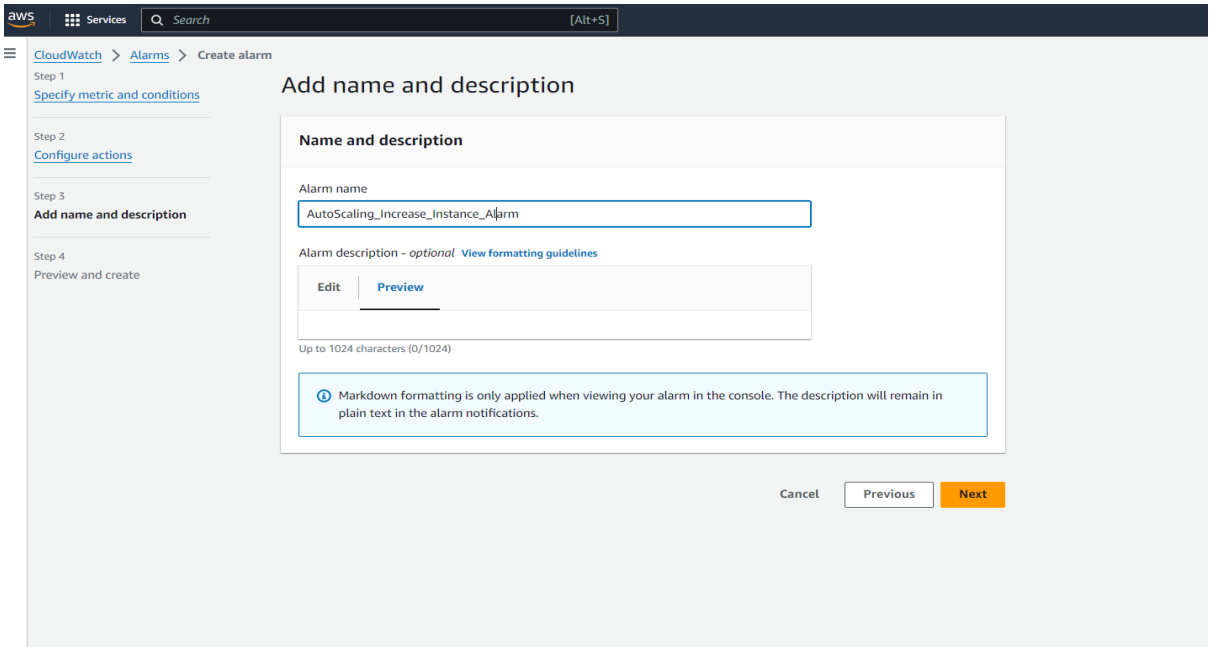
b. Select timeperiod”1 min” and threshold value(utilization>=50)



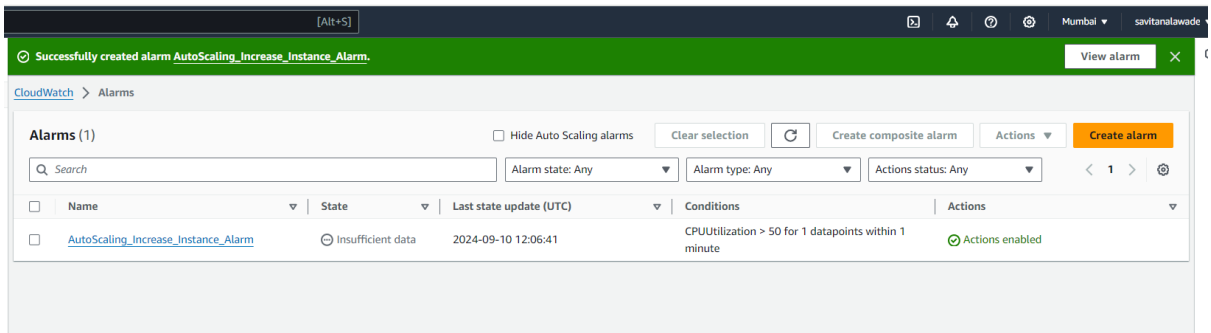
c. Select in which state you want to trigger “In alarm” and add SNS-topic



d. Add name for alarm

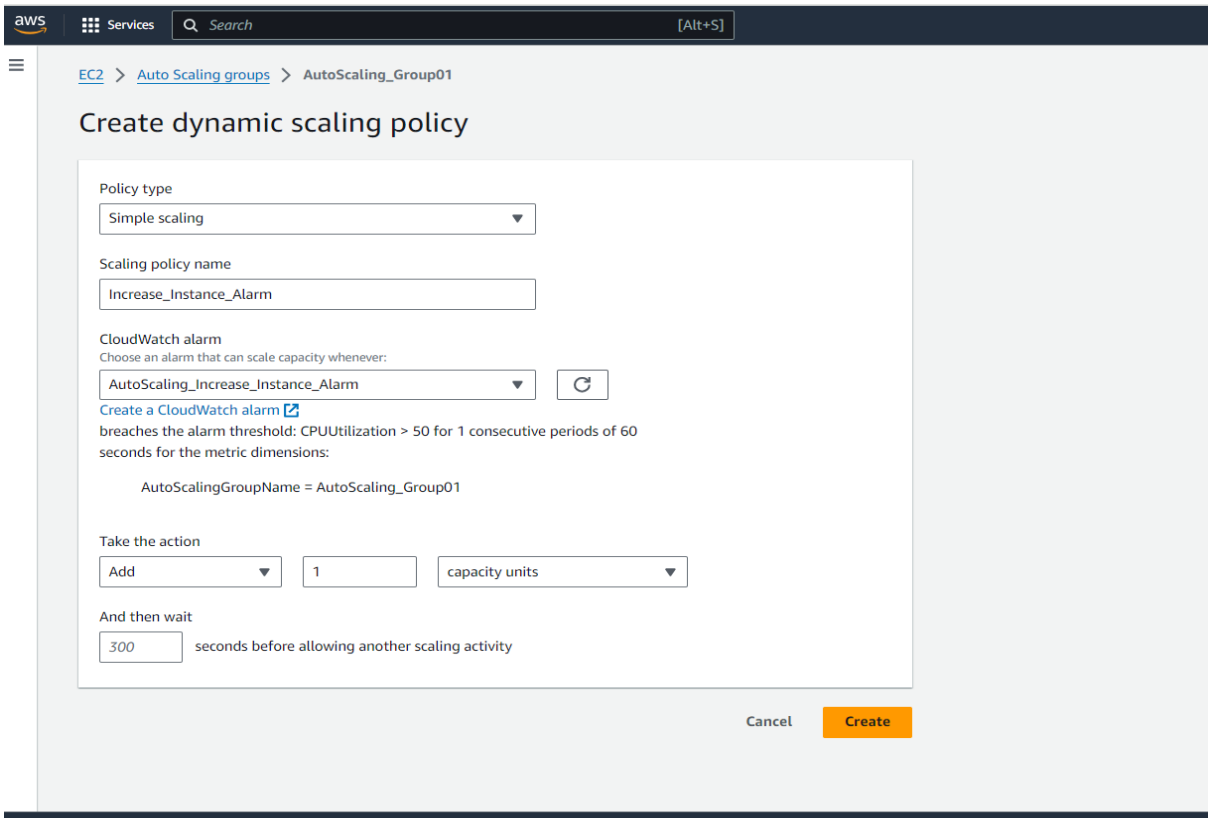


e. Alarm have been created.

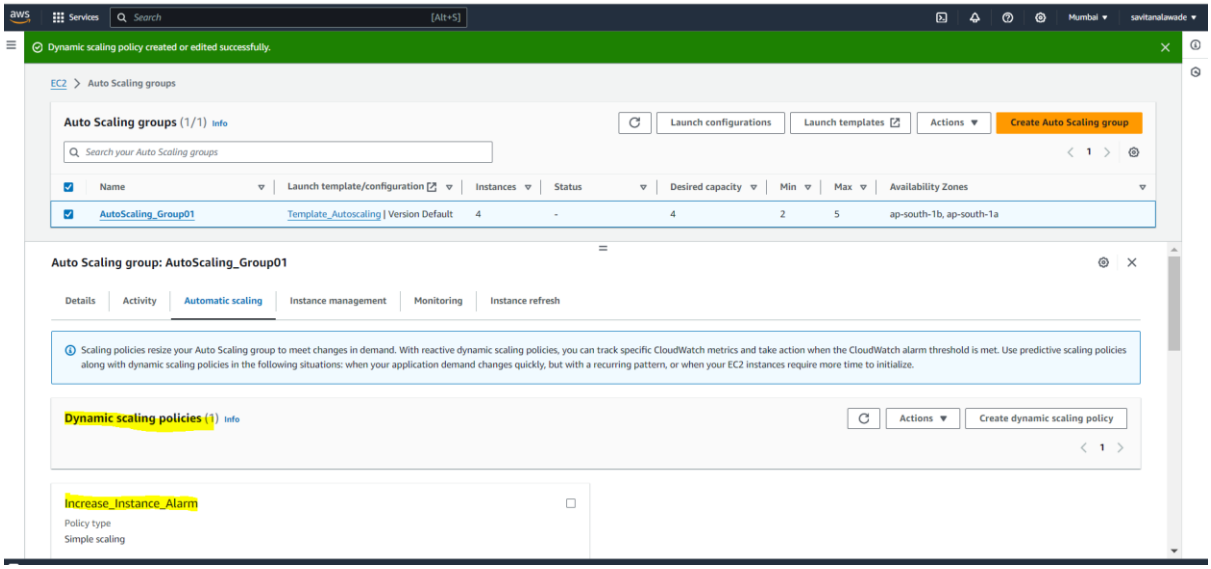


- **Create Dynamic policy** with alarm in Take Action added count as “1”

It means if utilization is goes above threshold i.e 50% it should add one more instance.

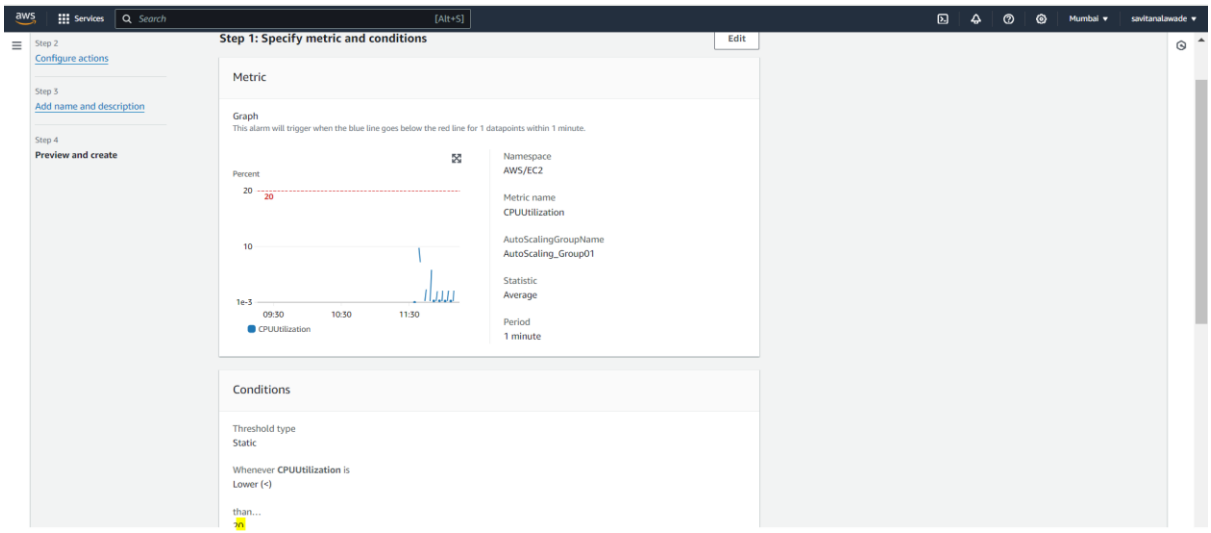


Created Dynamic policy as **"Increase_Instance_Alarm"**

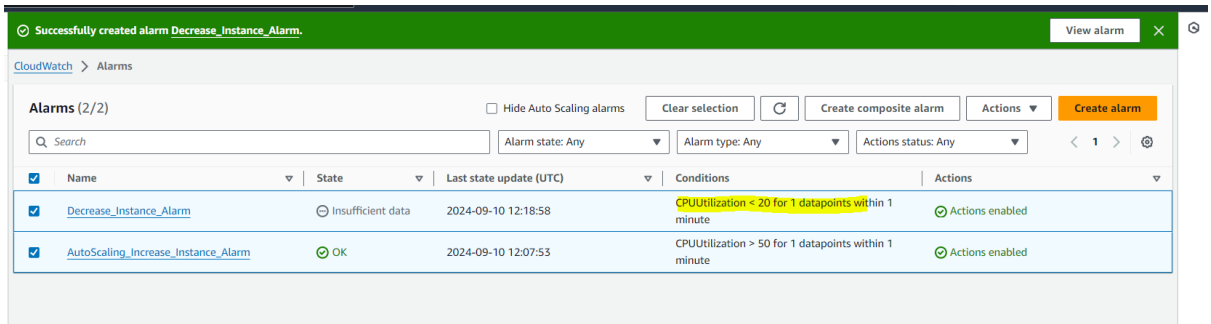


- For one more dynamic policy we need to create one more alarm as **"Decrease_Instance_Alarm"**

1.Now, condition is {if CPU utilization is below 20% it should trigger alarm}

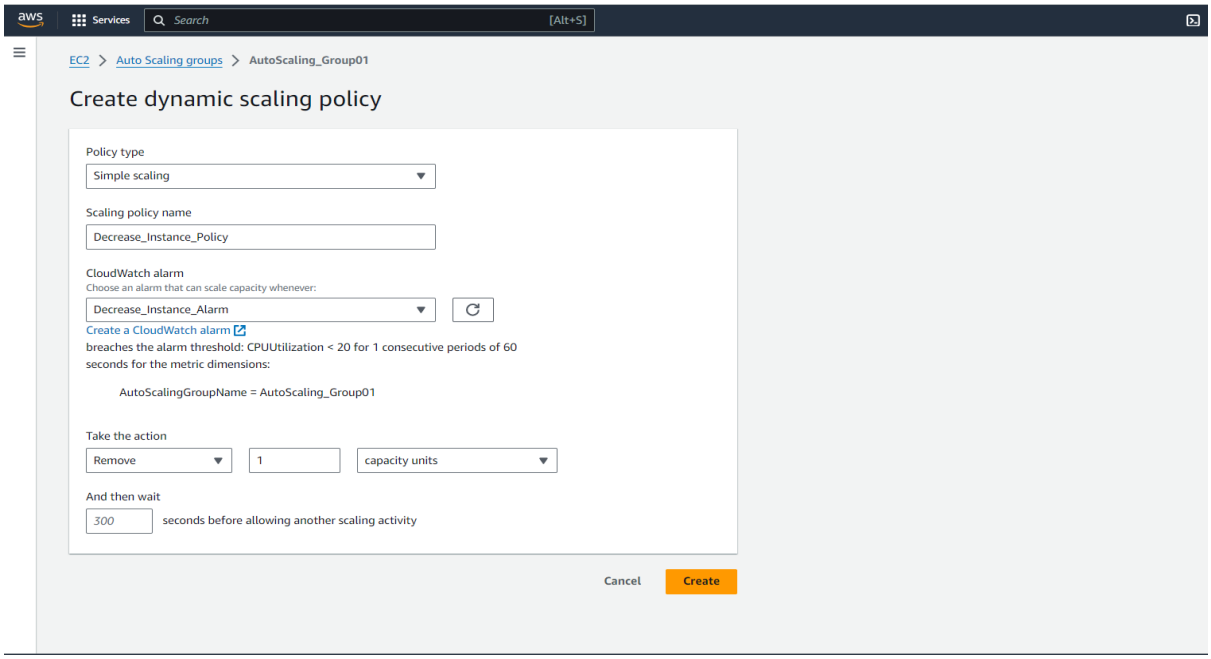


2.Alarm have been created

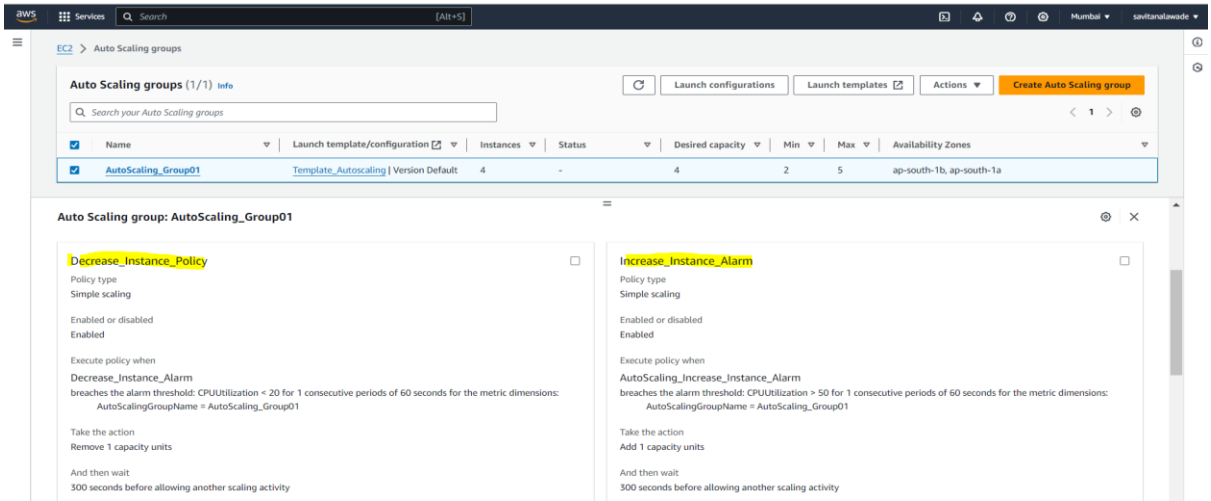


- Create policy as "**Decrease_Instance_Policy**" in Take Action select "Remove" and added count as "1"

It means if utilization its below threshold i.e 20% it should remove one instance.

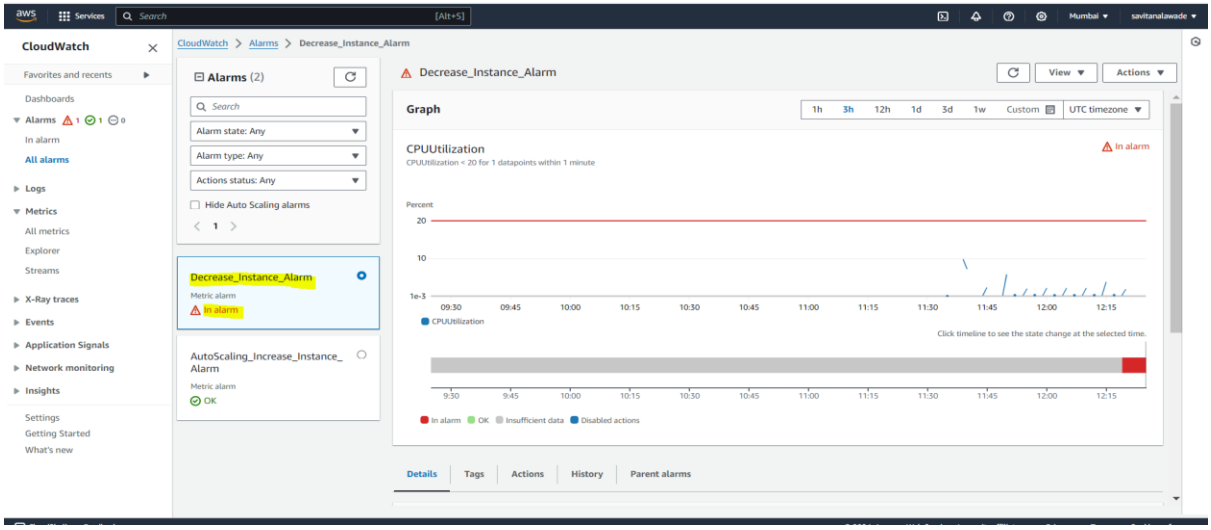


We have added two policy successfully and both are enabled



❖ Output

1. We could see one alarm has been triggered “Decrease_Instance_Alarm” As checked average cpu utilization of Instance are below 20% only .



ALARM: "Decrease_Instance_Alarm" in Asia Pacific (Mumbai) Inbox ×

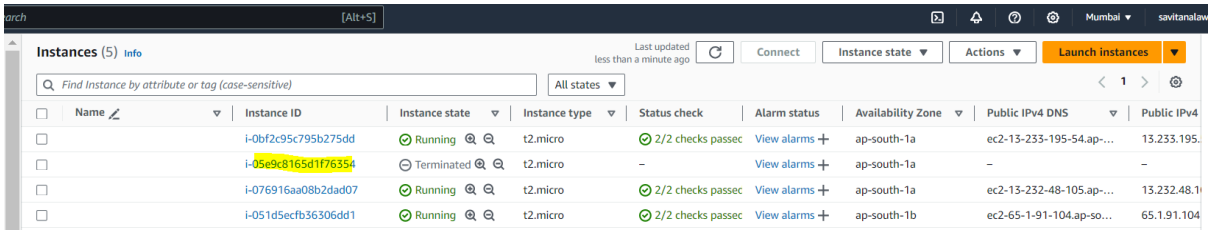
AWS Notifications <no-reply@sns.amazonaws.com> 5:49 PM (19 minutes ago)
to me

You are receiving this email because your Amazon CloudWatch Alarm "Decrease_Instance_Alarm" in the Asia Pacific (Mumbai) region has entered the because "Threshold Crossed: 1 out of the last 1 datapoints [4.538091111179893 (10/09/24 12:15:00)] was less than the threshold (20.0) (minimum 1 d -> ALARM transition)." at "Tuesday 10 September, 2024 12:19:04 UTC".

View this alarm in the AWS Management Console:
https://ap-south-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-south-1#alarmsV2:alarm/Decrease_Instance_Alarm

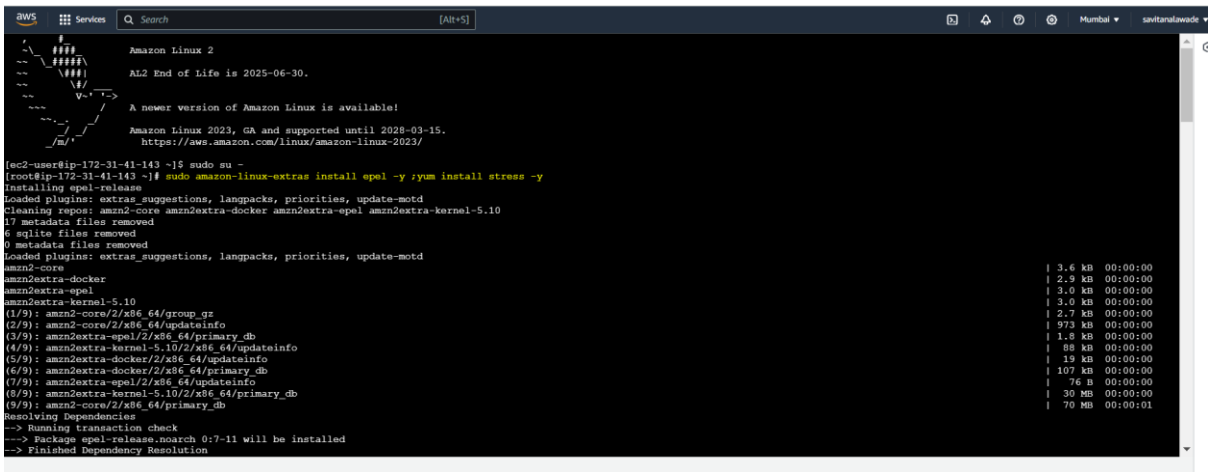
Alarm Details:
- Name: Decrease_Instance_Alarm
- Description:
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [4.538091111179893 (10/09/24 12:15:00)] was less than the threshold (1 datapoint for OK -> ALARM transition).
- Timestamp: Tuesday 10 September, 2024 12:19:04 UTC
- AWS Account: 664418982701
- Alarm Arn: arn:aws:cloudwatch:ap-south-1:664418982701:alarm:Decrease_Instance_Alarm

2. Now, It will remove one Instance out of 4 running Instance

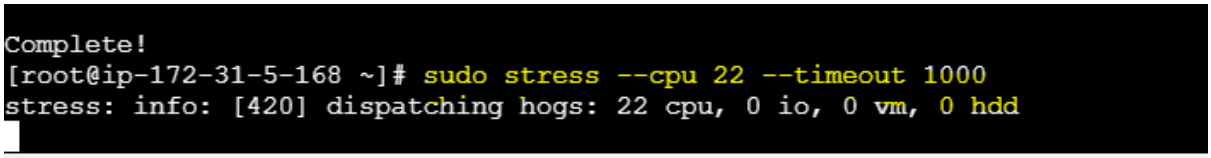


- To trigger “Increase_Instance_alarm” we need to make high cpu utilization of instances

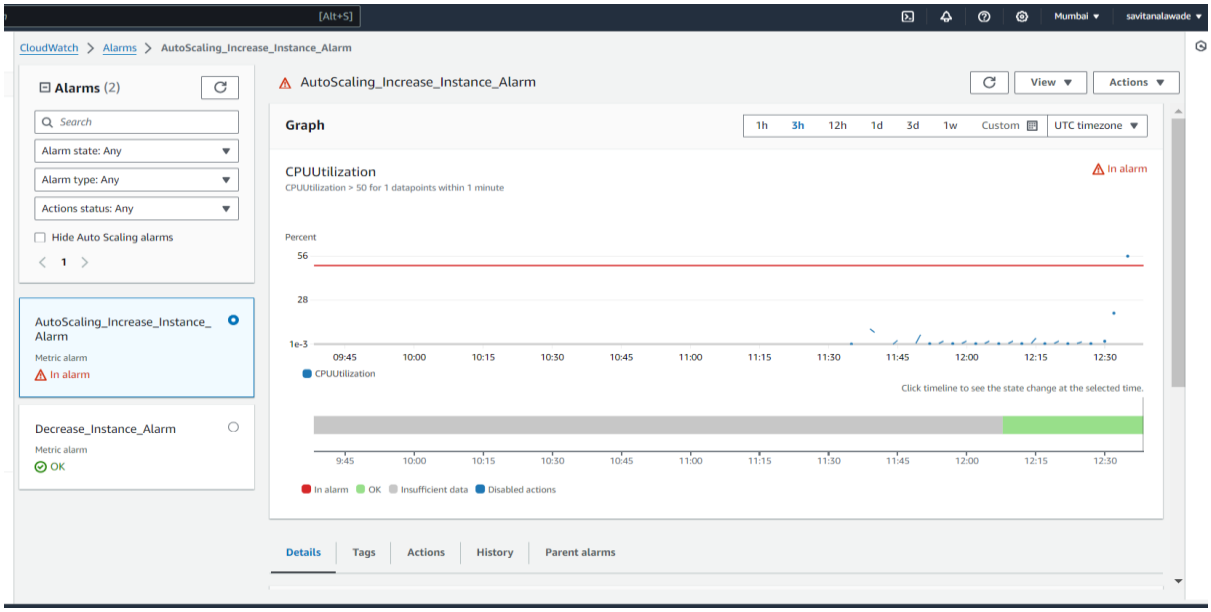
We’ll increase cpu-utilization by installing stress and made cpu changes on both instance:

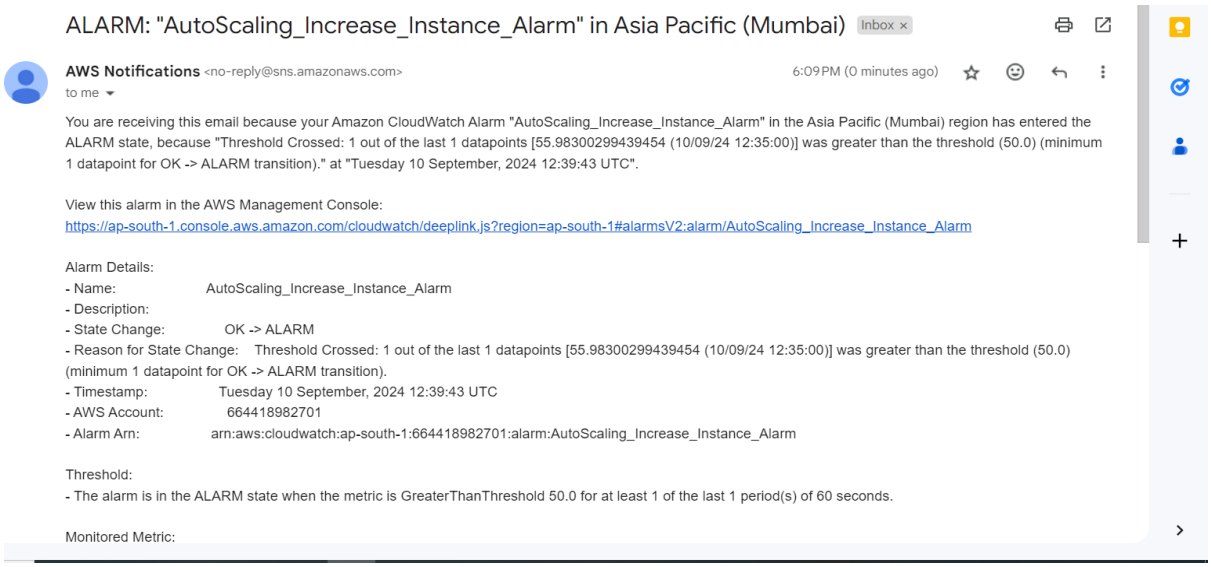


Increase cpu by stress command “sudo –cpu 22 –timeout 1000”

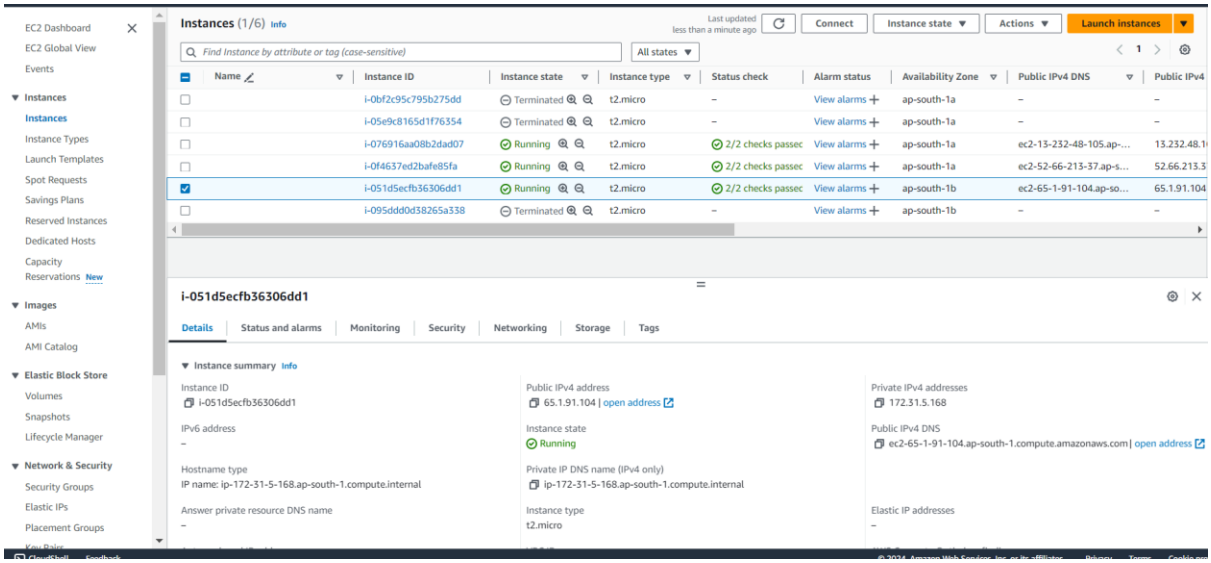


Now check cpu-utilization goes high and alarm will triggered



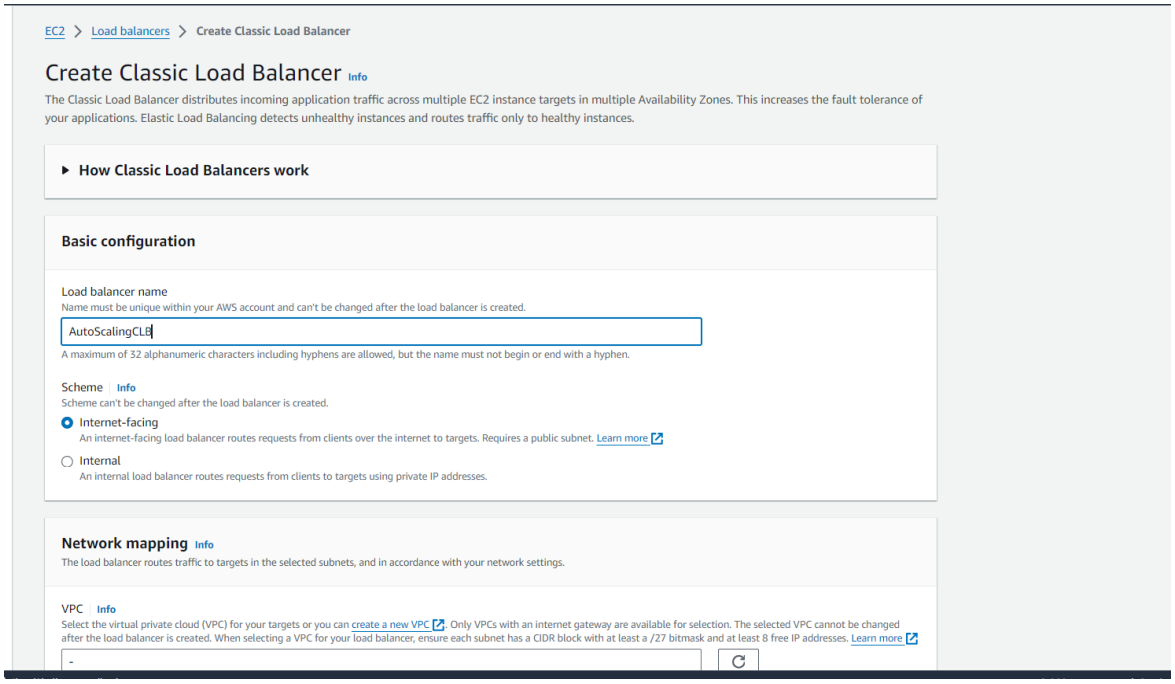


It will generate one instance

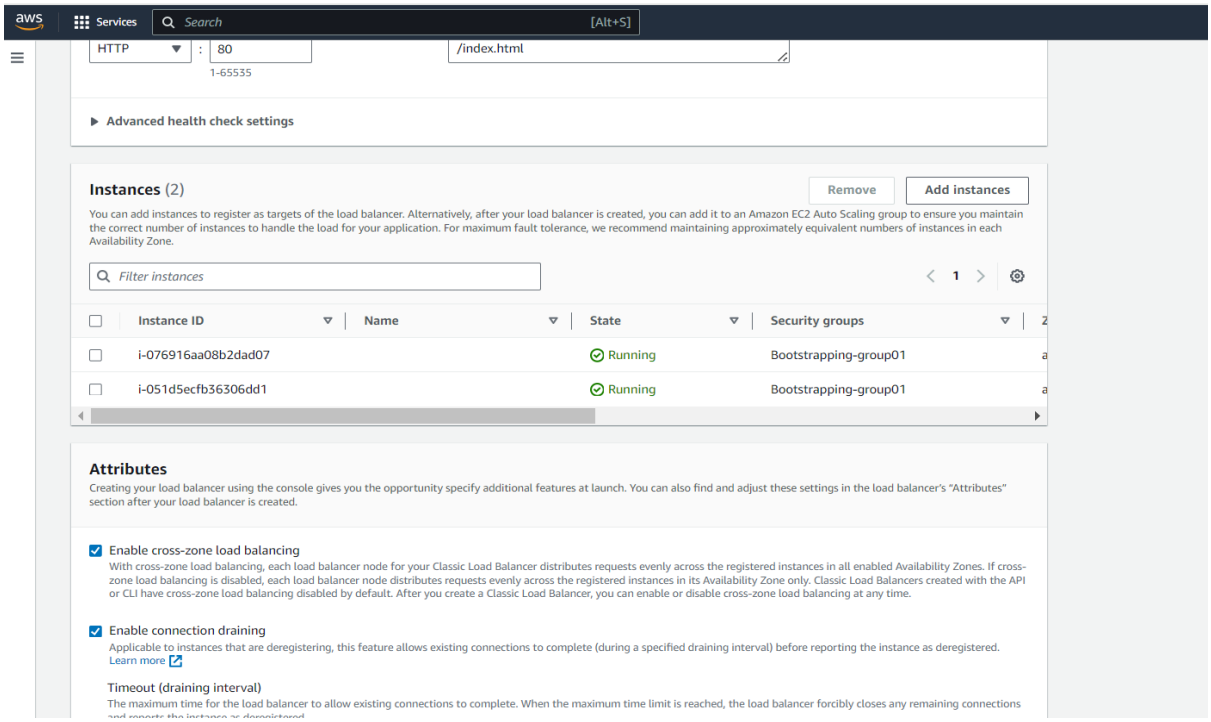


❖ Create Classic Load Balancer

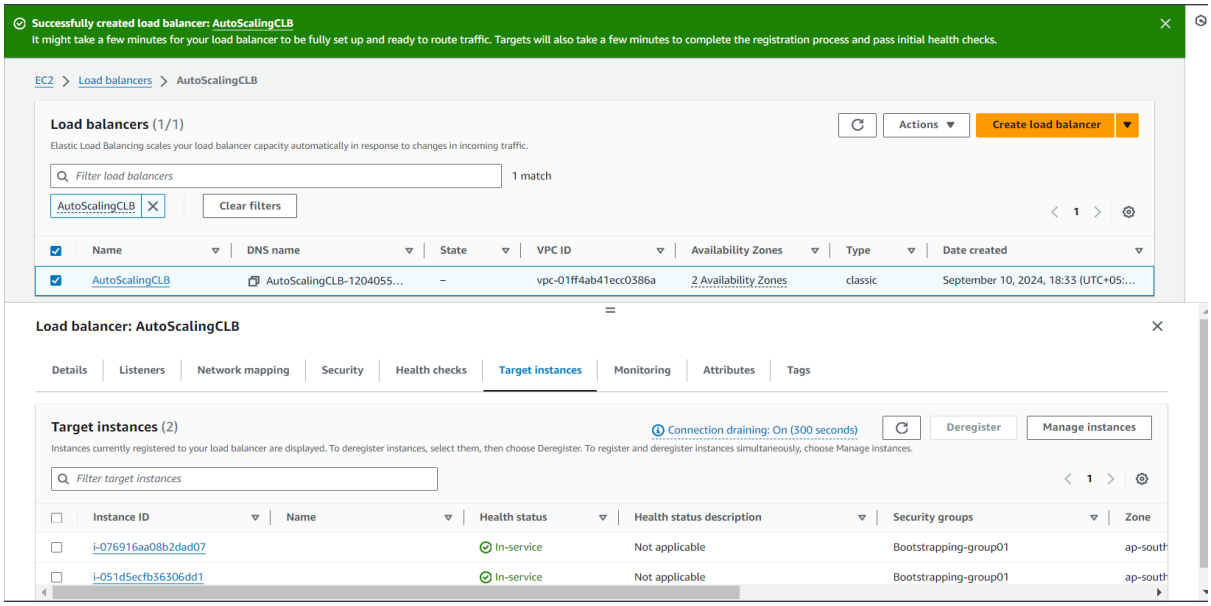
1.Create load balancer as “AutoScalingCLB”



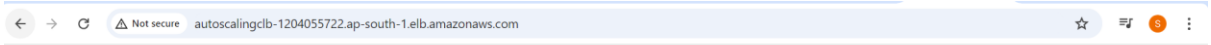
2.Select security group and add running Instances



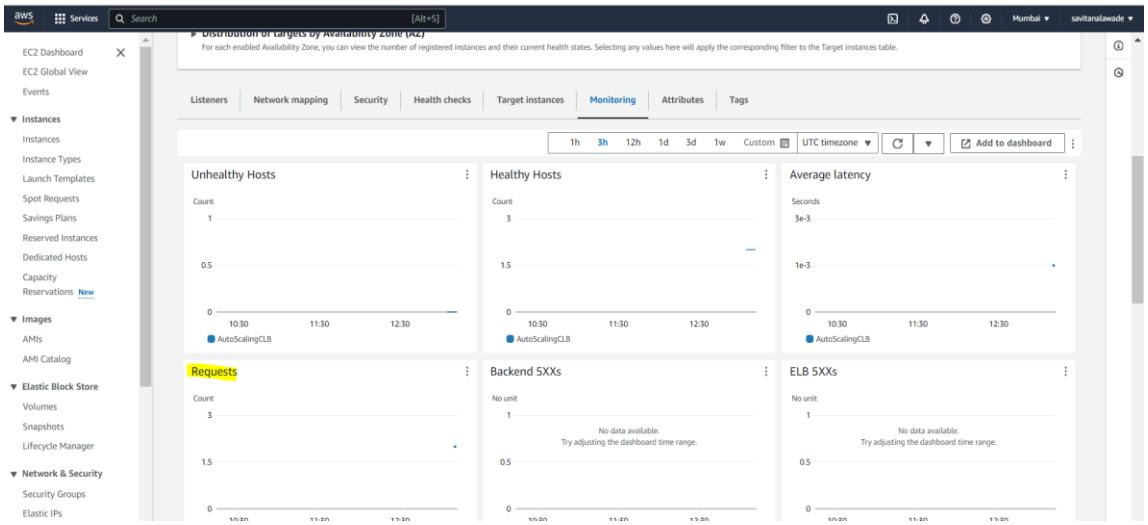
3.Load Balancer have been created and instances are in service



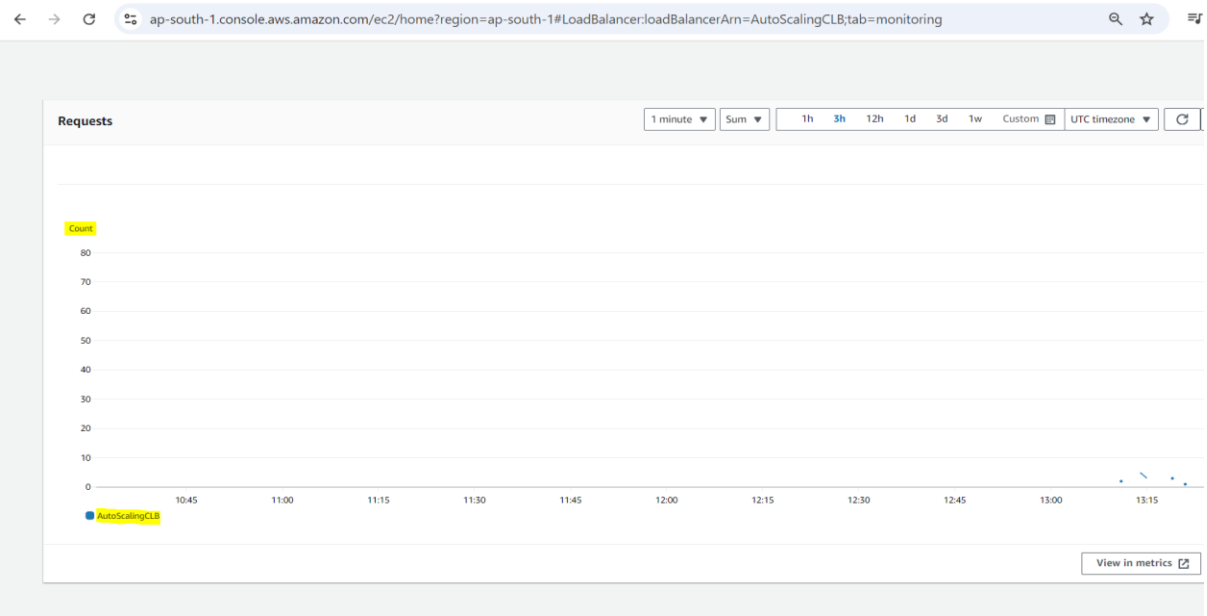
3. Using DNS of load balancer we can view application and if hit multiple times we can see in monitoring of LB



Create and configure the service front-end-service so its accessible through ClusterIP and routes to the existing pod named front-end



After hitting multiple times on browser we can see request are getting increased


ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LoadBalancer:loadBalancerArn=AutoScalingCLB;tab=monitoring
Requests
1 minute
Sum
1h
3h
12h
1d
3d
1w
Custom
UTC timezone
Count
80
70
60
50
40
30
20
10
0
10:45
11:00
11:15
11:30
11:45
12:00
12:15
12:30
12:45
13:00
13:15
AutoScalingCLB
View in metrics

Thank you…