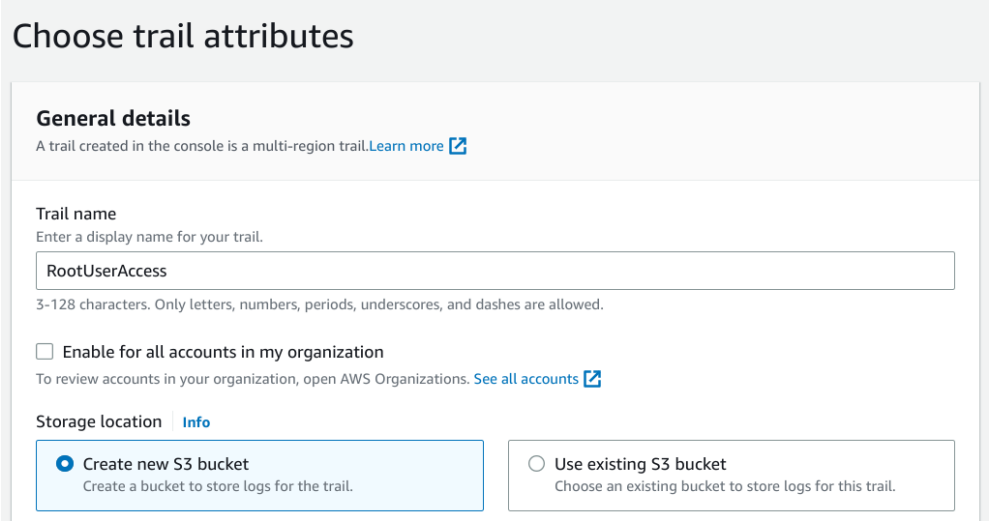
AWS Security

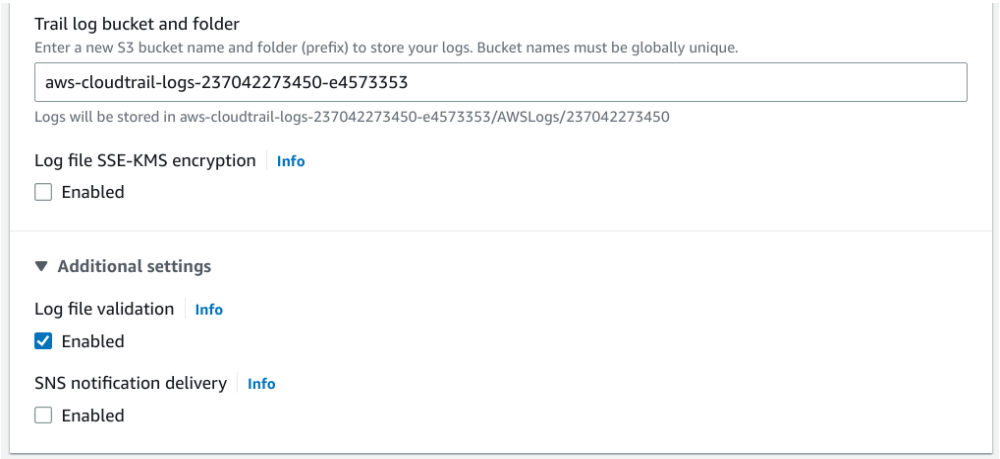
Day 2 - Assignment 11th July 2023

Assignment 1

Enable CloudTrail and use CloudWatch to generate alert through SNS

Step 1 - Create a new CloudTrail Trail. Select a new S3 bucket in which the logs will be stored, allow cloudwatch logs to store the logs in log groups.





CloudWatch Logs - *optional*

Configure CloudWatch Logs to monitor your trail logs and notify you when specific activity occurs. Standard CloudWatch and CloudWatch Logs charges apply. Learn more

CloudWatch Logs | Info

☑ Enabled

◉ New
○ Existing

Log group name

aws-cloudtrail-logs-237042273450-754d4402

1-512 characters. Only letters, numbers, dashes, underscores, forward slashes, and periods are allowed.

AWS CloudTrail assumes this role to send CloudTrail events to your CloudWatch Logs log group.

◉ New
○ Existing

Role name

CloudTrail_logsTo_cloudWatch

▶ Policy document

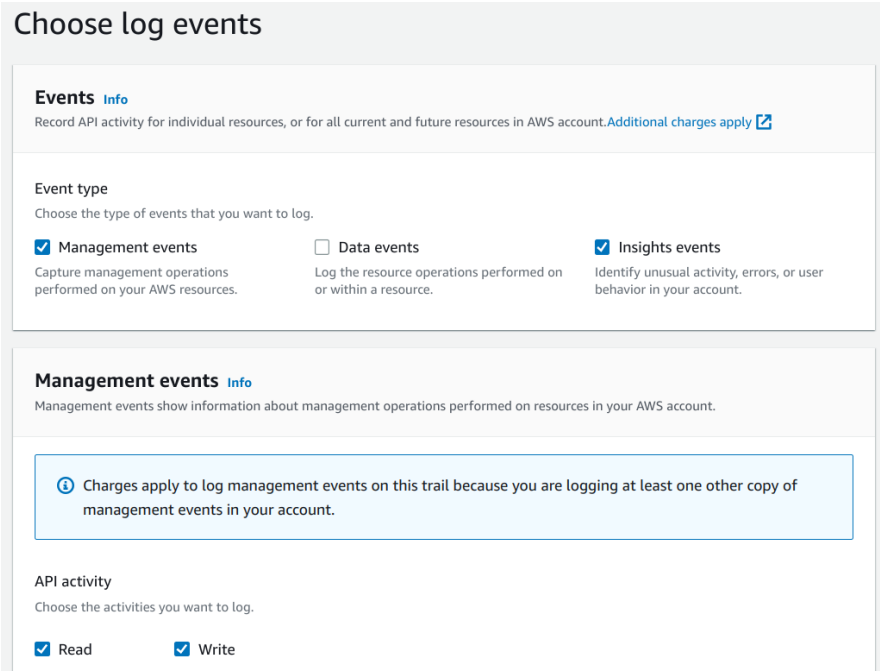
▼ Policy document

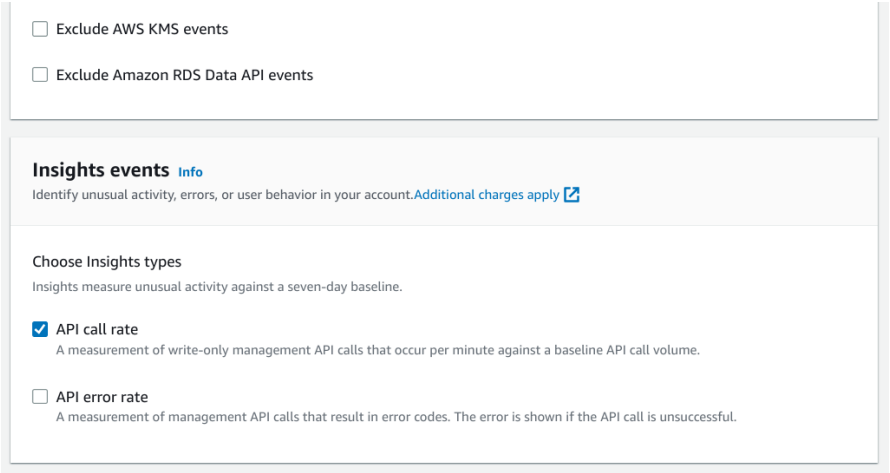
JSON view

Copy

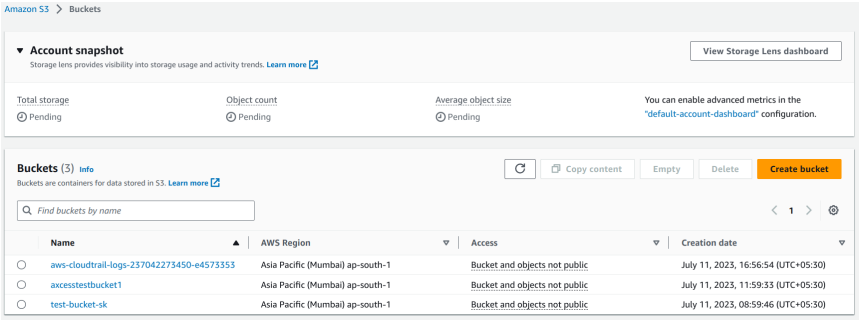

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSCloudTrailCreateLogStream2014110",
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream"
      ],
      "Resource": [
        "arn:aws:logs:ap-south-1:237042273450:log-group:aws-cloudtrail-logs-237042273450-754d4402:log-stream:237042273450_CloudTrail_ap-south-1*"
      ]
    },
    {
      "Sid": "AWSCloudTrailPutLogEvents20141101",
      "Effect": "Allow",
      "Action": [
        "logs:PutLogEvents"
      ],
      "Resource": [
        "arn:aws:logs:ap-south-1:237042273450:log-group:aws-cloudtrail-logs-237042273450-754d4402:log-stream:237042273450_CloudTrail_ap-south-1*"
      ]
    }
  ]
```

Step 2 - Choose a log event. To log events for root user sign-in, select insight events.





Step 3 - We can see the S3 bucket is created for storing cloudtrail logs.



Step 4 - Create a metric filter in the cloudwatch log group for root user signin event.

Define pattern

Create filter pattern

You can use metric filters to monitor events in a log group as they are sent to CloudWatch Logs. You can monitor and count specific terms or extract values from log events and associate the results with a metric. **Learn more about pattern syntax.**

Filter pattern

Specify the terms or pattern to match in your log events to create metrics.

```
{ $.userIdentity.type = "Root" && $.userIdentity.invokedBy NOT EXISTS && $.e
```

Test pattern

Select log data to test

Custom log data

Log event messages

Type log data to test with your Filter Pattern. Please use line breaks to separate log events.

```
[83078518-fcc1-4d30-9573-8b9737671438] BENCHMARK : Running Start Crawl
[83078518-fcc1-4d30-9573-8b9737671438] BENCHMARK : Classification compl
[83078518-fcc1-4d30-9573-8b9737671438] INFO : Crawler configured with Sch
[83078518-fcc1-4d30-9573-8b9737671438] INFO : Created table gluetest in dat
[83078518-fcc1-4d30-9573-8b9737671438] BENCHMARK : Finished writing to C
```

Test pattern

Assign metric

Create filter name

Log events that match the pattern you define are recorded to the metric that you specify. You can graph the metric and set alarms to notify you.

Filter name

RootAccountUsage

Filter pattern

```
{ $.userIdentity.type = "Root" && $.userIdentity.invokedBy NOT EXISTS && $.even
```

Metric details

Metric namespace

Namespaces let you group similar metrics. **Learn more**

CloudTrailMetrics

Create new

Namespaces can be up to 255 characters long; all characters are valid except for colon(:) at the start of the name.

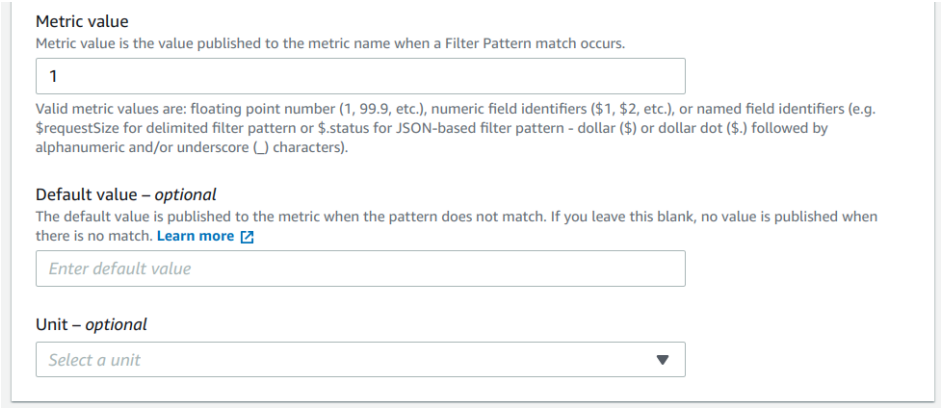
Metric name

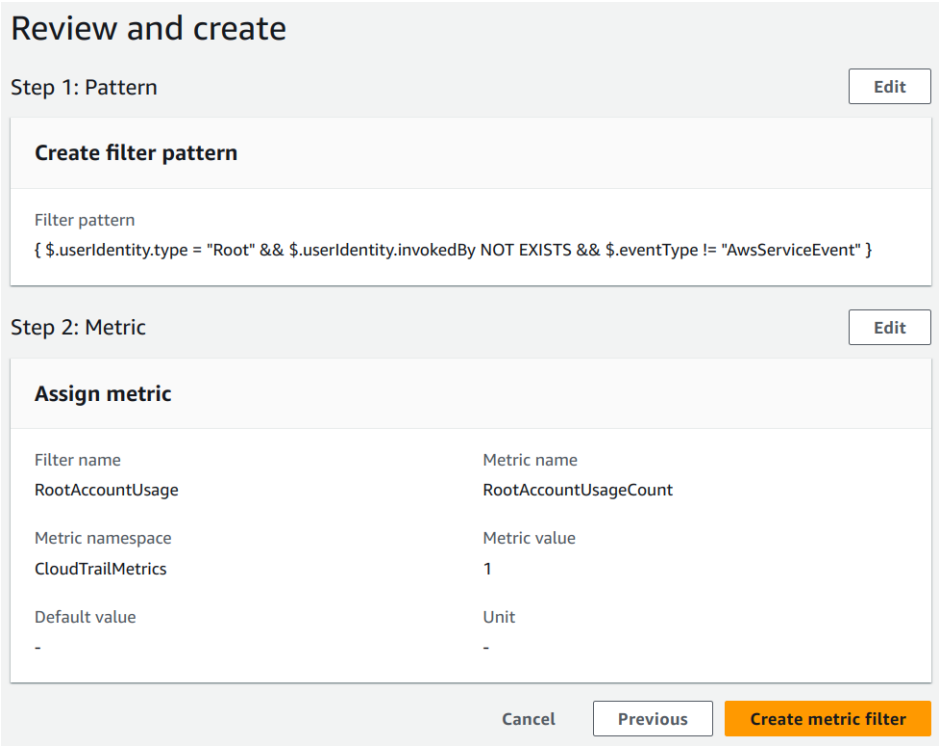
Metric name identifies this metric, and must be unique within the namespace. **Learn more**

RootAccountUsageCount

Metric name can be up to 255 characters long; all characters are valid except for colon(:), asterisk(*), dollar($), and space( ).

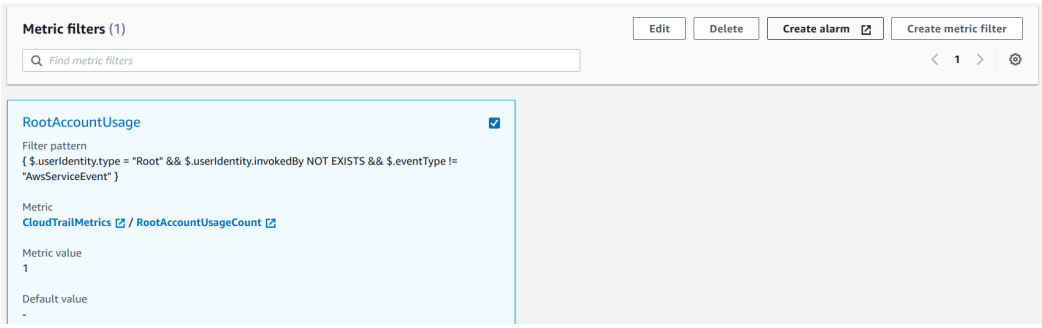
Metric value



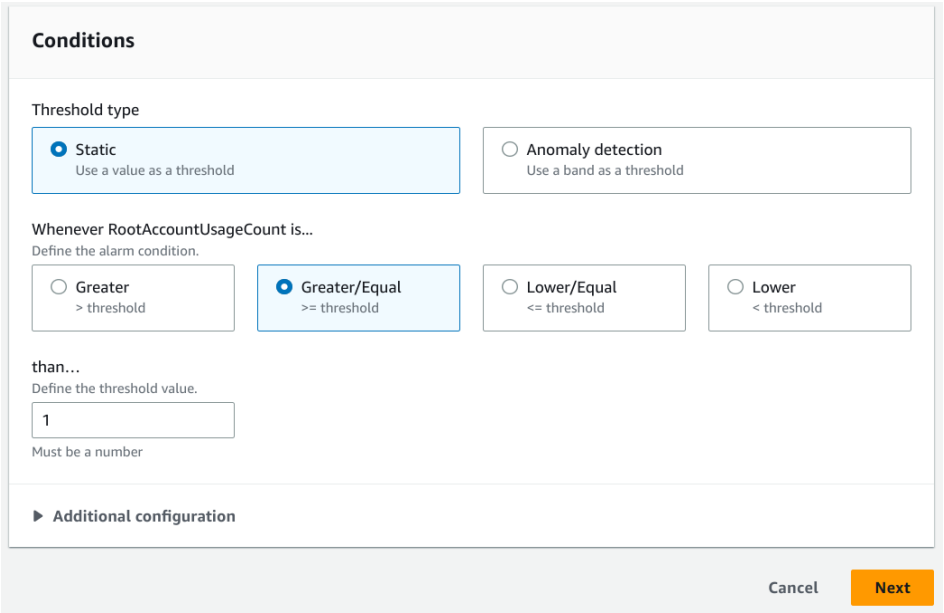


Step 5 - Create a new alarm for the above metric filter.

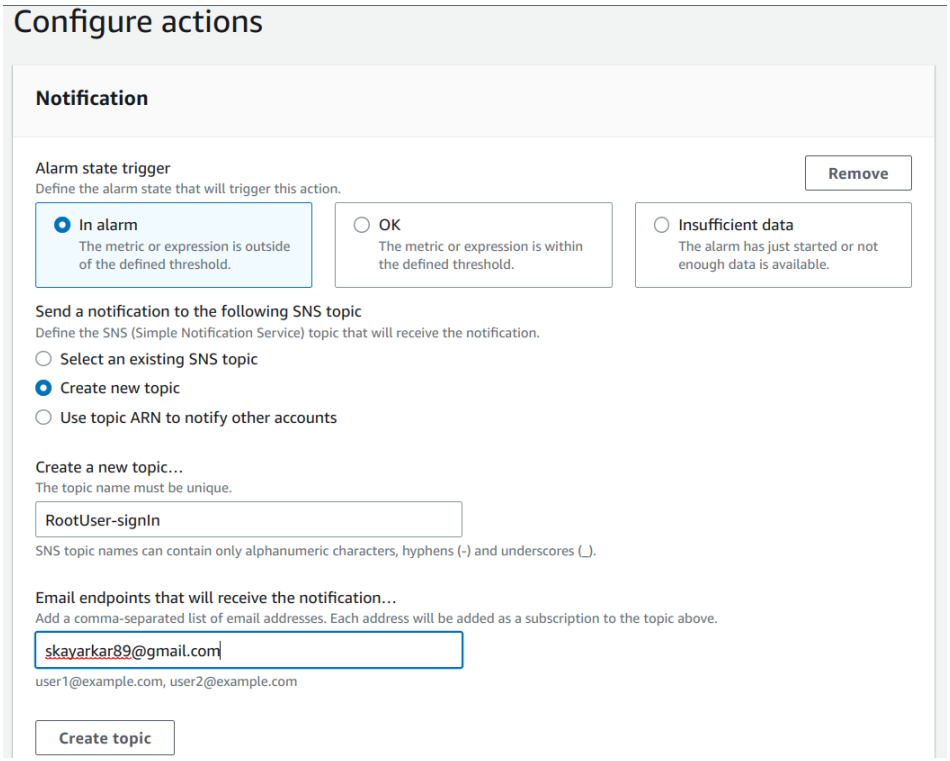
Click on “Create alarm”.

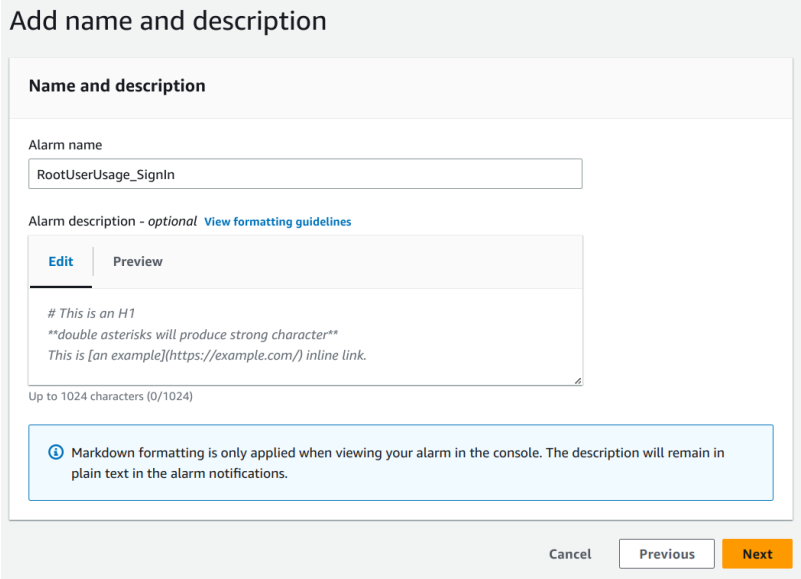


Define condition and threshold for the alarm.

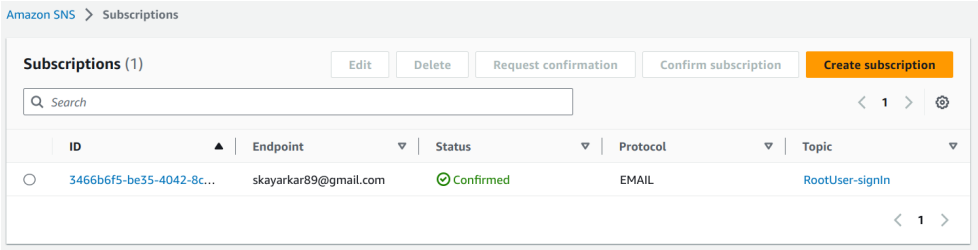


Select “In Alarm” for state, and create a new SNS topic.

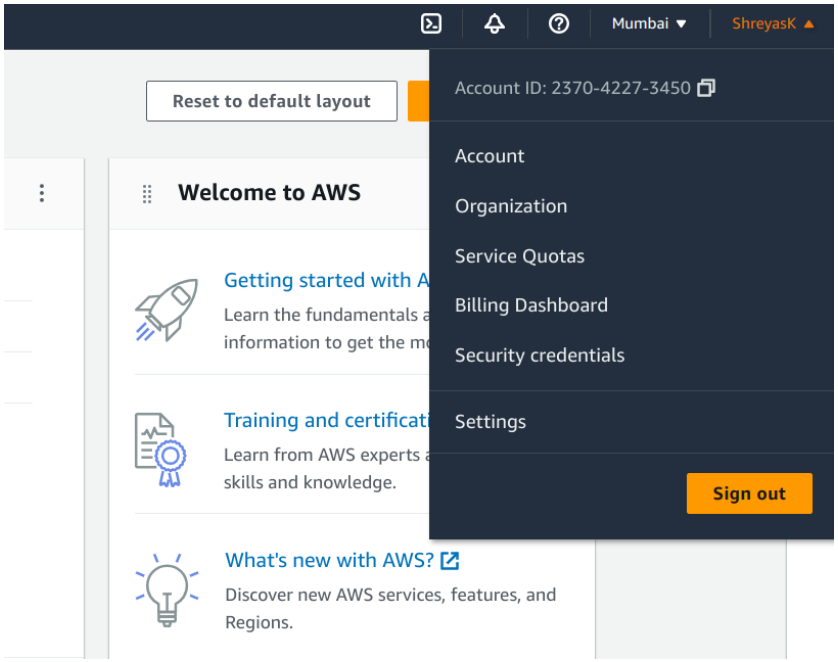




Step 6 - Confirm the SNS subscription.



Step 7 - Login to the root user.



Step 8 - SNS will send notification alert to the specified email address.

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

AWS Notifications <no-reply@sns.amazonaws.com> 5:53 PM (6 minutes ago)
to me

You are receiving this email because your Amazon CloudWatch Alarm "RootUserUsage_SignIn" in the Asia Pacific (Mumbai) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [6.0 (11/07/23 12:18:00)] was greater than or equal to the threshold (1.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Tuesday 11 July, 2023 12:23:15 UTC".

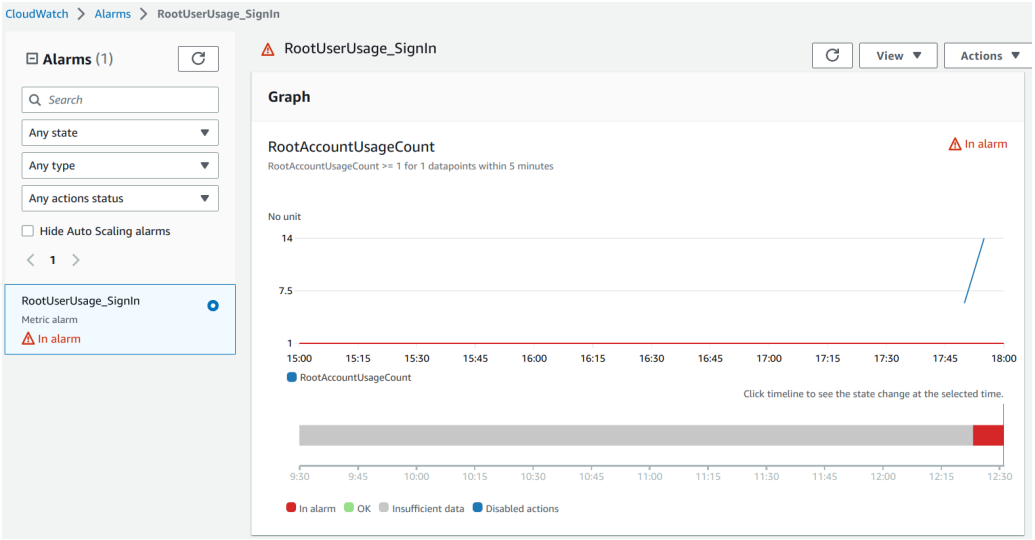
View this alarm in the AWS Management Console:
https://ap-south-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-south-1#alarmsV2:alarm/RootUserUsage_SignIn

Alarm Details:
- Name: RootUserUsage_SignIn
- Description:
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [6.0 (11/07/23 12:18:00)] was greater than or equal to the threshold (1.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Tuesday 11 July, 2023 12:23:15 UTC
- AWS Account: 237042273450
- Alarm Arn: arn:aws:cloudwatch:ap-south-1:237042273450:alarm:RootUserUsage_SignIn

Threshold:
- The alarm is in the ALARM state when the metric is GreaterThanOrEqualToThreshold 1.0 for at least 1 of the last 1 period(s) of 300 seconds.

Monitored Metric:
- MetricNamespace: CloudTrailMetrics
- MetricName: RootAccountUsageCount
- Dimensions:
- Period: 300 seconds
- Statistic: Sum

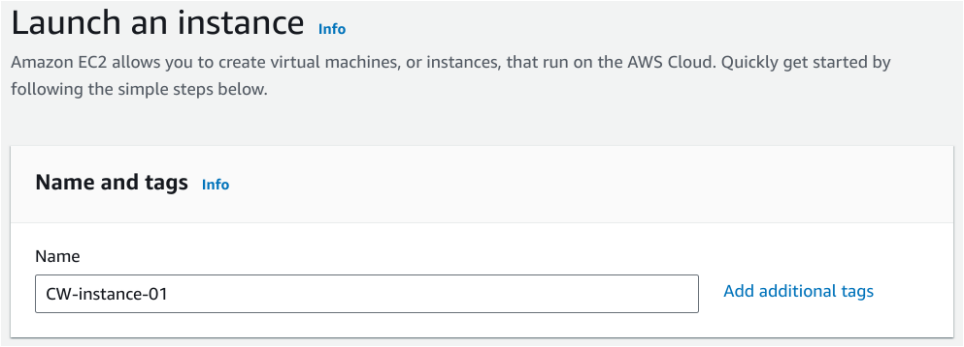
CloudWatch shows the “In Alarm” state for the metric.

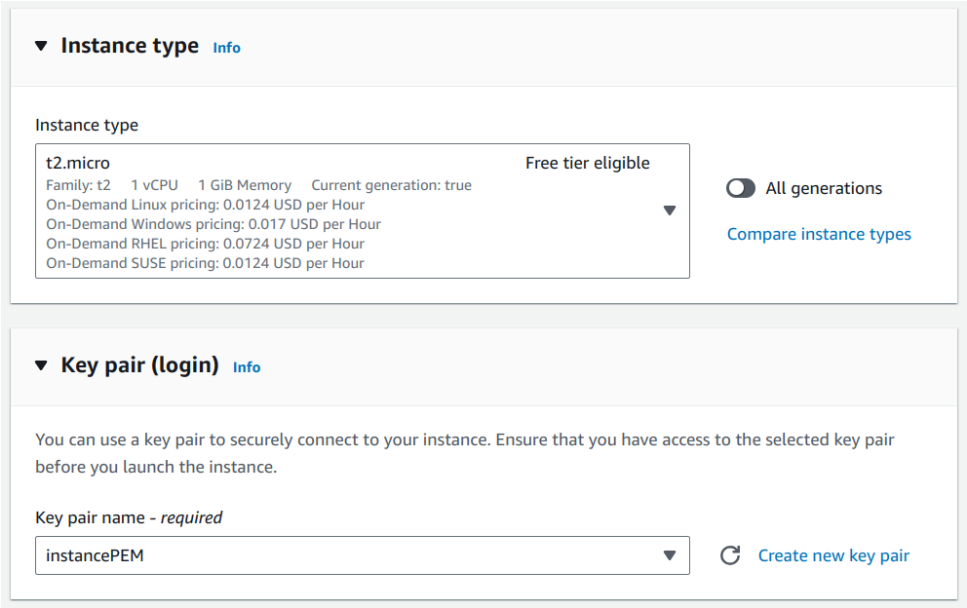


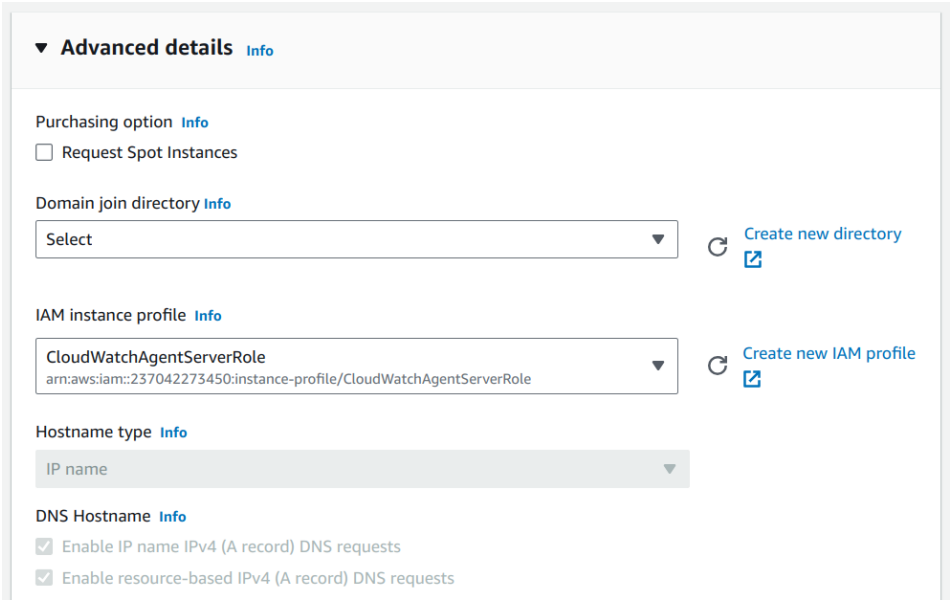
Assignment 2

Launch EC2 instance and configure CloudWatch agent on it to send metrics to cloudwatch.

Step 1 - Create an EC2 instance and attach an IAM role to it with "CloudWatchAgentServerPolicy" permission, which allows the instance to send metrics to cloudwatch.







Step 2 - Installing Cloudwatch agent in the EC2 instance.

```
[ec2-user@ip-172-31-8-217 ~]$ wget https://s3.amazonaws.com/amazoncloudwatch-agent/linux/amd64/latest/AmazonCloudWatchAgent.zip
--2023-07-11 15:42:42--  https://s3.amazonaws.com/amazoncloudwatch-agent/linux/amd64/latest/AmazonCloudWatchAgent.zip
Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.216.208.104, 54.231.231.56, 52.217.133.176, ...
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.216.208.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 70431160 (67M) [application/zip]
Saving to: 'AmazonCloudWatchAgent.zip'

AmazonCloudWatchAgent.zip                 100%[===========================================================================================>]  67.17M  8.28MB/s    in 9.6s

2023-07-11 15:42:53 (6.98 MB/s) - 'AmazonCloudWatchAgent.zip' saved [70431160/70431160]
```

```
[ec2-user@ip-172-31-8-217 ~]$ unzip AmazonCloudWatchAgent.zip
sudo ./install.sh
Archive:  AmazonCloudWatchAgent.zip
  inflating: amazon-cloudwatch-agent.rpm
  inflating: amazon-cloudwatch-agent.deb
  inflating: manifest.json
  inflating: install.sh
  inflating: uninstall.sh
  inflating: detect-system.sh
create group cwagent, result: 0
create user cwagent, result: 0
```

Step 3 - It is required to create a cloudwatch agent configuration file, it is a JSON file that specifies the metrics and logs that the agent is to collect, including custom metrics.

Enter the following values in the cloudwatch agent configuration file wizard.

On which OS are you planning to use the agent?	Linux
Are you using EC2 or On-Premises hosts?	EC2
Which user are you planning to run the agent?	Others - ec2-user
Do you want to turn on the StatsD daemon?	No

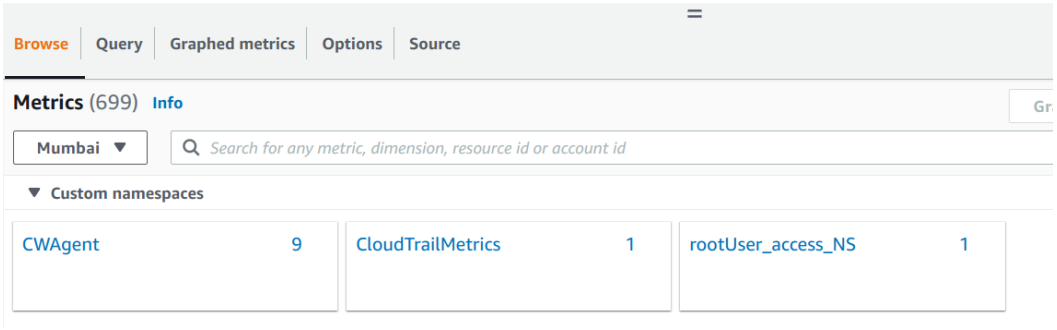
Do you want to monitor metrics from CollectD?	No
Do you want to monitor any host metrics?	Yes
Do you want to monitor cpu metrics per core?	Yes
Do you want to add ec2 dimensions?	Yes
Would you like to collect your metrics at high resolution?	60s
Which default metrics config do you want?	Standard
Are you satisfied with the above config?	Yes
Do you have any existing CloudWatch Log Agent?	No
Do you want to monitor any log files?	No
Do you want to store the config in the SSM parameter store?	No

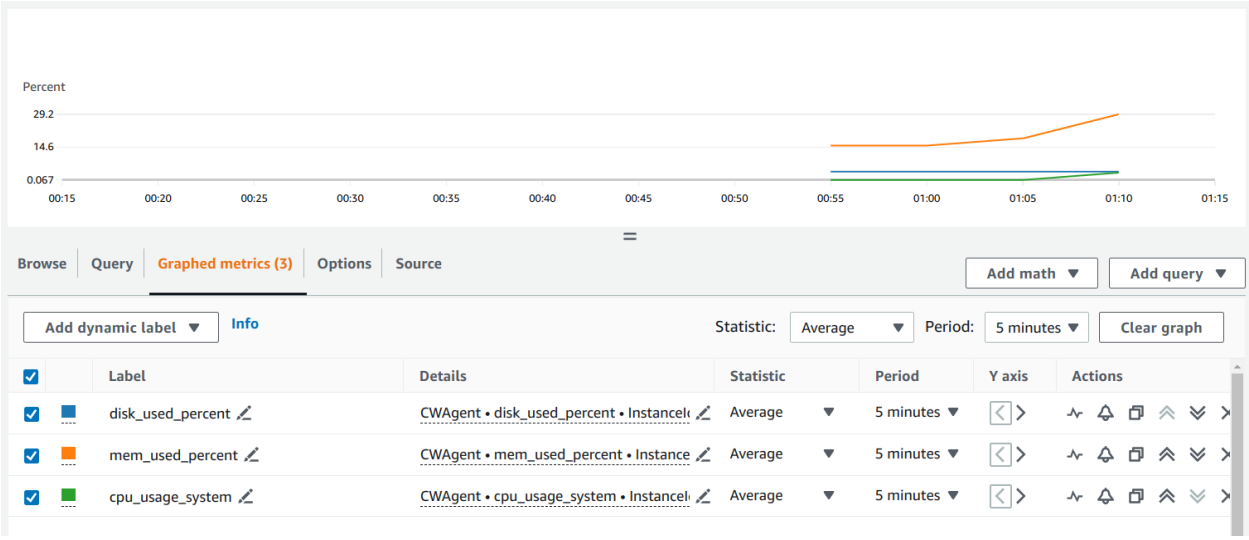
```
[ec2-user@ip-172-31-8-217 ~]$ sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard
=============================================================
= Welcome to the Amazon CloudWatch Agent Configuration Manager =
=                                                             =
= CloudWatch Agent allows you to collect metrics and logs from =
= your host and send them to CloudWatch. Additional CloudWatch =
= charges may apply.                                          =
=============================================================
```

Step 4 - Start the cloudwatch agent after creating the configuration file.

```
[ec2-user@ip-172-31-8-217 ~]$ sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch
ig.json -s
****** processing amazon-cloudwatch-agent ******
I! Trying to detect region from ec2 D! [EC2] Found active network interface Successfully fetched the config and saved in /opt/aws/amazon-cloudwatch-agent/etc
tch-agent.d/file_config.json.tmp
Start configuration validation...
2023/07/11 15:54:51 Reading json config file path: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp ...
2023/07/11 15:54:51 I! Valid Json input schema.
I! Detecting run_as_user...
I! Trying to detect region from ec2
D! [EC2] Found active network interface
No csm configuration found.
No log configuration found.
Configuration validation first phase succeeded
/opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -schematest -config /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.toml
Configuration validation second phase succeeded
Configuration validation succeeded
amazon-cloudwatch-agent has already been stopped
Created symlink /etc/systemd/system/multi-user.target.wants/amazon-cloudwatch-agent.service → /etc/systemd/system/amazon-cloudwatch-agent.service.
```

Step 5 - Open custom namespace "CWAgent"

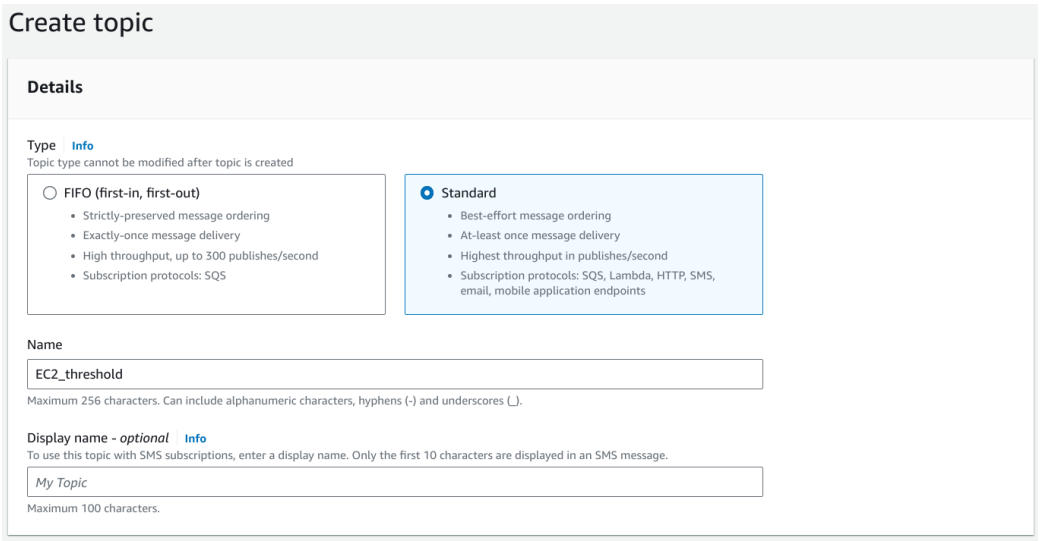




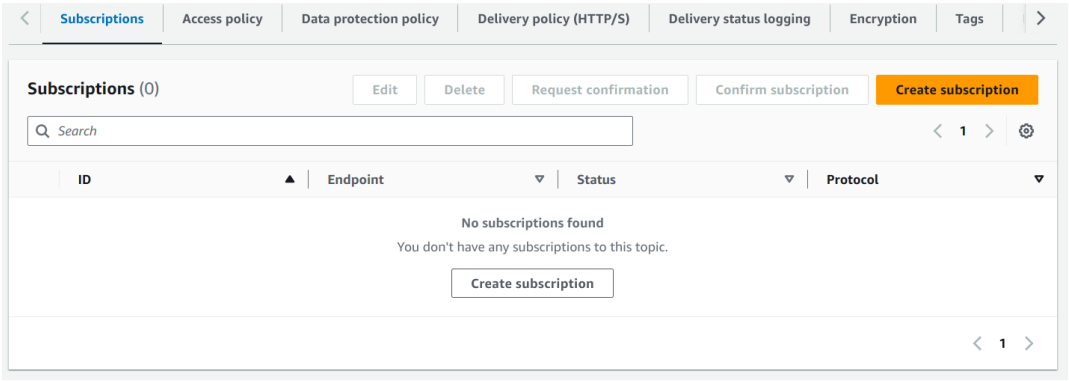
Assignment 3

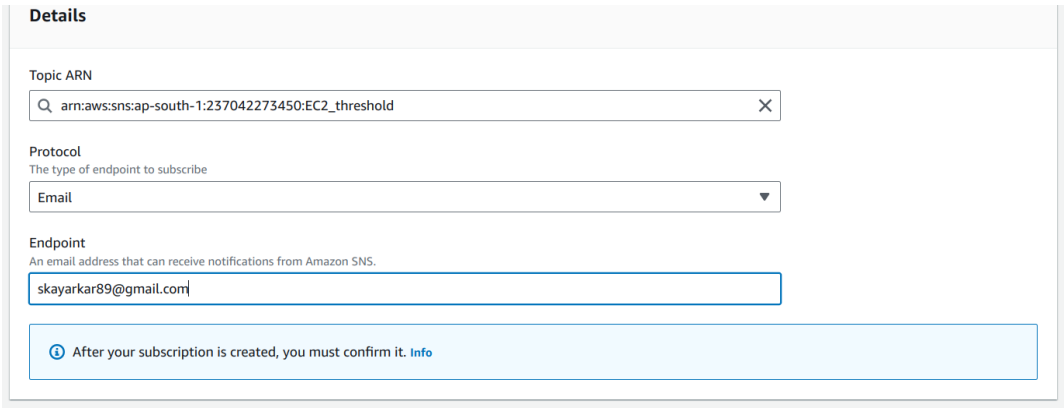
Create an SNS topic and send email notification if metrics of EC2 crosses the 50% threshold.

Step 1 - Browse to the SNS console and create a new topic.

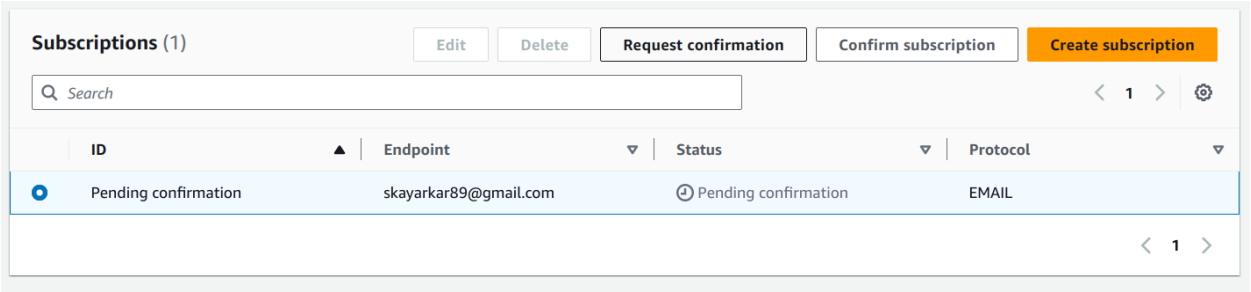


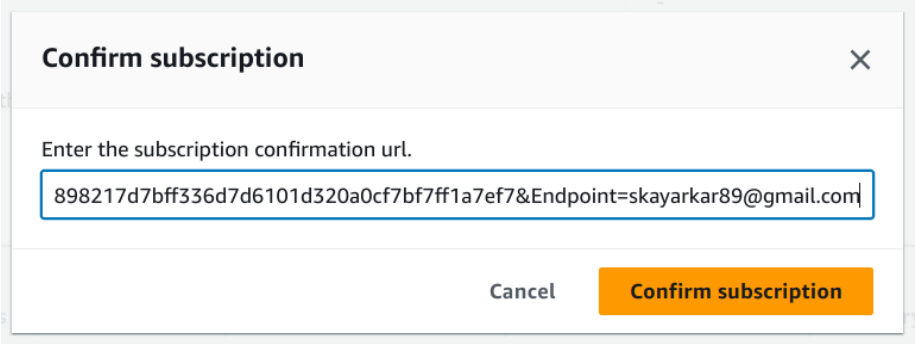
Step 2 - Create a new subscription for the above topic.



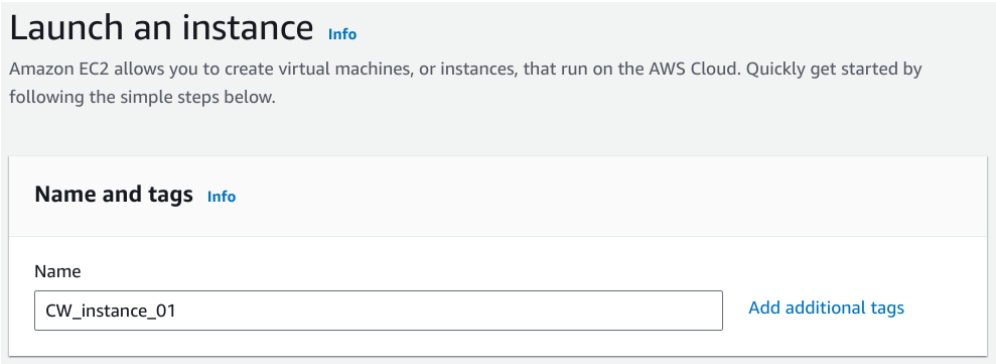


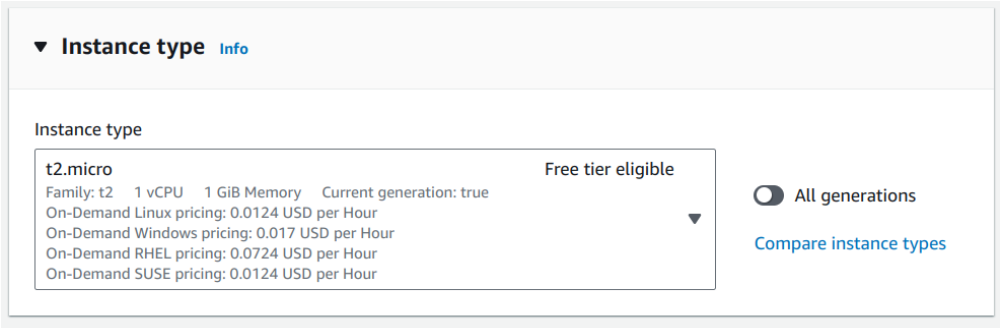
Step 3 - After creating a subscription, click on request confirmation for the subscription and confirm the email received or by copying the URL received in the confirm subscription field.

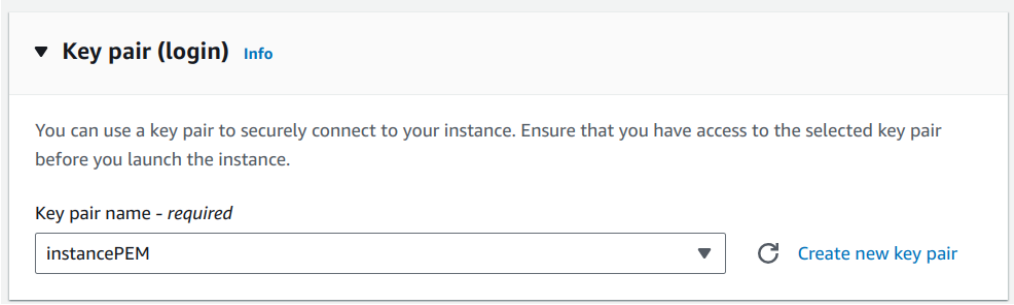


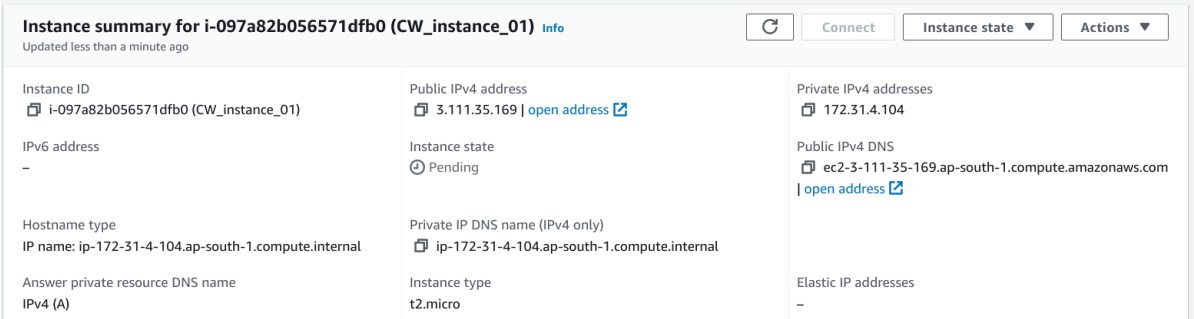


Step 4 - Launch an EC2 instance.

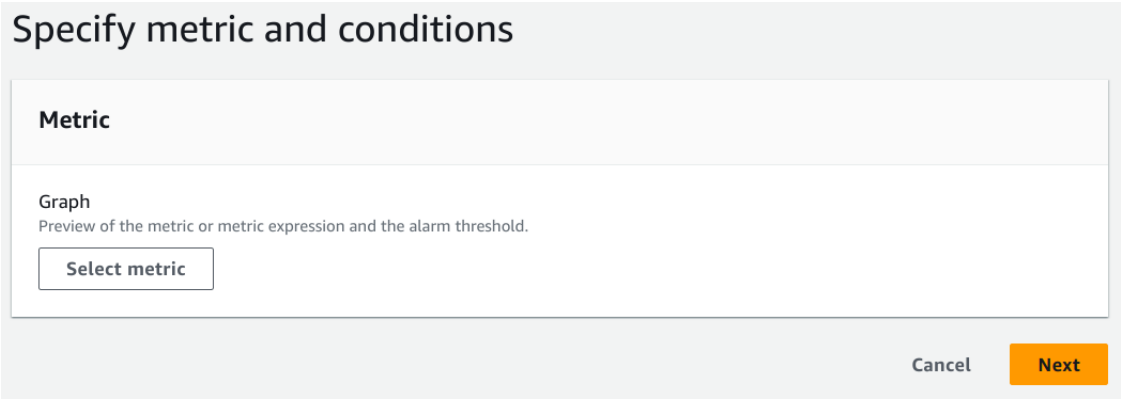




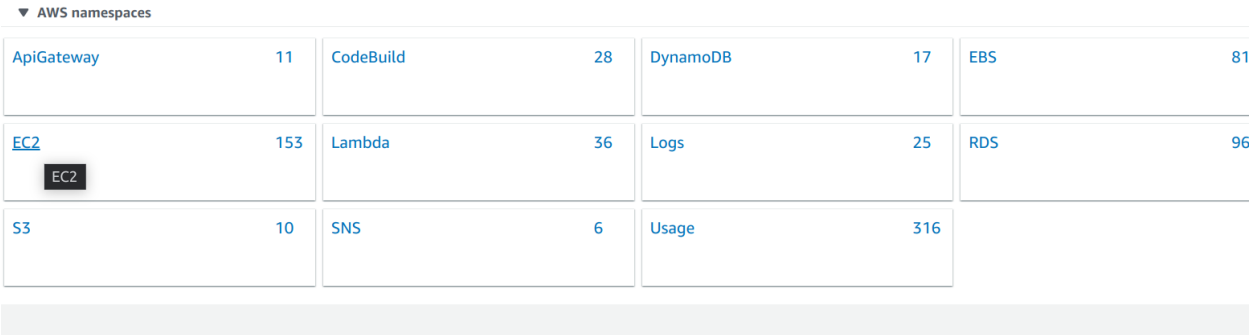




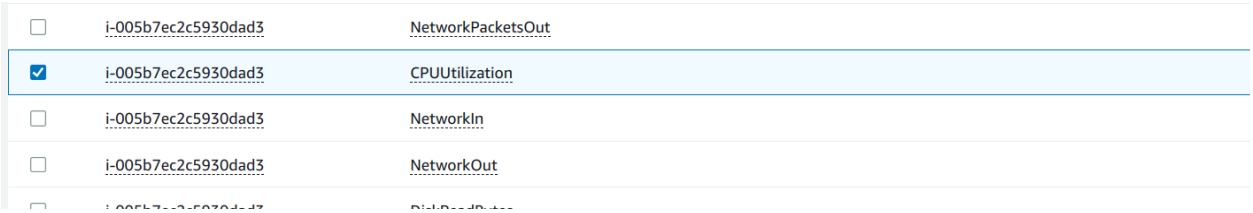
Step 5 - Create an alarm in Cloudwatch for the threshold required for the EC2 instance.



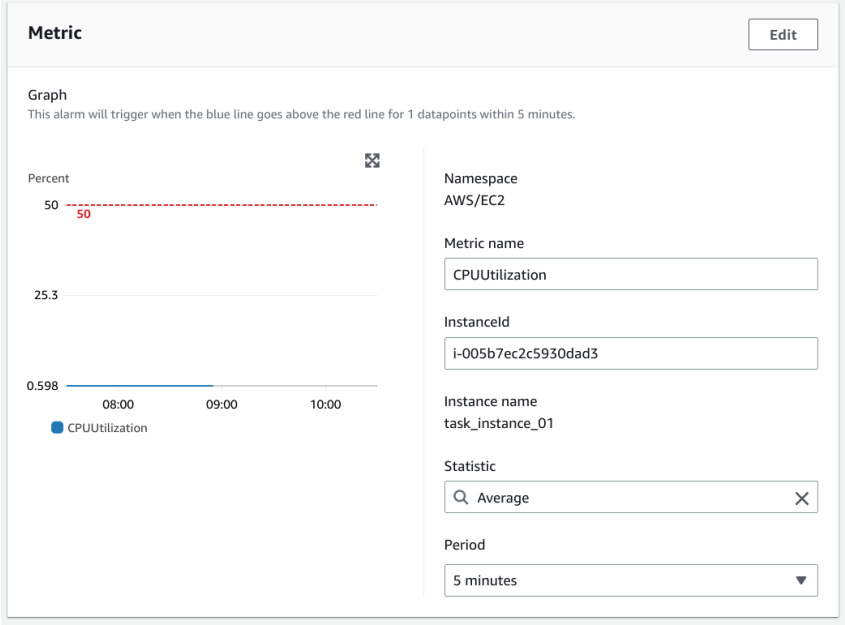
Click on “EC2”.

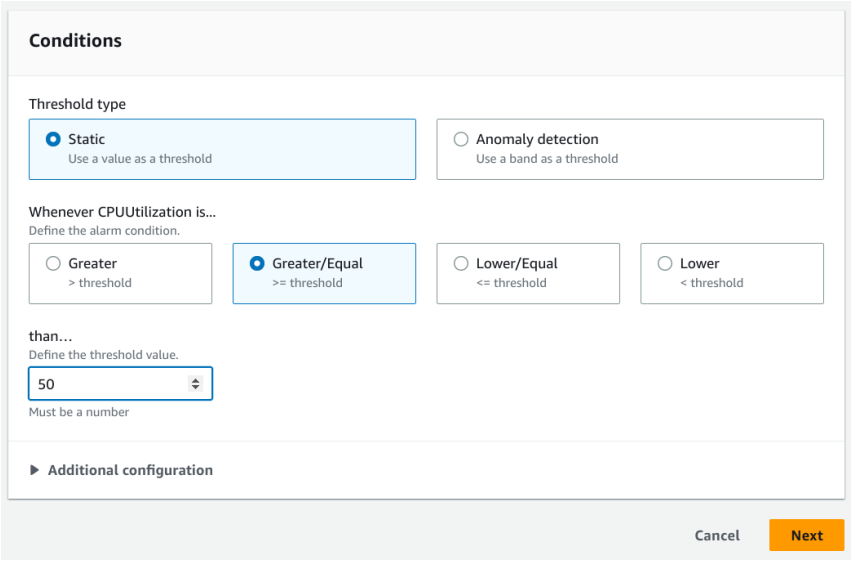


Select the metric for CPU usage.

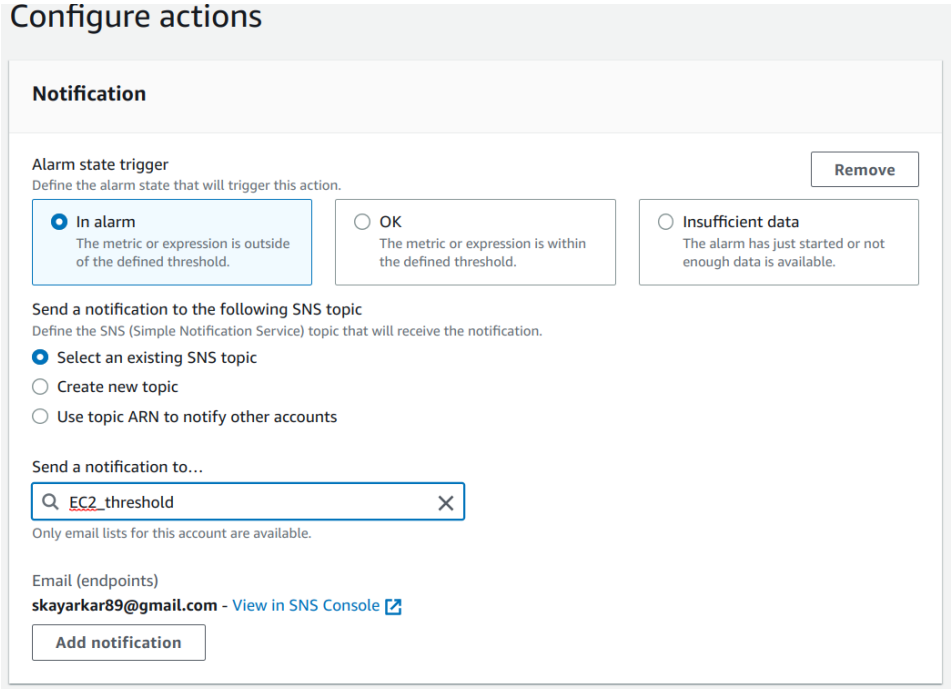


Select threshold value and condition.

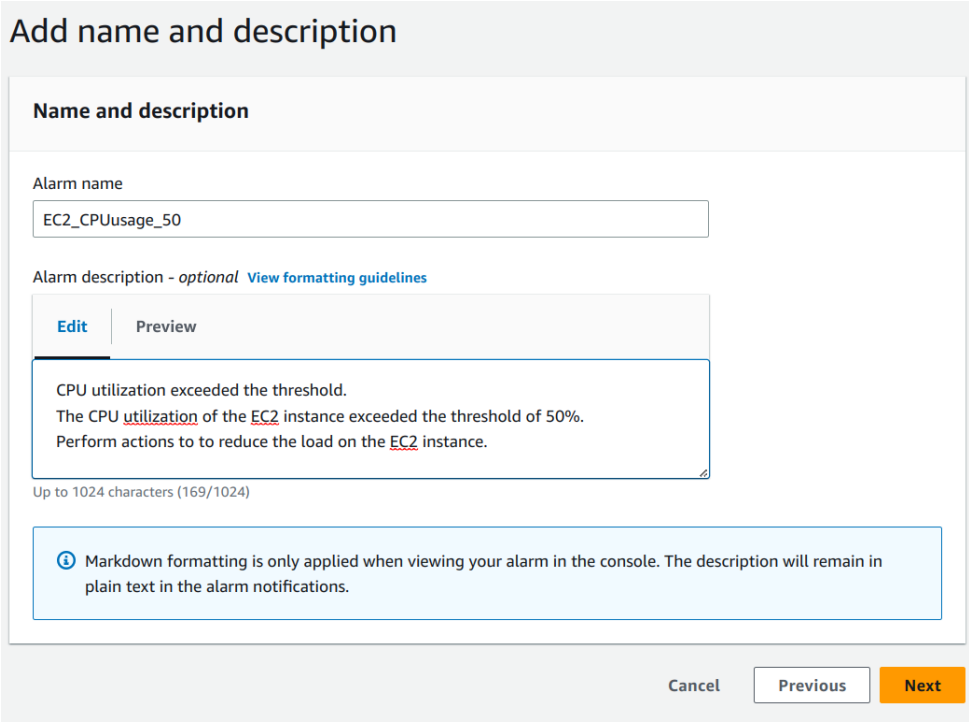




Configure actions for the threshold. Select the SNS topic created in earlier steps.



Enter the name for the alarm and the message to be sent on Email.



Step 6 - Create an alarm for disk usage in the same way.

Preview and create

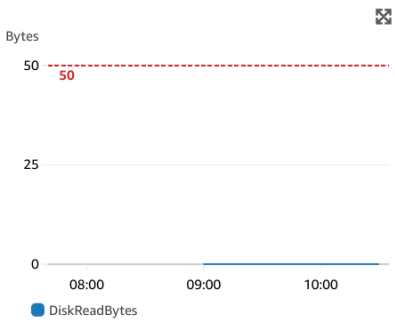
Step 1: Specify metric and conditions

Edit

Metric

Graph

This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.



Namespace
AWS/EC2

Metric name
DiskReadBytes

InstanceId
i-097a82b056571dfb0

Instance name
CW_instance_01

Statistic
Average

Period
5 minutes

Conditions

Threshold type
Static

Whenever **DiskReadBytes** is
Greater/Equal (>=)

than...
50

▶ **Additional configuration**

Step 2: Configure actions

Edit

Actions

Notification
When In alarm, send a notification to "EC2_threshold"

Step 3: Add name and description

Edit

Name and description

Name
EC2_DiskUsage_50

Description
Disk usage exceeded the threshold.
The disk usage of the EC2 instance exceeded the threshold of 50%.

Markdown formatting is only applied when viewing your alarm in the console. The description will remain in plain text in the alarm notifications.

Cancel | Previous | Create alarm

Step 7 - Alarms are in "OK" state initially.

Alarms (2)

Hide Auto Scaling alarms | Clear selection | Create composite alarm | Actions | Create alarm

Search | Any state | Any type | Any actions ... | 1

Name	State	Last state update	Conditions	Actions
EC2_DiskUsage_50	OK	2023-07-12 05:12:22	DiskReadBytes >= 50 for 1 datapoints within 5 minutes	Actions enabled
EC2_CPUusage_50	OK	2023-07-12 05:08:13	CPUUtilization >= 50 for 1 datapoints within 5 minutes	Actions enabled

Step 5 - Run a stress workload on the EC2 instance to increase the CPU utilization and disk usage more than 50%.

```
[ec2-user@ip-172-31-4-104 ~]$ sudo stress --cpu 8 --vm-bytes $(awk '/MemAvailable/{printf "%d\n", $2 * 0.9;}'
< /proc/meminfo)k --vm-keep -m 1
stress: info: [28452] dispatching hogs: 8 cpu, 0 io, 1 vm, 0 hdd
```

Step 6 - Email notification received for CPU usage exceeding 50% of threshold.

ALARM: "EC2_CPUusage_50" in Asia Pacific (Mumbai) Inbox ×

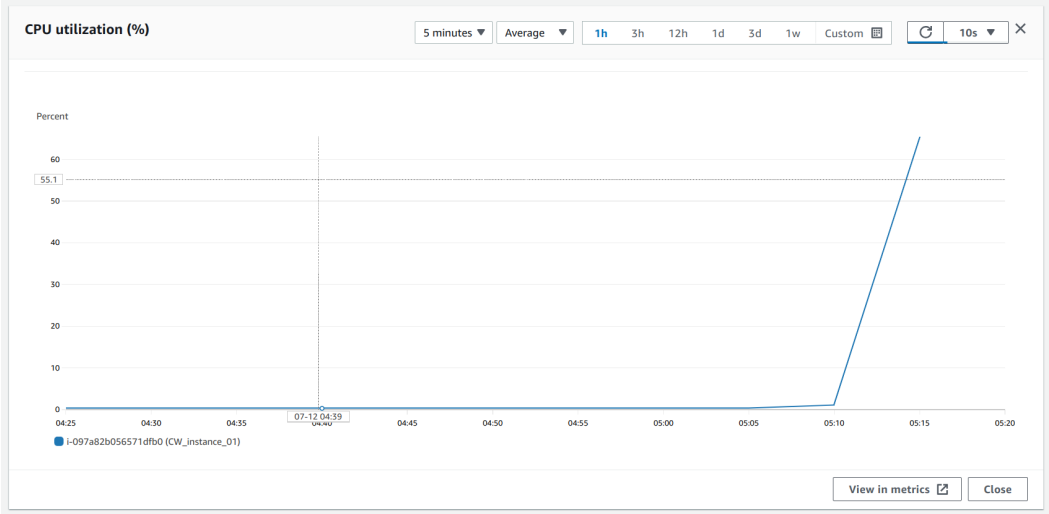
AWS Notifications <no-reply@sns.amazonaws.com> 10:52 AM (2 minutes ago)
to me

You are receiving this email because your Amazon CloudWatch Alarm "EC2_CPUusage_50" in the Asia Pacific (Mumbai) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [65.40591831064185 (12/07/23 05:17:00)] was greater than or equal to the threshold (50.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Wednesday 12 July, 2023 05:22:13 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/EC2_CPUusage_50

Alarm Details:
- Name: EC2_CPUusage_50
- Description: CPU utilization exceeded the threshold.
The CPU utilization of the EC2 instance exceeded the threshold of 50%.
Perform actions to to reduce the load on the EC2 instance.
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [65.40591831064185 (12/07/23 05:17:00)] was greater than or equal to the threshold (50.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Wednesday 12 July, 2023 05:22:13 UTC
- AWS Account: 237042273450
- Alarm Arn: arn:aws:cloudwatch:ap-south-1:237042273450:alarm:EC2_CPUusage_50

Name	State	Last state update	Conditions	Actions
EC2_CPUusage_50	In alarm	2023-07-12 05:22:13	CPUUtilization >= 50 for 1 datapoints within 5 minutes	Actions enabled


CPU utilization (%)
5 minutes
Average
1h
3h
12h
1d
3d
1w
Custom
10s
Percent
60
55.1
50
40
30
20
10
0
04:25
04:30
04:35
07-12 04:39
04:45
04:50
04:55
05:00
05:05
05:10
05:15
05:20
i-097a82b056571dfb0 (CW_instance_01)
View in metrics
Close