SNS,CloudWatch and SES

Trainee Name : Chhavi Sharma
Newers ID : 4023
College : UPES

1.Monitor Your Estimated Charges Using CloudWatch

Step 1: Enable Billing Alerts

Step 2: Create a Billing Alarm

Step 3: Check the Alarm Status

Step 4: Create & Subscribe to SNS Topic

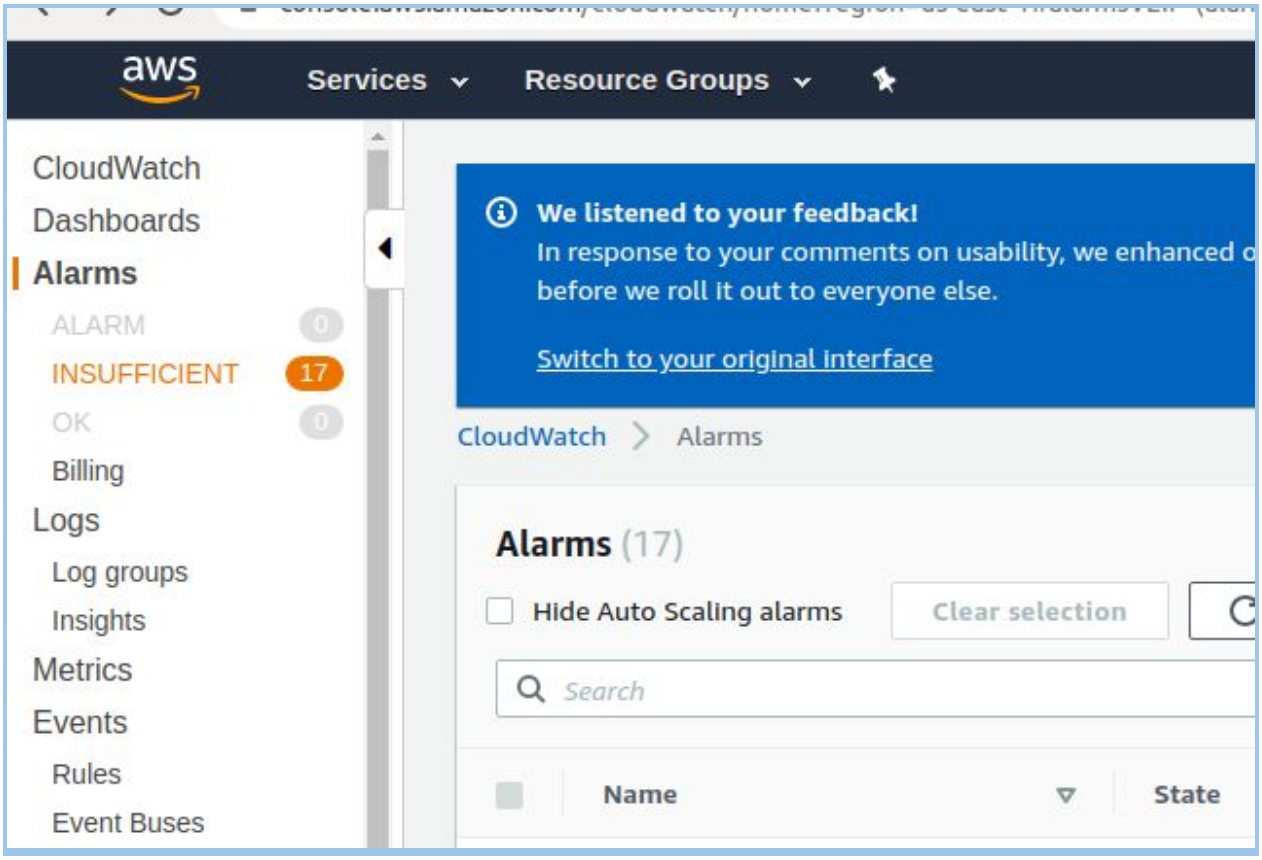
Step 5: Send a notification to all the stakeholder, if AWS resource pricing reaches the threshold value.

Ans.

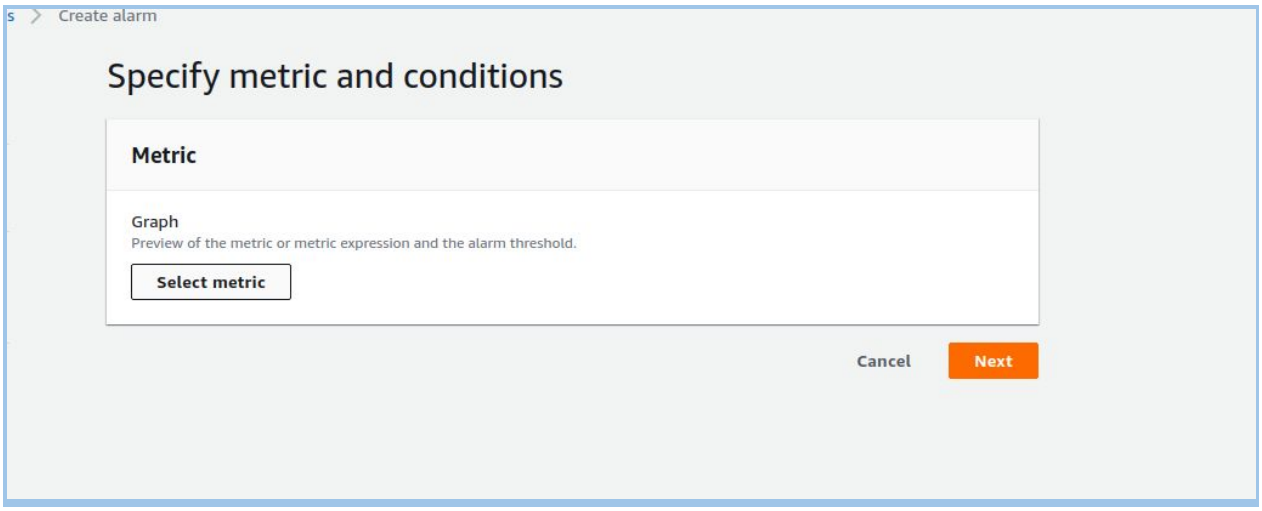
Billing Matrix is not available.

Alarm created on CPU Utilization.

Step 1: Goto CloudWatch -> Alarms->Create Alarm

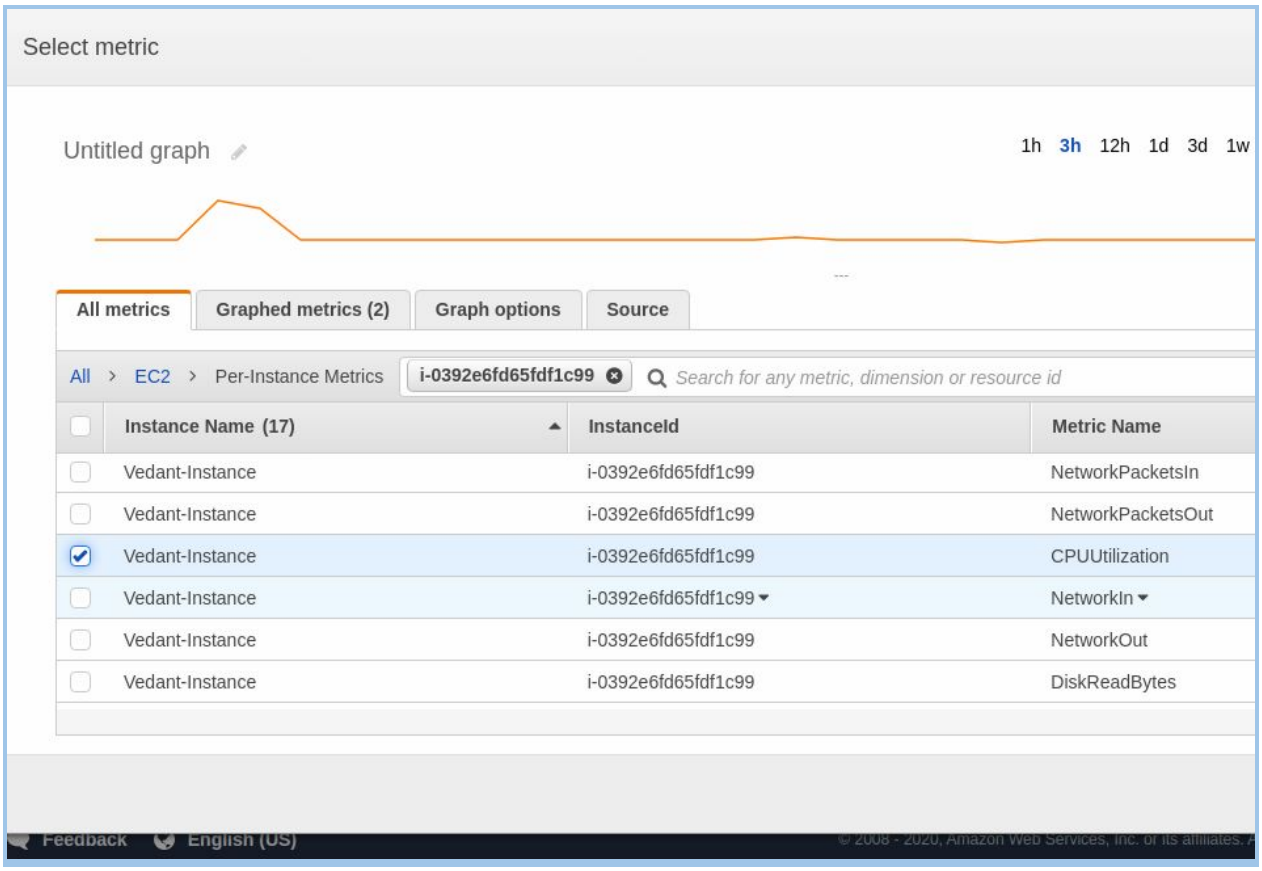


Step 2: Select a metric

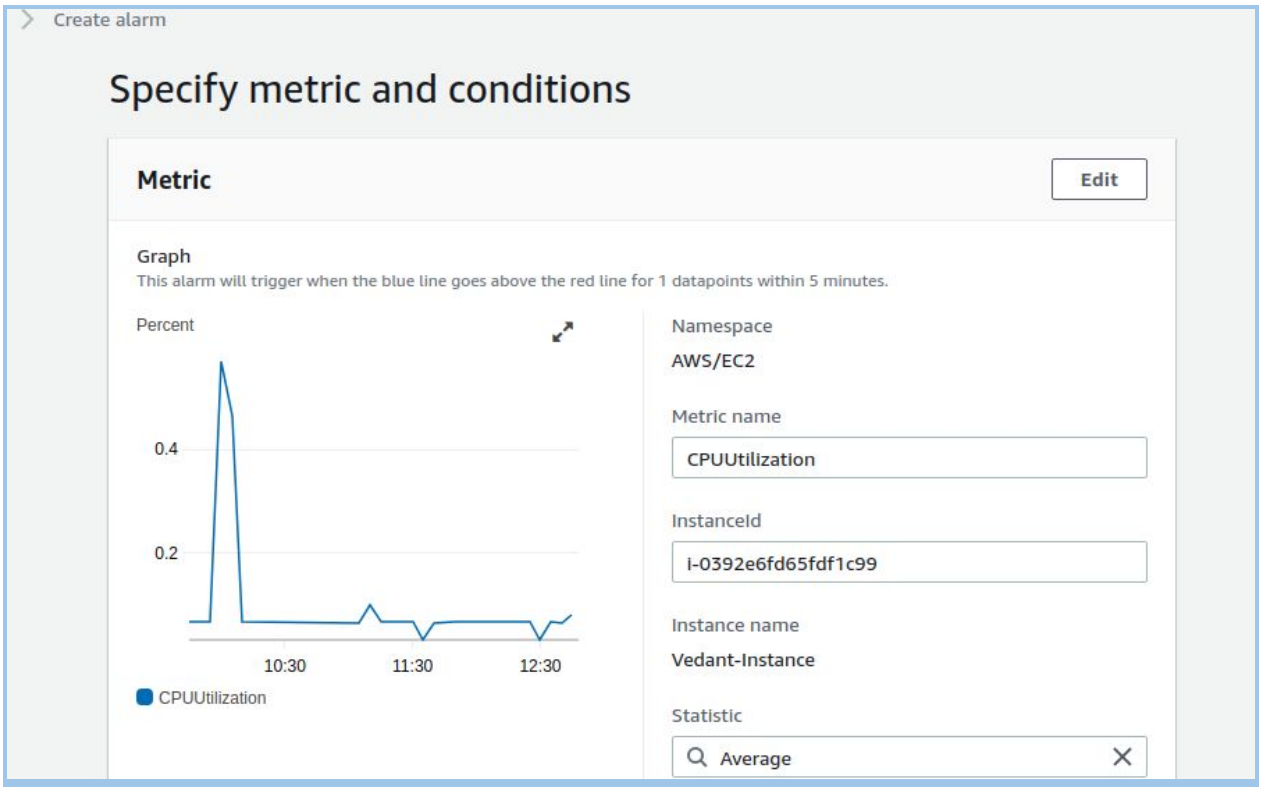


Ec2 -> Per instance metric -> cpu utilization(a particular instance)

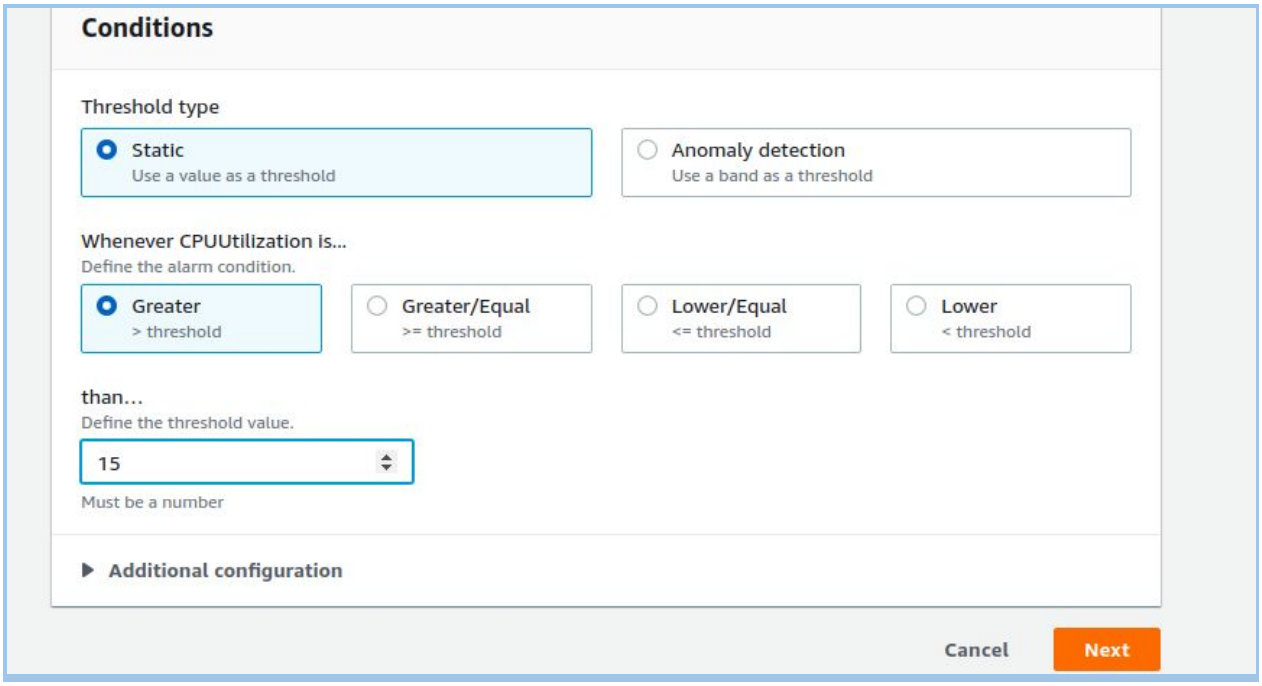
Used an instance already created in aws.



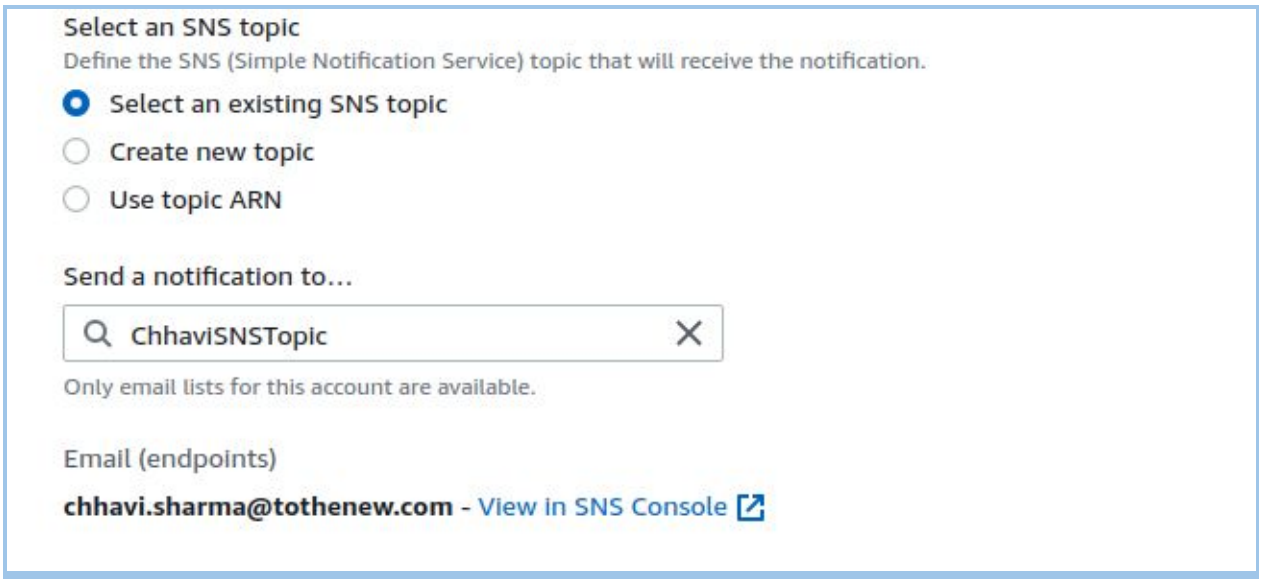
Specify metric condition.



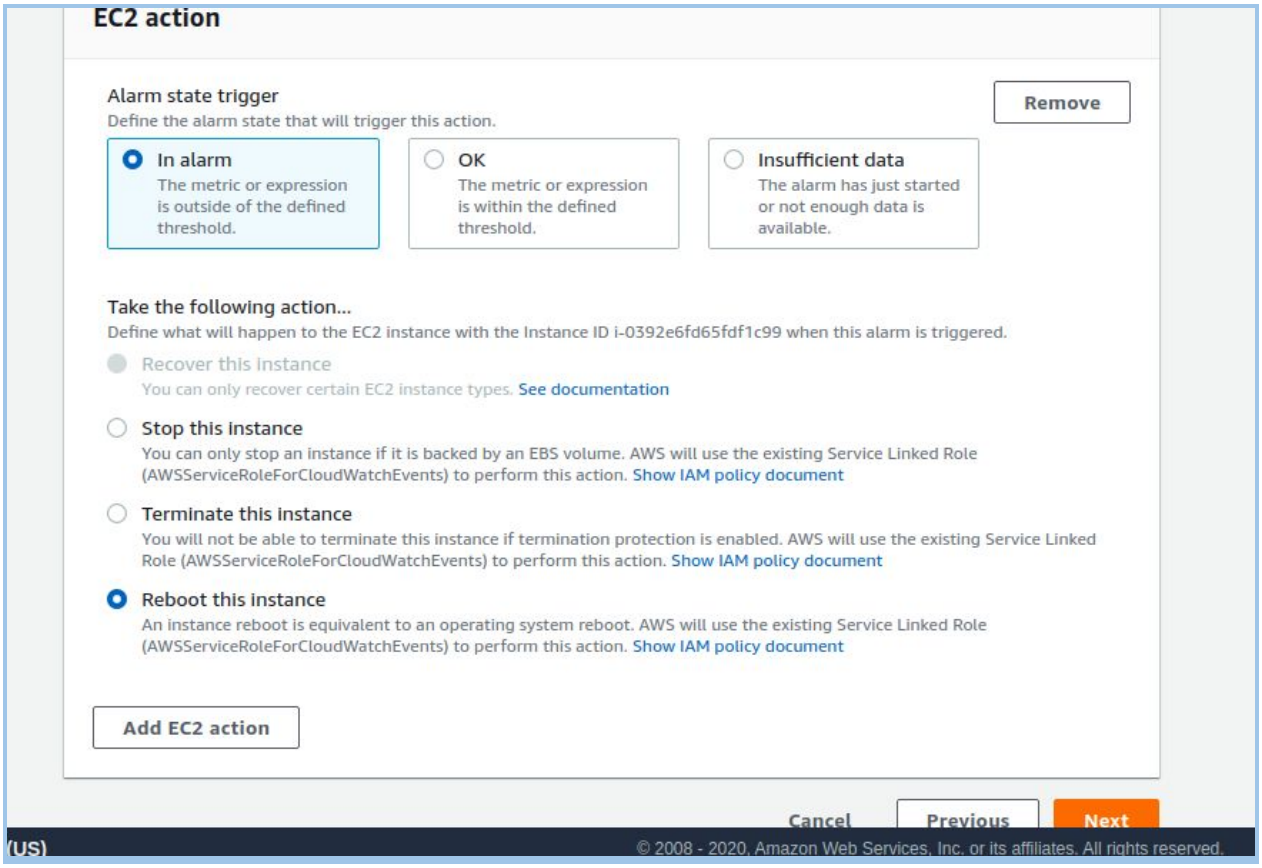
Specify a threshold value.



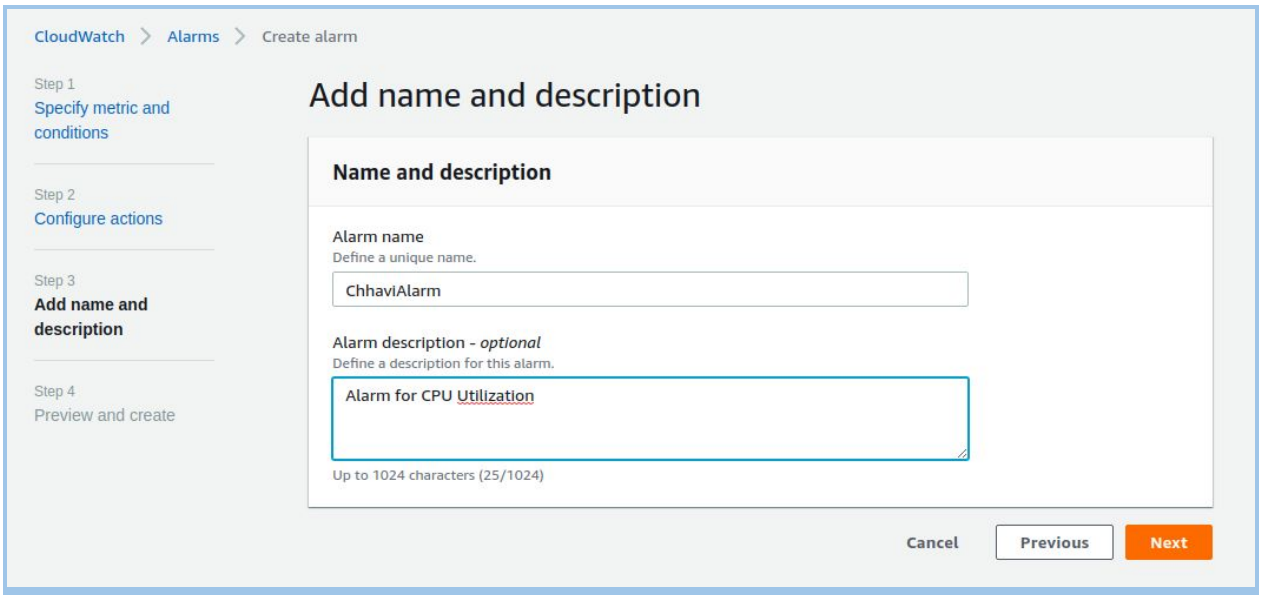
Configure actions -> create a topic



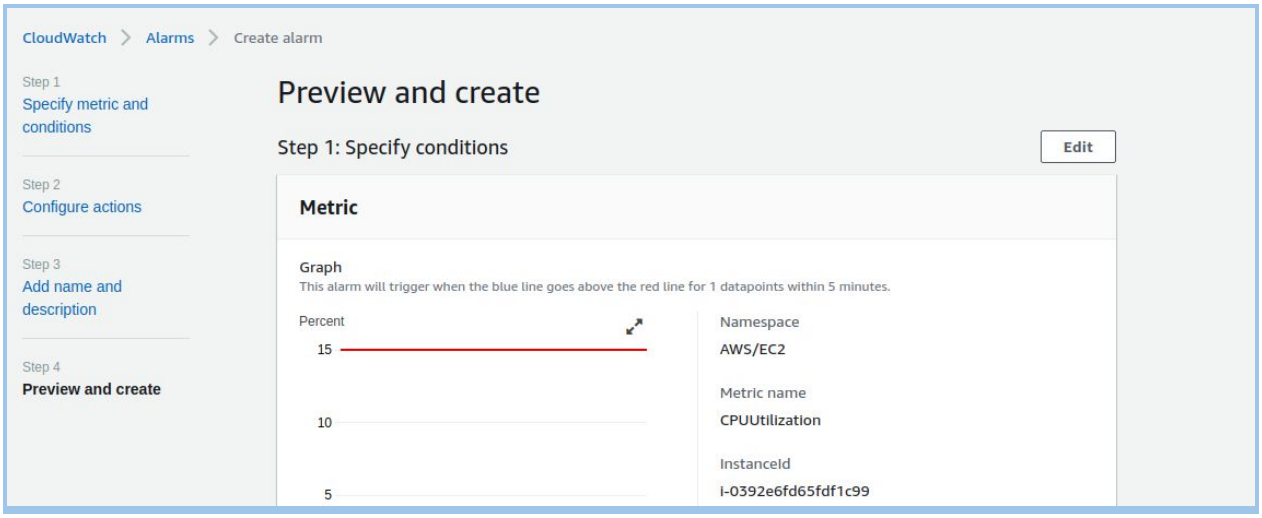
In EC2 Action -> select the desired action to be taken.

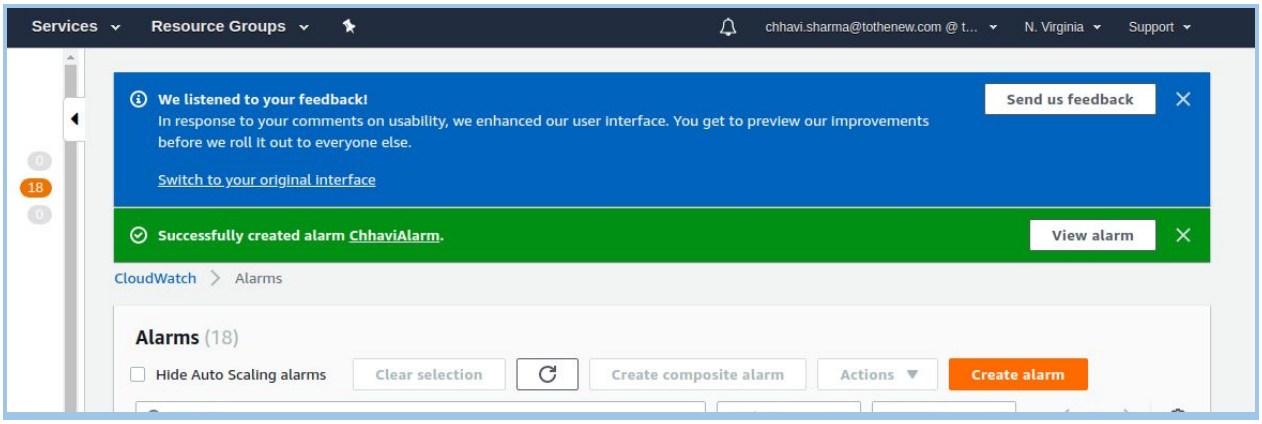


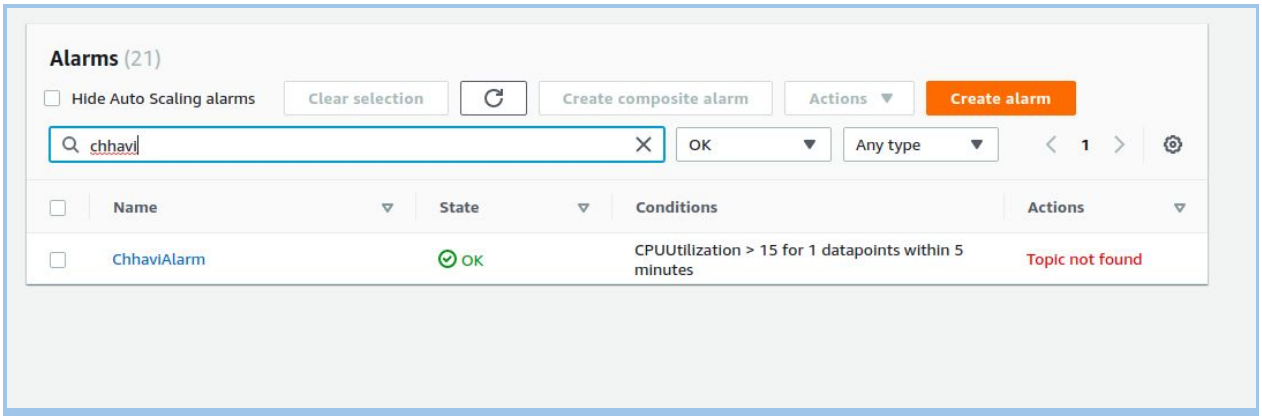
Next. Create an alarm.


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
Define a unique name.
ChhaviAlarm
Alarm description - optional
Define a description for this alarm.
Alarm for CPU Utilization
Up to 1024 characters (25/1024)
Cancel
Previous
Next

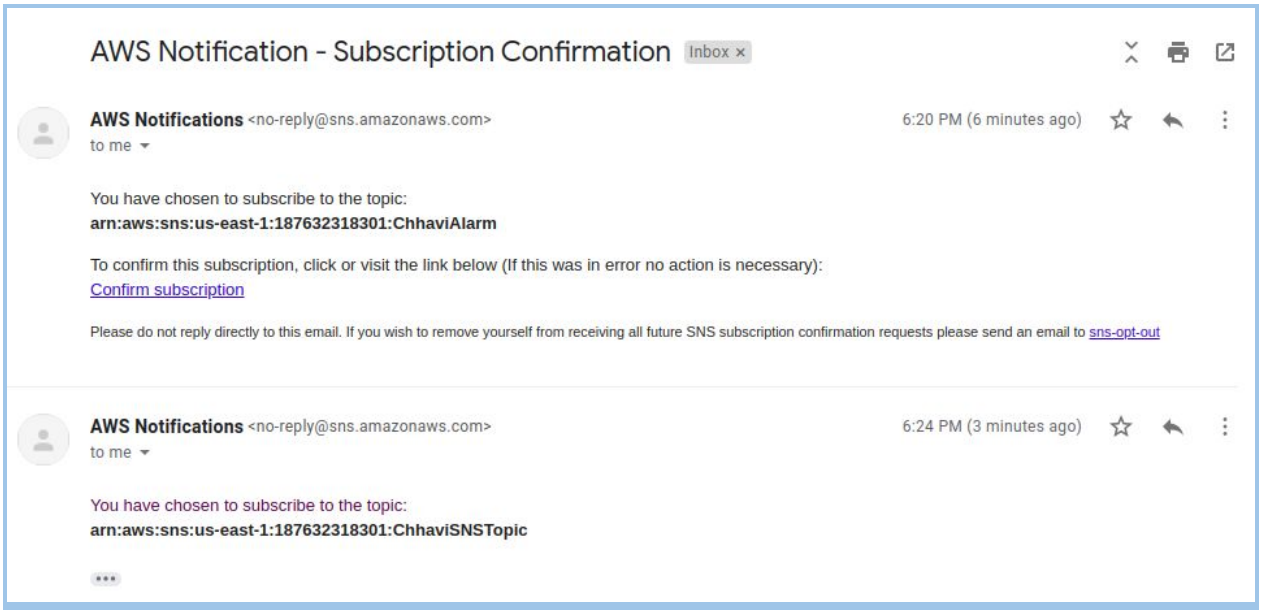
Preview

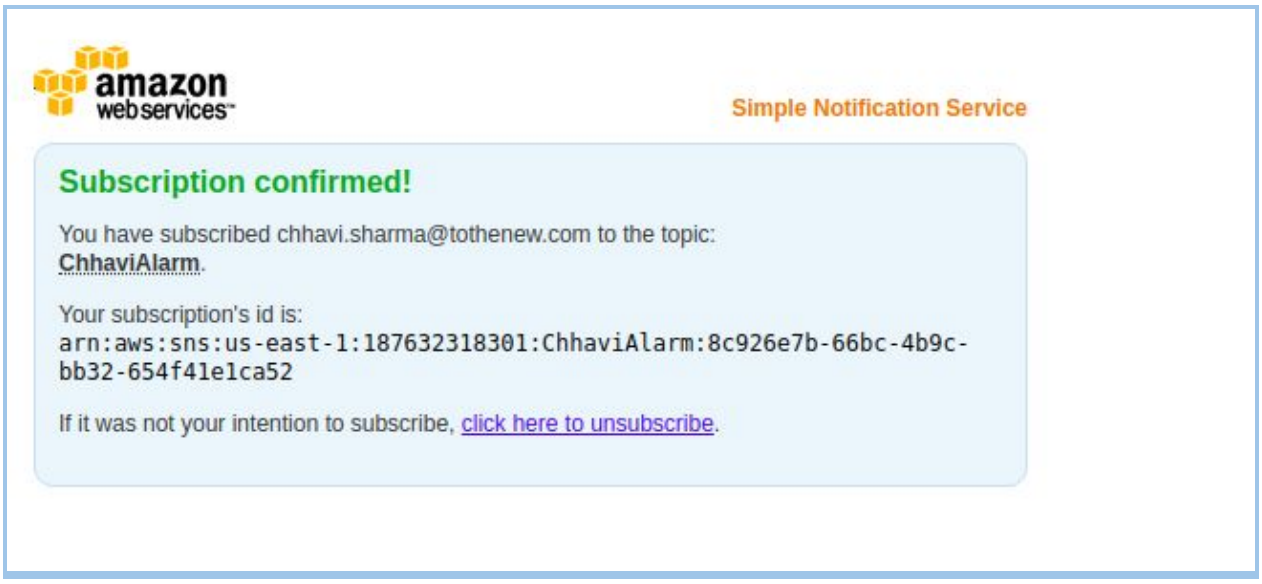

CloudWatch > Alarms > Create alarm
Step 1
Specify metric and conditions
Step 2
Configure actions
Step 3
Add name and description
Step 4
Preview and create
Preview and create
Step 1: Specify conditions
Edit
Metric
Graph
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.
Percent
15
10
5
Namespace
AWS/EC2
Metric name
CPUUtilization
InstanceId
i-0392e6fd65fdf1c99


Services
Resource Groups
chhavi.sharma@tothenew.com @ t...
N. Virginia
Support
We listened to your feedback!
In response to your comments on usability, we enhanced our user interface. You get to preview our improvements before we roll it out to everyone else.
Switch to your original interface
Send us feedback
Successfully created alarm ChhaviAlarm.
View alarm
CloudWatch > Alarms
Alarms (18)
Hide Auto Scaling alarms
Clear selection
Create composite alarm
Actions
Create alarm



Now confirm from your mail.



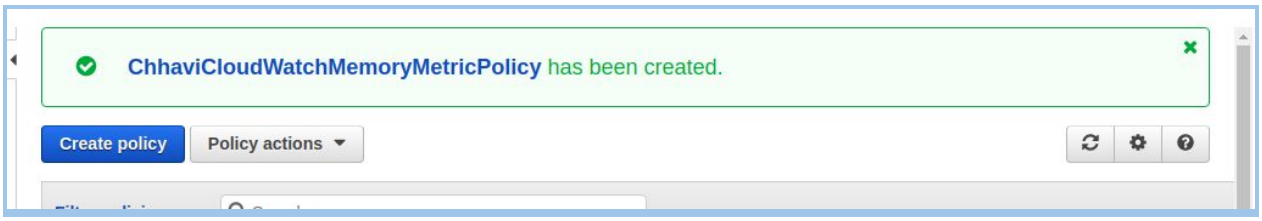


2. Create a custom Memory metric in CloudWatch and set up alarm at 80 % which will autoscale the instance in the autoscaling group.

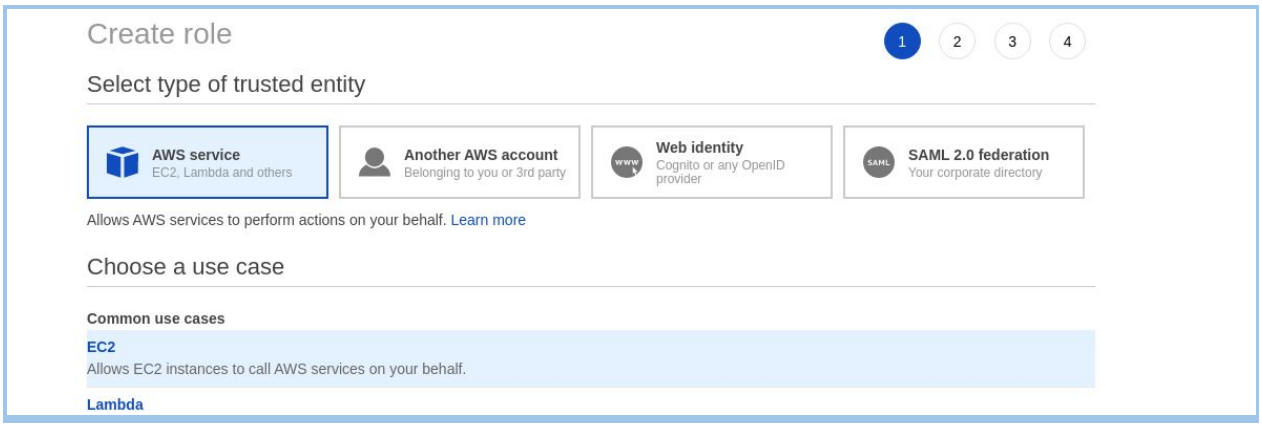
Ans.

Create a policy so that we can push the data from ec2 to cloudWatch.

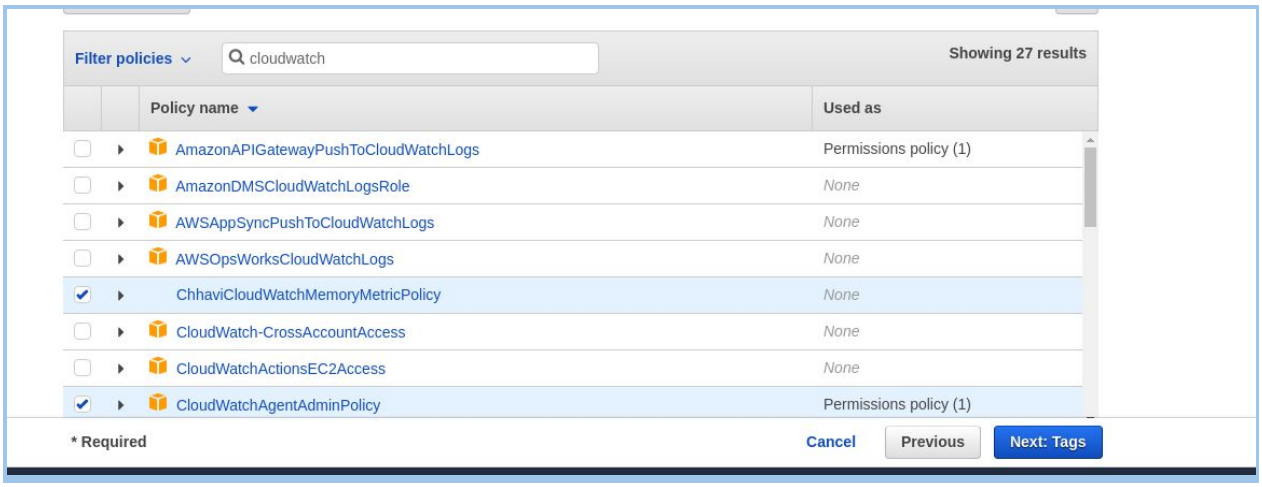
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "cloudwatch:PutMetricData",
            "Resource": "*"
        }
    ]
}
```



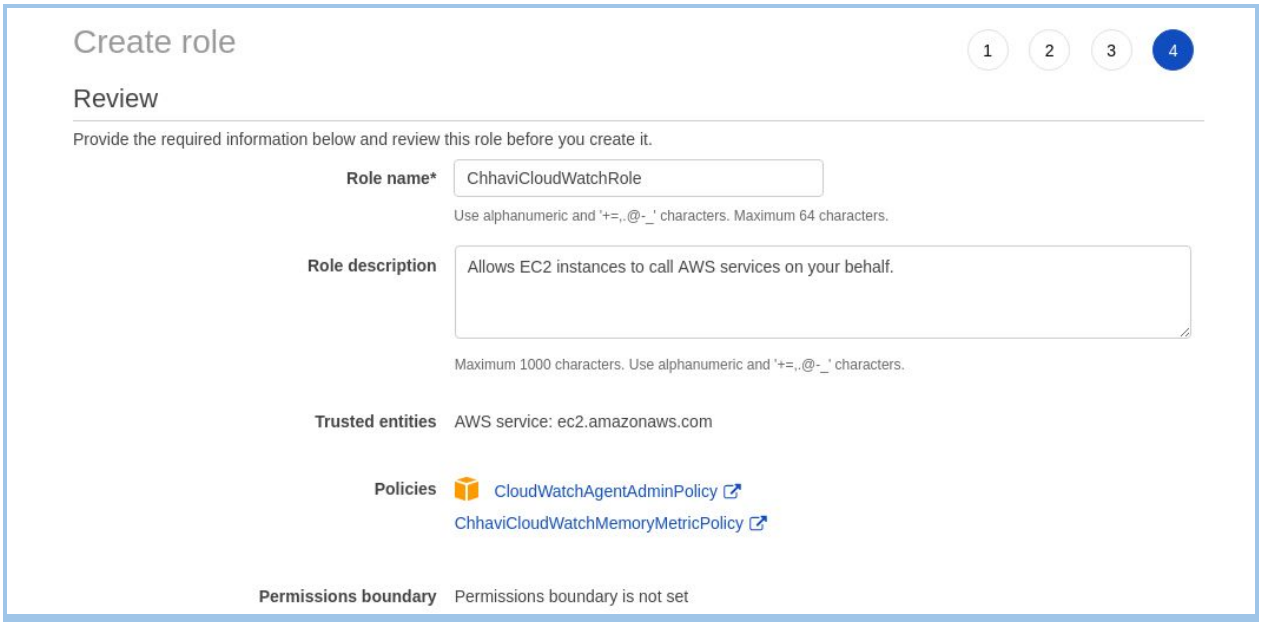
Create a role for ec2.



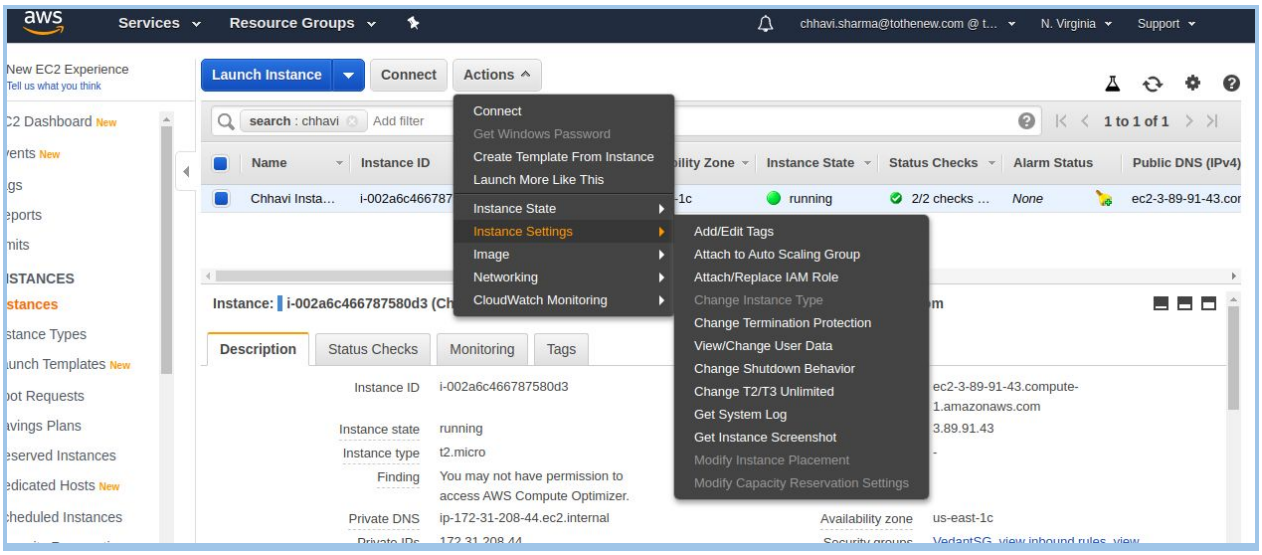
Select the policies

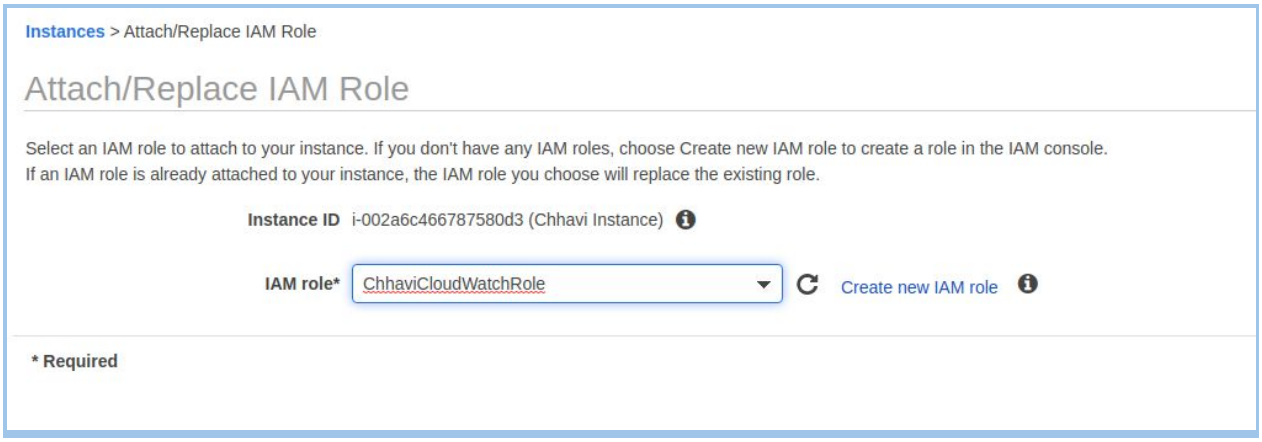


->CloudWatchAgentAdminPolicy



Now attach the role created to your ec2 instance.





Now ssh into your instance.

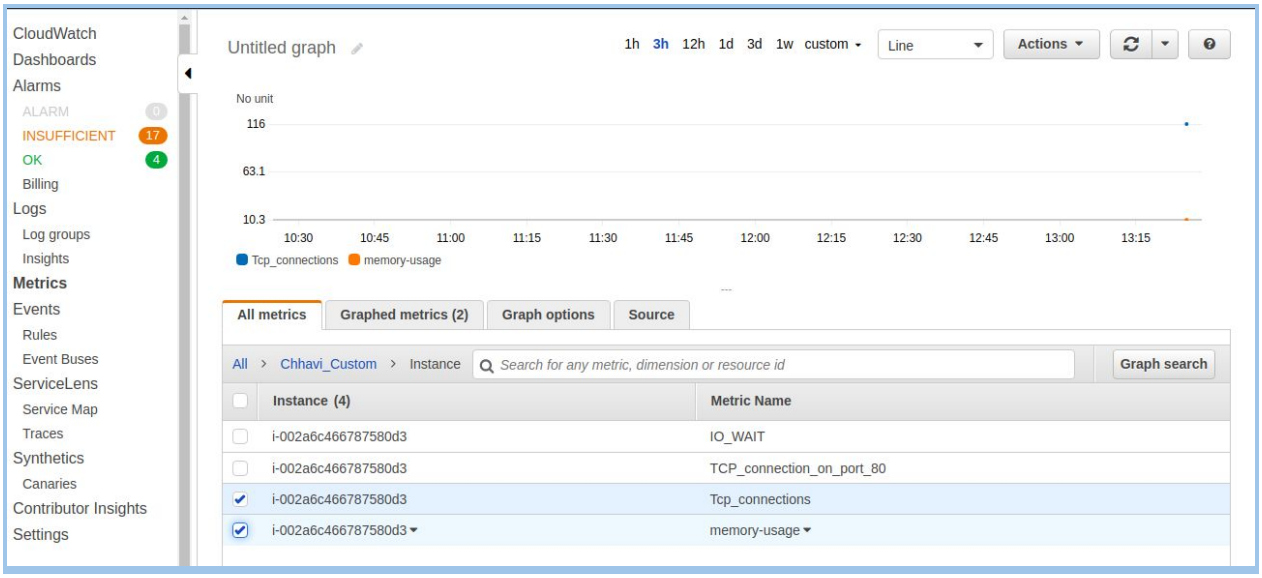
Paste the following script and run the script.

```
#!/bin/bash
USEDMEMORY=$(free -m | awk 'NR==2{printf "%.2f\t", $3*100/$2 }')
TCP_CONN=$(netstat -an | wc -l)
TCP_CONN_PORT_80=$(netstat -an | grep 80 | wc -l)
USERS=$(uptime |awk '{ print $6 }')
IO_WAIT=$(iostat | awk 'NR==4 {print $5}')

aws cloudwatch put-metric-data --metric-name memory-usage --dimensions Instance=i-002a6c
466787580d3  --namespace "Chhavi_Custom" --value $USEDMEMORY
aws cloudwatch put-metric-data --metric-name Tcp_connections --dimensions Instance=i-002
a6c466787580d3  --namespace "Chhavi_Custom" --value $TCP_CONN
aws cloudwatch put-metric-data --metric-name TCP_connection_on_port_80 --dimensions Inst
ance=i-002a6c466787580d3  --namespace "Chhavi_Custom" --value $TCP_CONN_PORT_80
aws cloudwatch put-metric-data --metric-name No_of_users --dimensions Instance=i-002a6c4
66787580d3  --namespace "Chhavi_Custom" --value $USERS
aws cloudwatch put-metric-data --metric-name IO_WAIT --dimensions Instance=i-002a6c46678
7580d3  --namespace "Chhavi_Custom" --value $IO_WAIT
~
~
~
~
~
```

(install iostat)

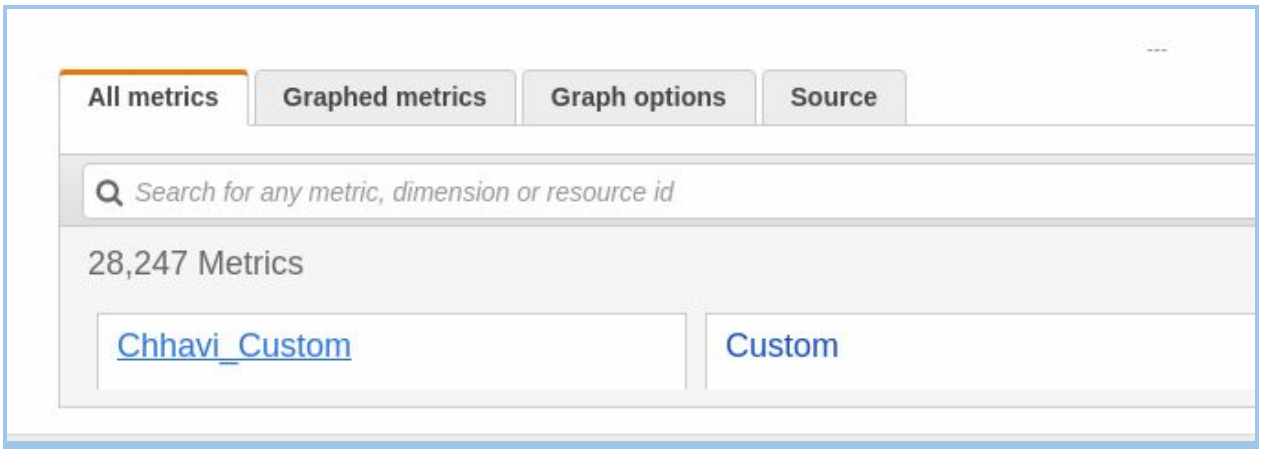
```
ubuntu@ip-172-31-208-44:~$ ./script.sh

[<class 'decimal.ConversionSyntax'>]
ubuntu@ip-172-31-208-44:~$
```

Now goto your console. Goto CloudWatch and select metric.choose your metric.

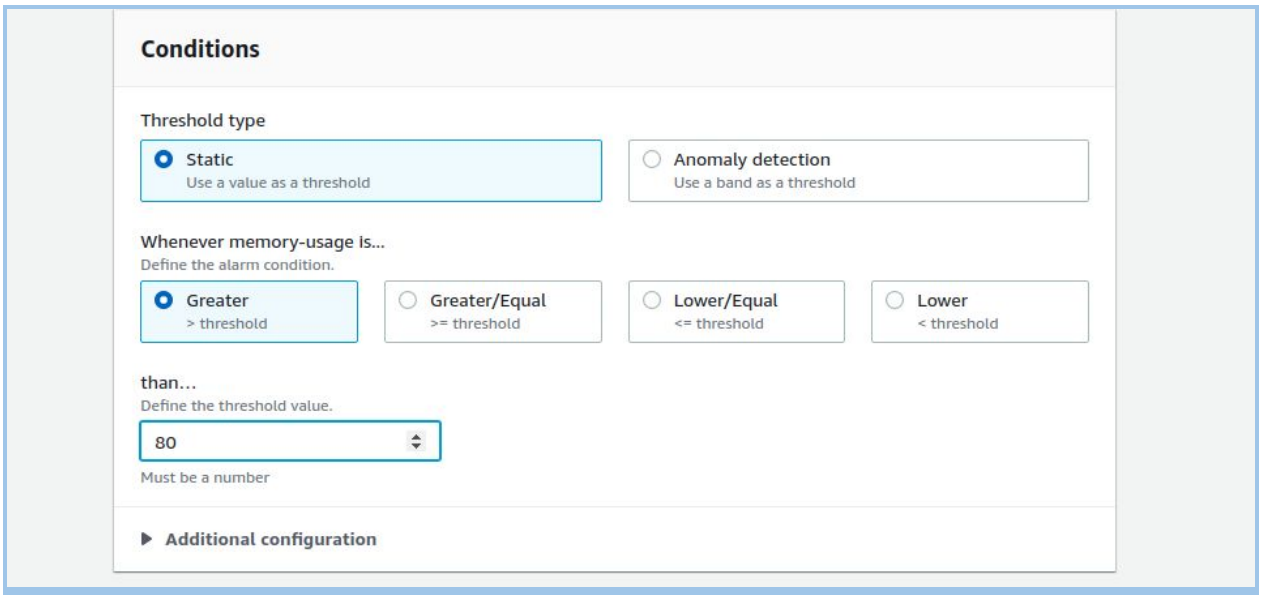


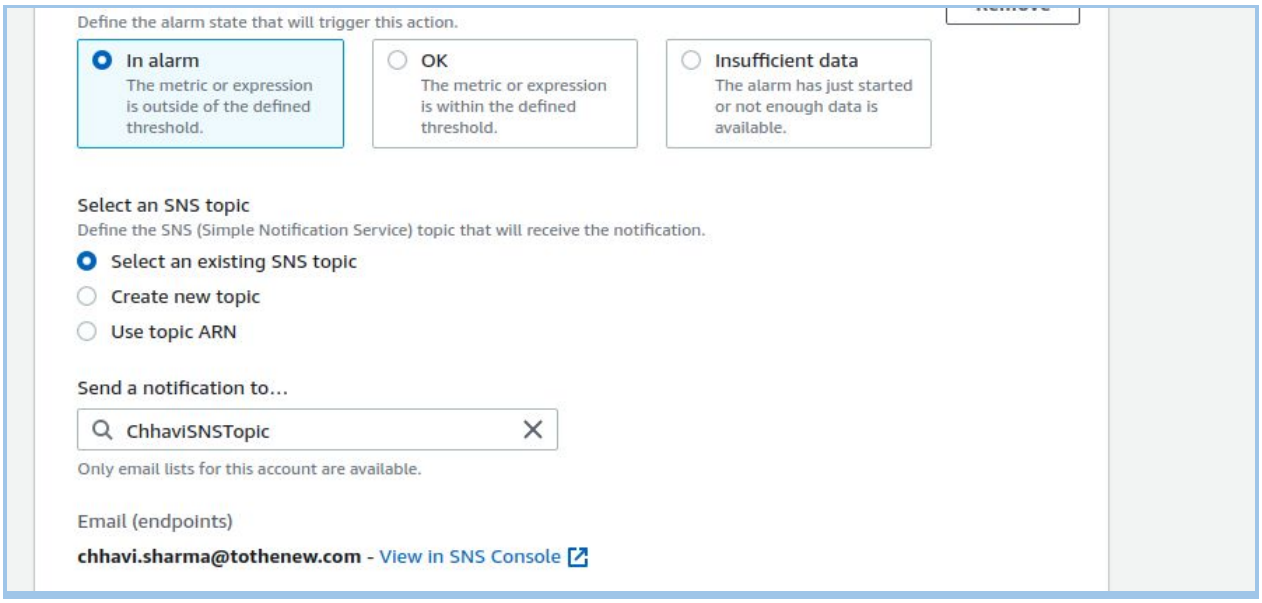
Now create an alarm for 80% utilization in memory.

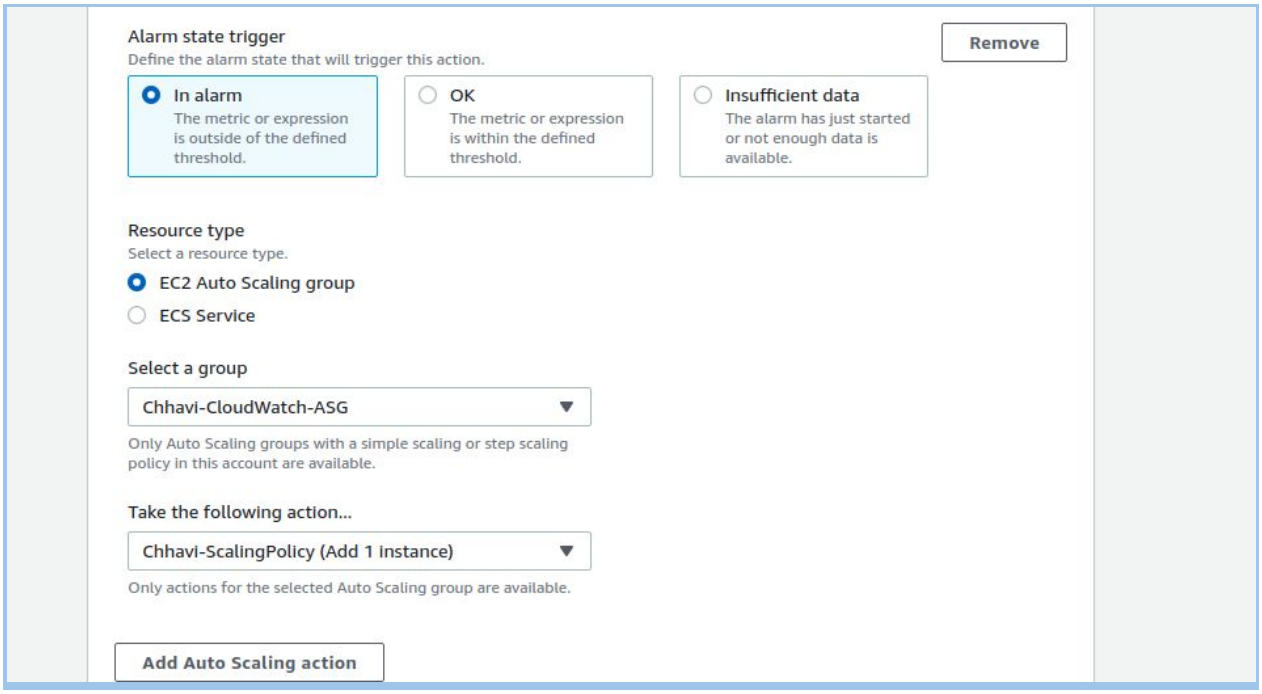
Now when you create an alarm you will get your metric to choose.

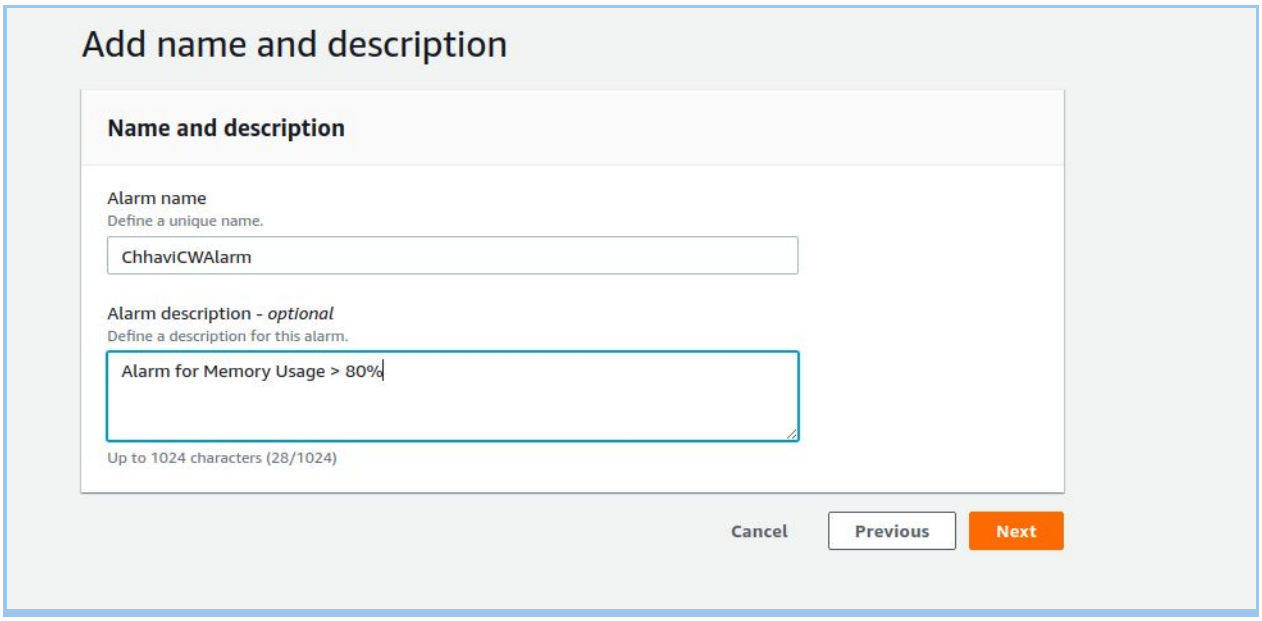


Set threshold to 80 %

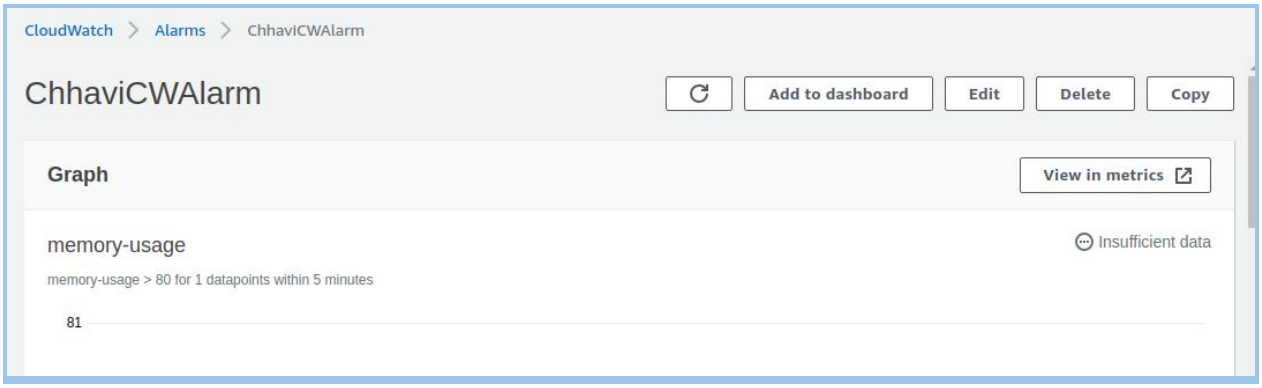



Define the alarm state that will trigger this action.
In alarm
The metric or expression is outside of the defined threshold.
OK
The metric or expression is within the defined threshold.
Insufficient data
The alarm has just started or not enough data is available.
Select an SNS topic
Define the SNS (Simple Notification Service) topic that will receive the notification.
Select an existing SNS topic
Create new topic
Use topic ARN
Send a notification to...
ChhaviSNSTopic
Only email lists for this account are available.
Email (endpoints)
chhavi.sharma@tothenew.com - View in SNS Console


Alarm state trigger
Define the alarm state that will trigger this action.
Remove
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.
Resource type
Select a resource type.
EC2 Auto Scaling group
ECS Service
Select a group
Chhavi-CloudWatch-ASG
Only Auto Scaling groups with a simple scaling or step scaling policy in this account are available.
Take the following action...
Chhavi-ScalingPolicy (Add 1 instance)
Only actions for the selected Auto Scaling group are available.
Add Auto Scaling action


Add name and description
Name and description
Alarm name
Define a unique name.
ChhaviCWAlarm
Alarm description - optional
Define a description for this alarm.
Alarm for Memory Usage > 80%
Up to 1024 characters (28/1024)
Cancel
Previous
Next

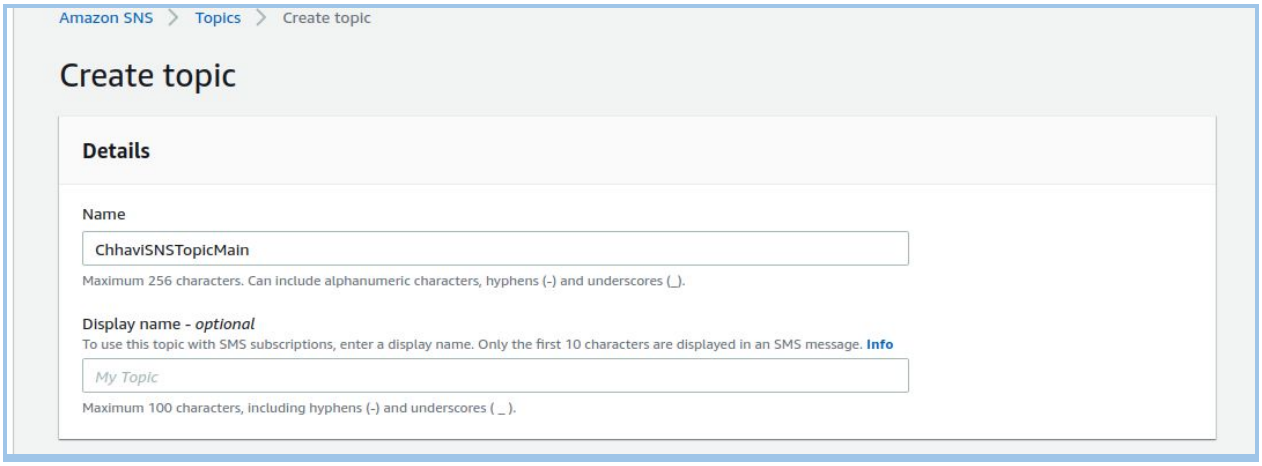

Create Scaling policy
Cancel
Name:
Chhavi-ScalingPolicy
Execute policy when:
ChhaviCloudWatchAlarm
Create new alarm
breaches the alarm threshold: memory-usage > 80 for 300 seconds
for the metric dimensions Instance = i-002a6c466787580d3
Take the action:
Add
1
capacity units
And then wait:
300
seconds before allowing another scaling activity
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy

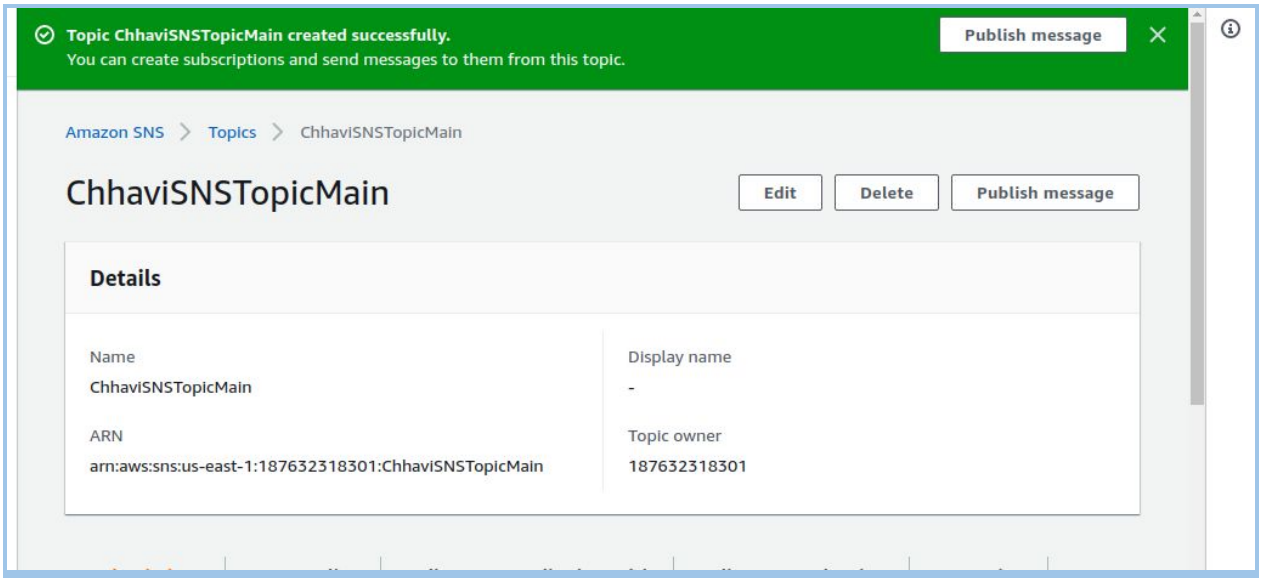

CloudWatch
Alarms
ChhaviCWAlarm
ChhaviCWAlarm
Add to dashboard
Edit
Delete
Copy
Graph
View in metrics
memory-usage
Insufficient data
memory-usage > 80 for 1 datapoints within 5 minutes
81

3. Create SNS topic, subscribe to a topic, publish message, unsubscribe the message and delete the topic.

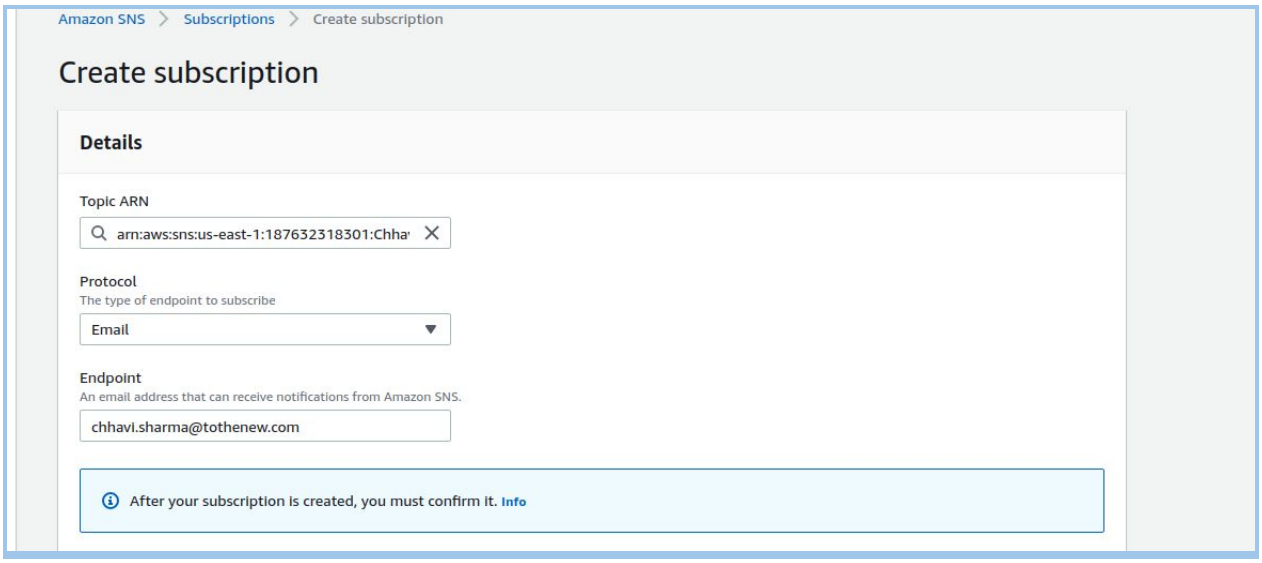
Ans.

Step 1: Select SNS Service -> Select Topic from the left menu. -> Create Topic

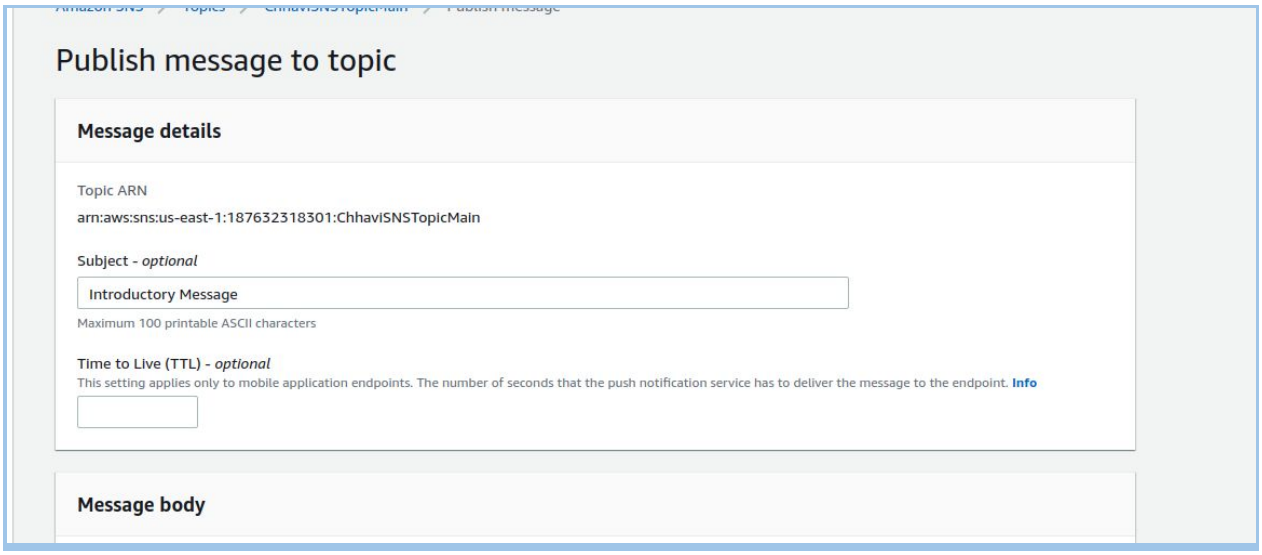




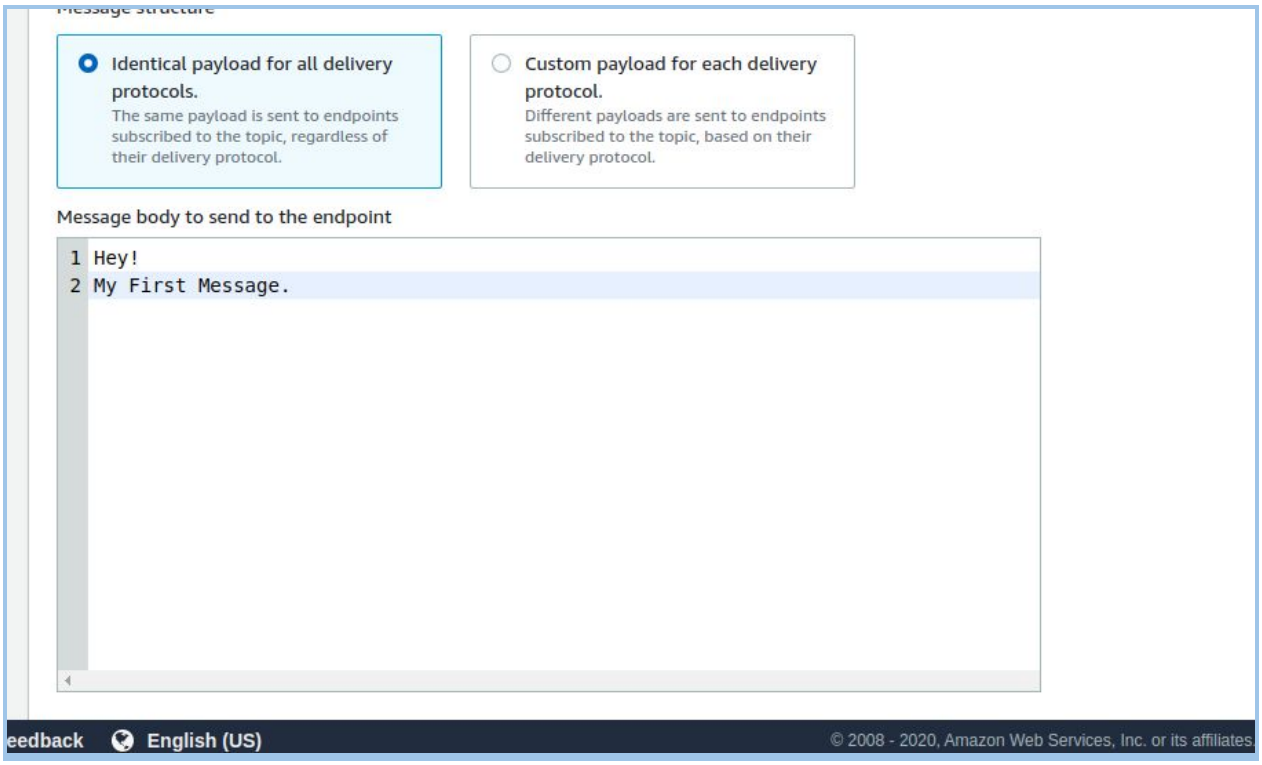
Step 2: Create a subscription.


Amazon SNS > Subscriptions > Create subscription
Create subscription
Details
Topic ARN
arn:aws:sns:us-east-1:187632318301:Chha
Protocol
The type of endpoint to subscribe
Email
Endpoint
An email address that can receive notifications from Amazon SNS.
chhavi.sharma@tothenew.com
After your subscription is created, you must confirm it. Info

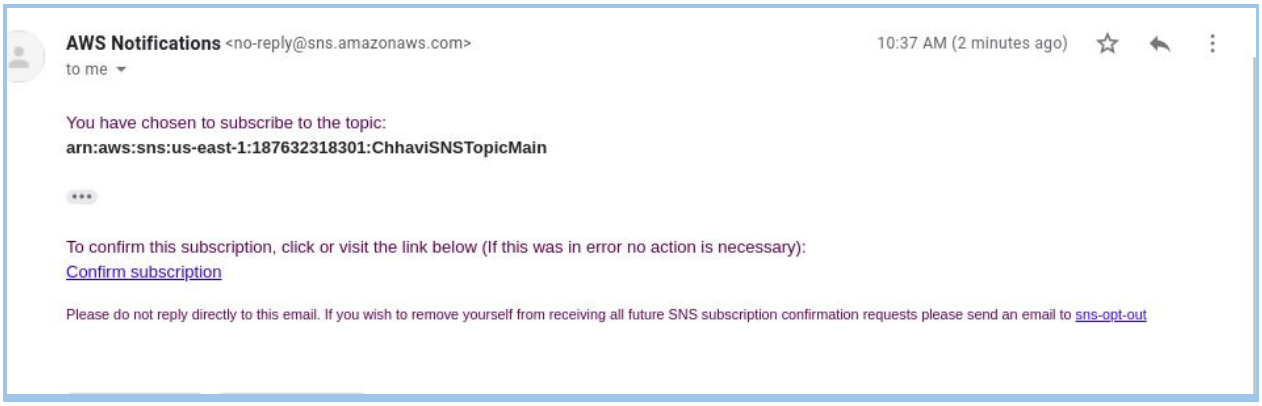
Step 3:

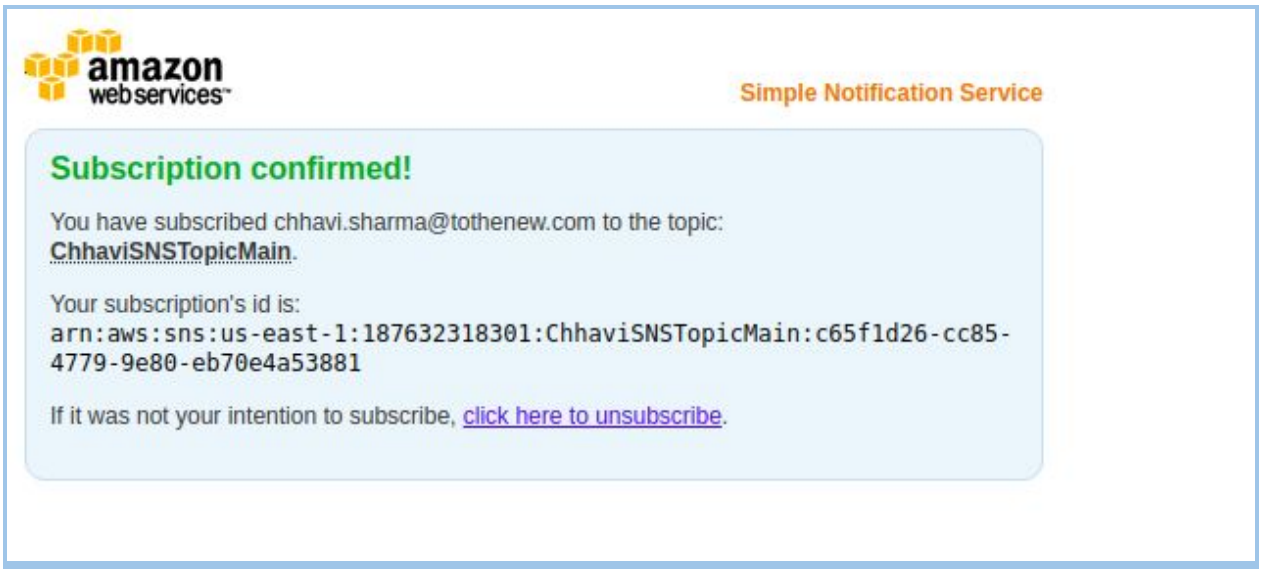

Publish message to topic
Message details
Topic ARN
arn:aws:sns:us-east-1:187632318301:ChhaviSNSTopicMain
Subject - optional
Introductory Message
Maximum 100 printable ASCII characters
Time to Live (TTL) - optional
This setting applies only to mobile application endpoints. The number of seconds that the push notification service has to deliver the message to the endpoint. Info
Message body

Step 4:



Step 5: Confirm Subscription.





4. Send a sample mail using SES.

Ans. Not Authorized to perform this action.