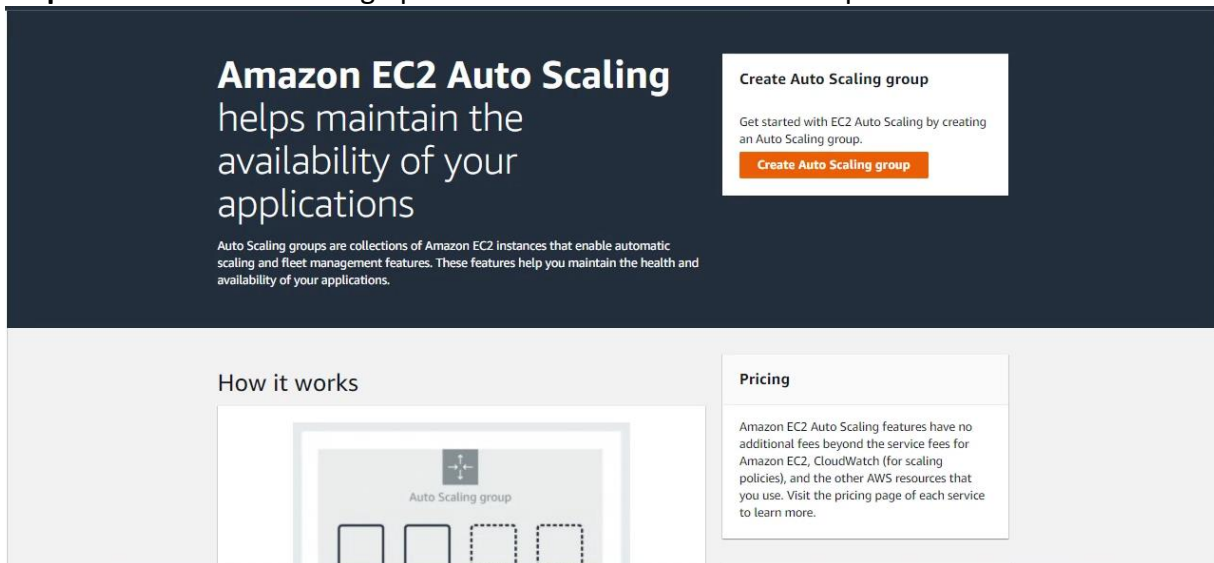


## LAB10

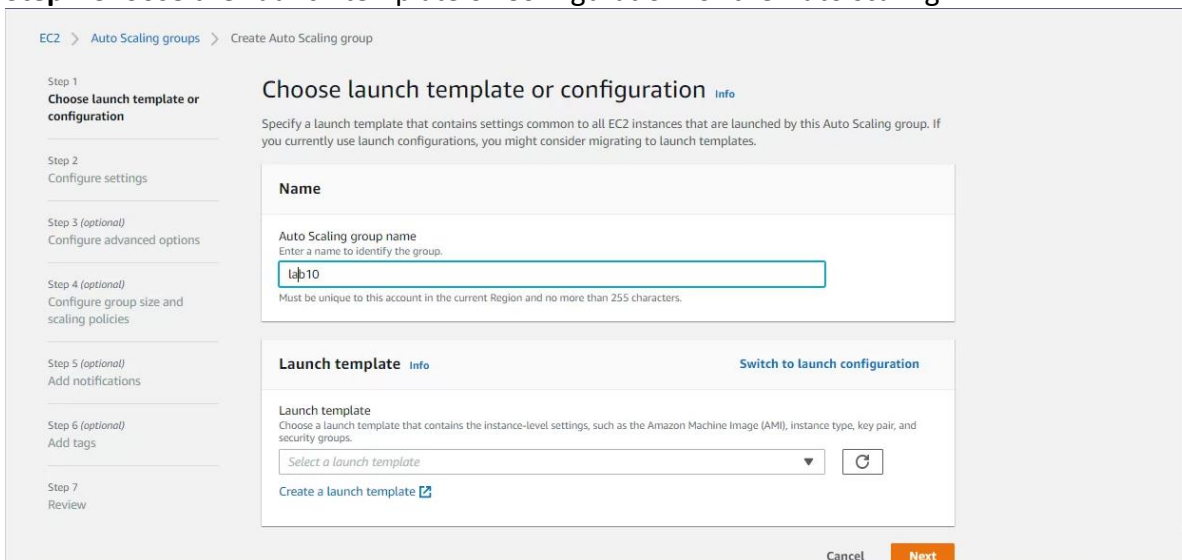
### Demonstrate auto scaling group concept in cloud.

Amazon EC2 Auto Scaling can detect when an instance is unhealthy, terminate it, and launch an instance to replace it. You can also configure Amazon EC2 Auto Scaling to use multiple Availability Zones. If one Availability Zone becomes unavailable, Amazon EC2 Auto Scaling can launch instances in another one to compensate. Amazon EC2 Auto Scaling helps ensure that your application always has the right amount of capacity to handle the current traffic demand. Amazon EC2 Auto Scaling can dynamically increase and decrease capacity as needed. Because you pay for the EC2 instances you use, you save money by launching instances when they are needed and terminating them when they aren't.

**Step1:** Select the Autoscaling option from the from AWS services panel.



**Step2:** Choose the Launch template or Configuration for the Auto scaling.



**Step3:** Before Configuring the Auto Scaling create Launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

### Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

Amazon machine image (AMI) - required [Info](#)

AMI - required

Amazon Linux 2 AMI (HVM), SSD Volume Type  
ami-0a36eb8fad976275  
Catalog: Quick Start    virtualization: hvm    architecture: 64-bit (x86)

Instance type [Info](#)

Instance type

t1.micro  
Family: t1    1 vCPU    0.612 GiB Memory  
On-Demand Linux pricing: 0.02 USD per Hour  
On-Demand Windows pricing: 0.02 USD per Hour

Free tier eligible

[Instance types](#)

Key pair (login) [Info](#)

Key pair name

Don't include in launch template

[Create new key pair](#)

Network settings

EC2 Experience

Tell us what you think

EC2 Dashboard

Events

Tags

Limits

INSTANCES

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Scheduled Instances

Capacity Reservations

IMAGES

AMIs

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Step 1

Choose launch template or configuration

Step 2

Configure settings

Step 3 (optional)

Configure advanced options

Step 4 (optional)

Configure group size and scaling policies

Step 5 (optional)

Add notifications

Step 6 (optional)

Add tags

Step 7

Review

Choose launch template or configuration

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Name

Auto Scaling group name

Enter a name to identify the group.

lab10

Must be unique to this account in the current region and no more than 255 characters.

Launch template

Switch to launch configuration

Launch template

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

lab10

Create a launch template

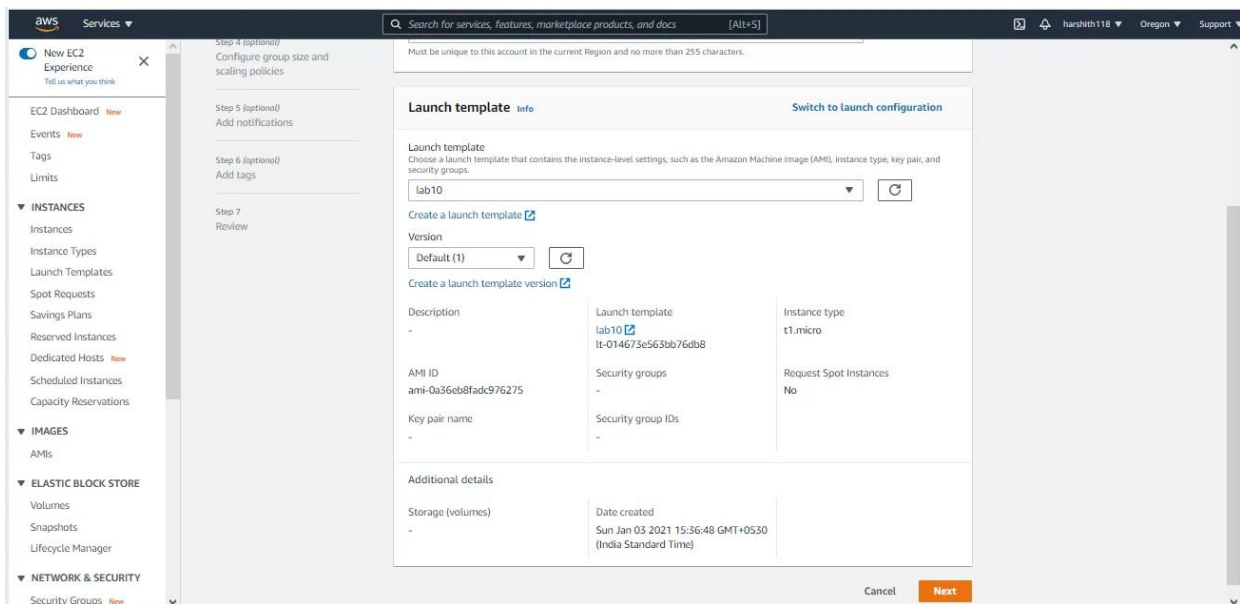
Version

Default (1)

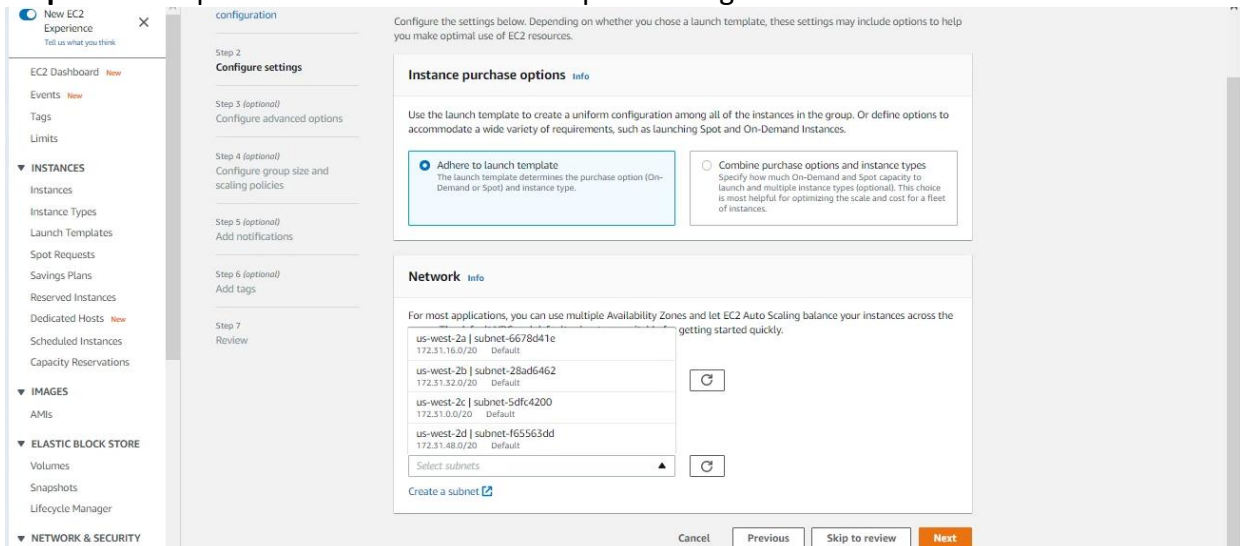
Create a launch template version

Description	Launch template	Instance type
-	lab10 lt-014675e563bb76db8	t1.micro
AMI ID	Security groups	Request Spot Instances

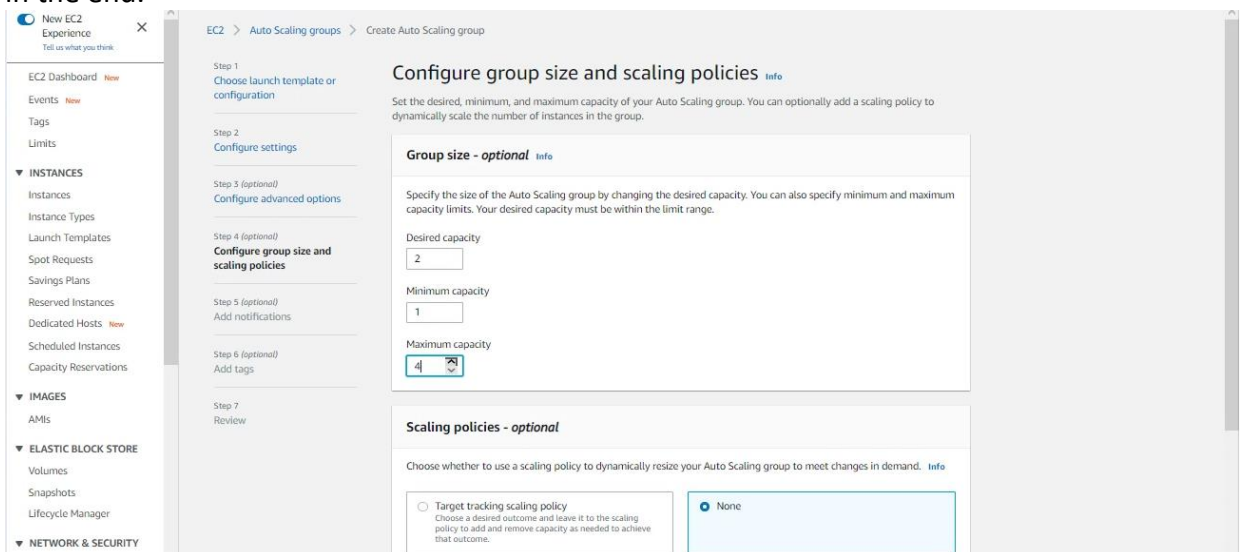
**Step6:**Now add the Additional information and click on Next.



**Step7:** Select vpc and subnets in Launch template configuration.



**Step8:**Now Configure group size and scaling policies as requirement.And create auto scaling group in the end.



**Step9:**Now if we check instances 2 instances will be created automatically.

Welcome to the new Instances experience! We're redesigning the EC2 console to make it easier to use. To switch between the old console and the new console, use the New EC2 Experience toggle above the navigation panel. We'll release updates continuously based on customer feedback.

**Instances (2)** Info

Filter instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input type="checkbox"/>	-	i-0fd74e120327a4233	Running	t1.micro	Initializing	No alarms +	us-west-2b	ec2-34-222-7-24.us-w...	34.222.7.24
<input type="checkbox"/>	-	i-0a11dd0d1ac349f2e	Running	t1.micro	Initializing	No alarms +	us-west-2c	ec2-54-69-146-91.us-...	54.69.146.91

Select an instance above

**Step10:** We can also check the activity history in auto scaling groups panel.

Send to On instance action

No notifications are currently specified

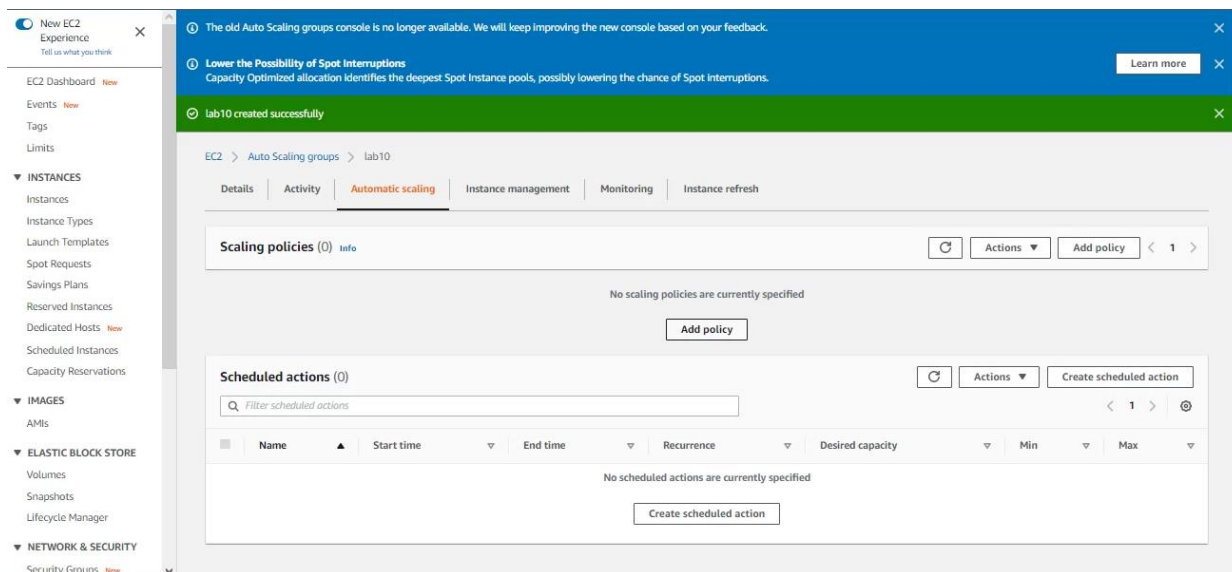
Create notification

**Activity history (3)**

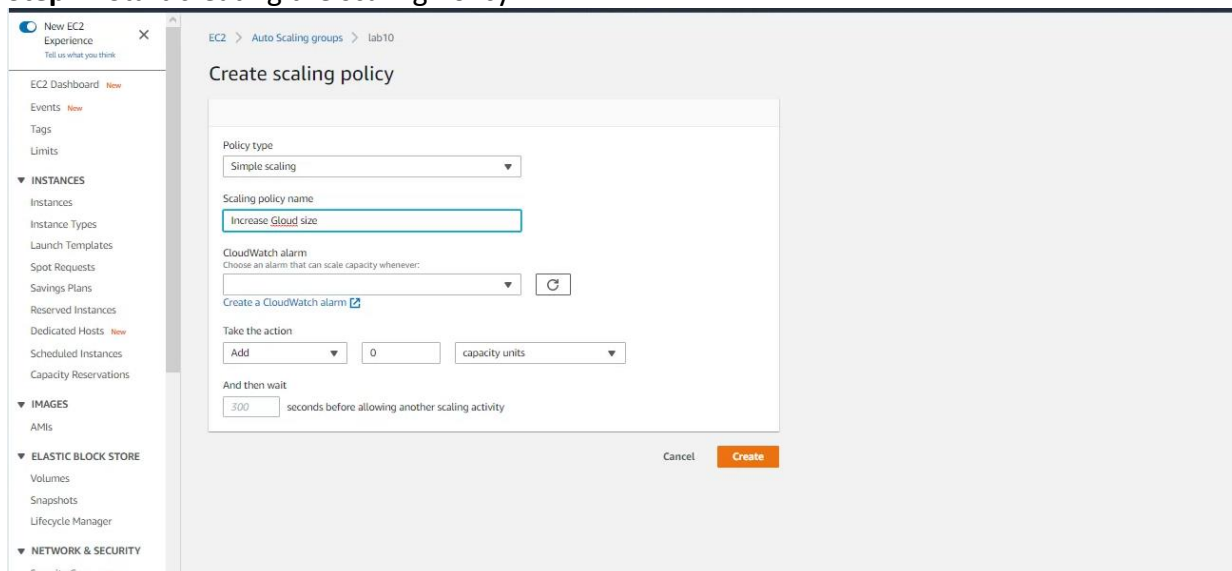
Filter activity history

Status	Description	Cause	Start time	End time
Successful	Launching a new EC2 instance: i-0fd74e120327a4233	At 2021-01-03T10:12:30Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 1 to 2.	2021 January 03, 03:42:32 PM +05:30	2021 January 03, 03:43:04 PM +05:30
Successful	Launching a new EC2 instance: i-0a11dd0d1ac349f2e	At 2021-01-03T10:12:04Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2021-01-03T10:12:10Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 January 03, 03:42:12 PM +05:30	2021 January 03, 03:42:44 PM +05:30
Failed	Launching a new EC2 Instance. Status Reason: Your requested instance type (t1.micro) is not supported in your requested Availability Zone (us-west-2d). Please retry your request by not specifying an Availability Zone or choosing us-west-2a, us-west-2b, us-west-2c.	At 2021-01-03T10:12:04Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2021-01-03T10:12:10Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 January 03, 03:42:11 PM +05:30	2021 January 03, 03:42:11 PM +05:30

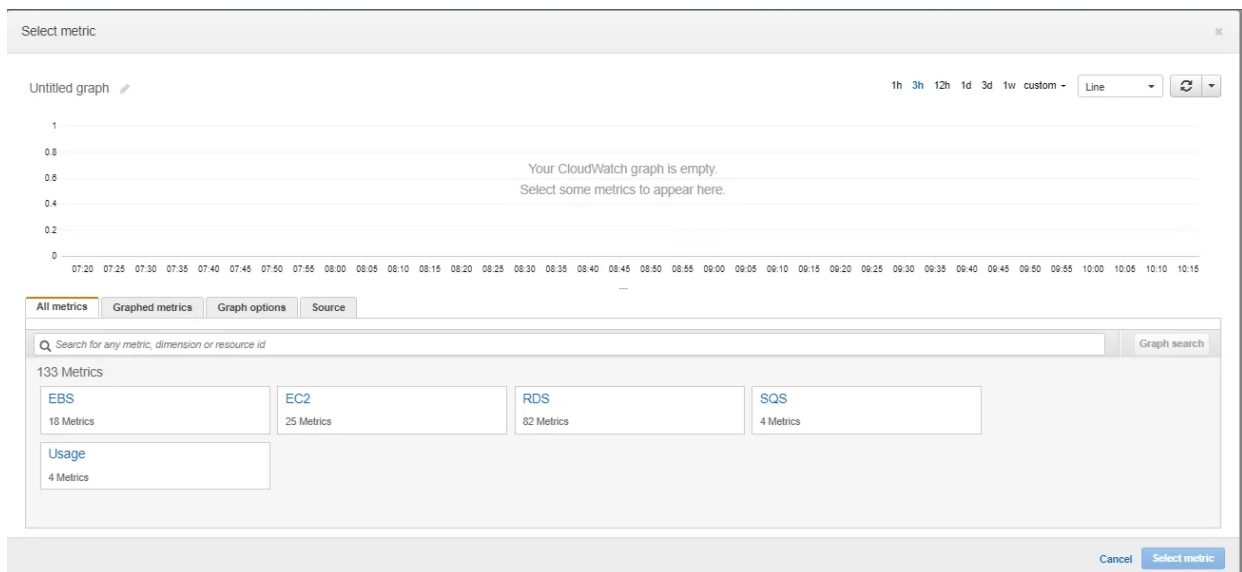
**Step11:** Now go to auto scaling group and select Add Policy.



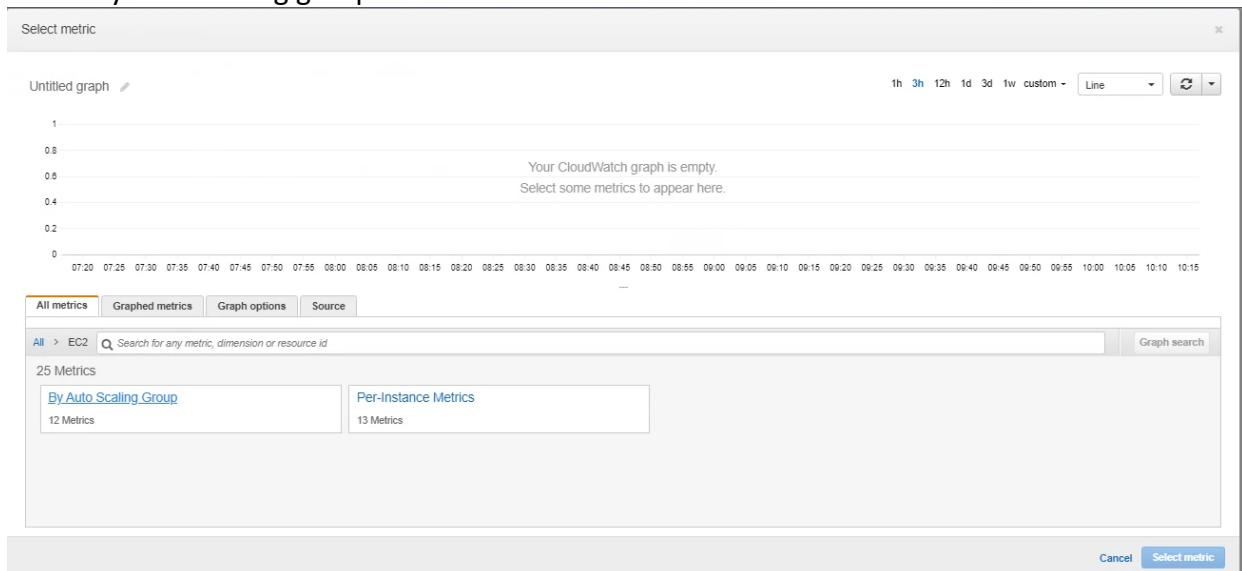
## Step12: Start creating the scaling Policy



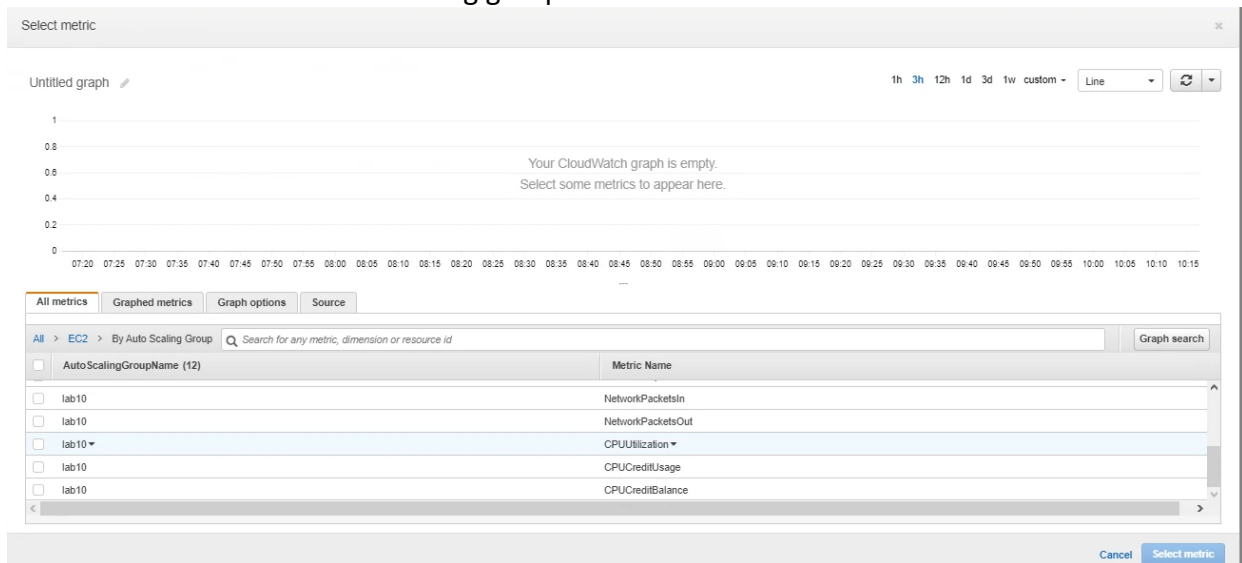
**Step13:** Before creating the scaling policy CloudWatch alarm by selecting Create a cloudWatch alarm. In new tab Specify Metric and Conditions click on select metric. Inside metric select EC2.



## Select By Auto scaling group in EC2



## Select CPU utilization in Auto scaling group



**Step14:**Enter the conditions as threshold type = static,whenever CPUUtilization is Greater,Than = 60 sec.click next.

0.1  
0.05  
0  
07:30 08:30 09:30  
CPUUtilization

AutoScalingGroupName  
lab10

Statistic  
Average

Period  
5 minutes

**Conditions**

Threshold type  
☒ Static  
Use a value as a threshold  
☐ Anomaly detection  
Use a band as a threshold

Whenever CPUUtilization is...  
Define the alarm condition.  
☒ Greater  
> threshold  
☐ Greater/Equal  
≥ threshold  
☐ Lower/Equal  
≤ threshold  
☐ Lower  
< threshold

than...  
Define the threshold value.  
64  
Must be a number

► Additional configuration

Cancel Next

**Step15:**In configure actions select an SNS topic as create new topic as IncreaseGroupSize, enter Email to notify And create topic.

Step 3  
Add name and description

Step 4  
Preview and create

Alarm state trigger  
Define the alarm state that will trigger this action.  
☒ In alarm  
The metric or expression is outside of the defined threshold.  
☐ OK  
The metric or expression is within the defined threshold.  
☐ Insufficient data  
The alarm has just started or not enough data is available.

Select an SNS topic  
Define the SNS (Simple Notification Service) topic that will receive the notification.  
☐ Select an existing SNS topic  
☒ Create new topic  
☐ Use topic ARN

Create a new topic...  
The topic name must be unique.  
IncreaseGroupSize  
SNS topic names can contain only alphanumeric characters, hyphens (-) and underscores (\_).

Email endpoints that will receive the notification...  
Add a comma-separated list of email addresses. Each address will be added as a subscription to the topic above.  
user@example.com  
user1@example.com, user2@example.com

Create topic

Add notification

**Auto Scaling action**

Add Auto Scaling action

Add the name,description and click next

CloudWatch > Alarms > Create alarm

Step 1  
Specify metric and conditions

Step 2  
Configure actions

Step 3  
Add name and description

Step 4  
Preview and create

**Add name and description**

Name and description

Alarm name  
IncreaseGroupSize

Alarm description - optional  
Alarm description

Up to 1024 characters (0/1024)

Cancel Previous Next

Preview and create the alarm.



**Step16:**Now add one more policy with Name DecreaseGroupSize by repeating the same Steps.in conditons just change whenever CPUUtilization is Lowest/equal.

The screenshot shows the 'Create Alarm' wizard in AWS CloudWatch, specifically the 'Conditions' step. The 'Threshold type' is set to 'Static'. Under 'Whenever CPUUtilization is...', the 'Lower/Equal' condition is selected. The threshold value is set to 60. The 'Metric name' is 'CPUUtilization' and the 'AutoScalingGroupName' is 'lab10'. The 'Statistic' is 'Average' and the 'Period' is '5 minutes'.

**Step17:**Now if we check Alarms there will be two alarms as per our creation.

The screenshot shows the 'Alarms' page in AWS CloudWatch. There are two alarms listed:

Name	State	Last state update	Conditions	Actions
Decrease Group Size	Insufficient data	2021-01-03 15:55:13	CPUUtilization <= 60 for 1 datapoints within 5 minutes	Pending confirmation
IncreaseGroupSize	OK	2021-01-03 15:52:18	CPUUtilization > 60 for 1 datapoints within 5 minutes	Pending confirmation

**Step18:**Now continue with creation of scaling policy by selecting DecreaseGroupSize alarm.



EC2 > Auto Scaling groups > lab10

### Create scaling policy

Policy type: Simple scaling

Scaling policy name: DecreaseGroupSize

CloudWatch alarm: Choose an alarm that can scale capacity whenever:  
Decrease Group Size

Create a CloudWatch alarm [Create a CloudWatch alarm](#)  
breaches the alarm threshold: CPUUtilization <= 60 for 1 consecutive periods of 300 seconds for the metric dimensions:  
AutoScalingGroupName = lab10

Take the action: Remove 1 capacity units

And then wait: 60 seconds before allowing another scaling activity

Now there are two scaling policies one is DecreaseGroupSize and another one id IncreaseGroupSize

EC2 > Auto Scaling groups > lab10

The old Auto Scaling groups console is no longer available. We will keep improving the new console based on your feedback.

Details | Activity | **Automatic scaling** | Instance management | Monitoring | Instance refresh

Scaling policy created or edited successfully

Scaling policies (2) [Info](#)    < 1 >

**DecreaseGroupSize**

Policy type: Simple scaling

Enabled or disabled? Enabled

Execute policy when:  
Decrease Group Size  
breaches the alarm threshold: CPUUtilization <= 60 for 1 consecutive periods of 300 seconds for the metric dimensions:  
AutoScalingGroupName = lab10

Take the action:  
Remove 1 capacity units

And then wait:  
60 seconds before allowing another scaling activity

**Increase Group size**

Policy type: Simple scaling

Enabled or disabled? Enabled

Execute policy when:  
Increase GroupSize  
breaches the alarm threshold: CPUUtilization > 60 for 1 consecutive periods of 300 seconds for the metric dimensions:  
AutoScalingGroupName = lab10

Take the action:  
Add 1 capacity units

And then wait:  
60 seconds before allowing another scaling activity

**Step19:**Now select any one Scaling policy and execute it.

EC2 > Auto Scaling groups > lab10

Scaling policy created or edited successfully

Scaling policies (1/2) [Info](#)    < 1 >

**DecreaseGroupSize**

Policy type: Simple scaling

Enabled or disabled? Enabled

Execute policy when:  
Decrease Group Size  
breaches the alarm threshold: CPUUtilization <= 60 for 1 consecutive periods of 300 seconds for the metric dimensions:  
AutoScalingGroupName = lab10

Take the action:  
Remove 1 capacity units

And then wait:  
60 seconds before allowing another scaling activity

**Increase Group size**

Policy type: Simple scaling

Enabled or disabled? Enabled

Execute policy when:  
Increase GroupSize  
breaches the alarm threshold: CPUUtilization > 60 for 1 consecutive periods of 300 seconds for the metric dimensions:  
AutoScalingGroupName = lab10

Take the action:  
Add 1 capacity units

And then wait:  
60 seconds before allowing another scaling activity

Scheduled actions (0)

Once the IncreaseCloudSize scaling policy executes it will create new instances automatically. and when this new instance creation exceeds alarm decrease group size will be triggered and instance will be deleted.

New EC2 Experience

EC2 Dashboard

Events

Tags

Limits

INSTANCES

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Scheduled Instances

Capacity Reservations

IMAGES

AMIs

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Activity notifications (0)

Filter notifications

Send to

On instance action

No notifications are currently specified

Create notification

Activity history (4)

Filter activity history

Status	Description	Cause	Start time	End time
InProgress	Terminating EC2 instance: i-0a11dd0d1ac349f2e	At 2021-01-03T10:26:34Z a user request executed policy Increase Group size changing the desired capacity from 2 to 3. At 2021-01-03T10:26:41Z a monitor alarm Decrease Group Size in state ALARM triggered policy DecreaseGroupSize changing the desired capacity from 3 to 1. At 2021-01-03T10:26:58Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 2 to 1. At 2021-01-03T10:26:58Z instance i-0a11dd0d1ac349f2e was selected for termination.	2021 January 03, 03:56:58 PM +05:30	
Successful	Launching a new EC2 instance: i-0fd74e120327a4233	At 2021-01-03T10:12:30Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 1 to 2.	2021 January 03, 03:42:32 PM +05:30	2021 January 03, 03:43:04 PM +05:30
Successful	Launching a new EC2 instance: i-0a11dd0d1ac349f2e	At 2021-01-03T10:12:04Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2021-01-03T10:12:10Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 January 03, 03:42:12 PM +05:30	2021 January 03, 03:42:44 PM +05:30
	Launching a new EC2 instance. Status Reason: Your requested Instance type (t1.micro) is not			