

SQS,SNS,CW,CT

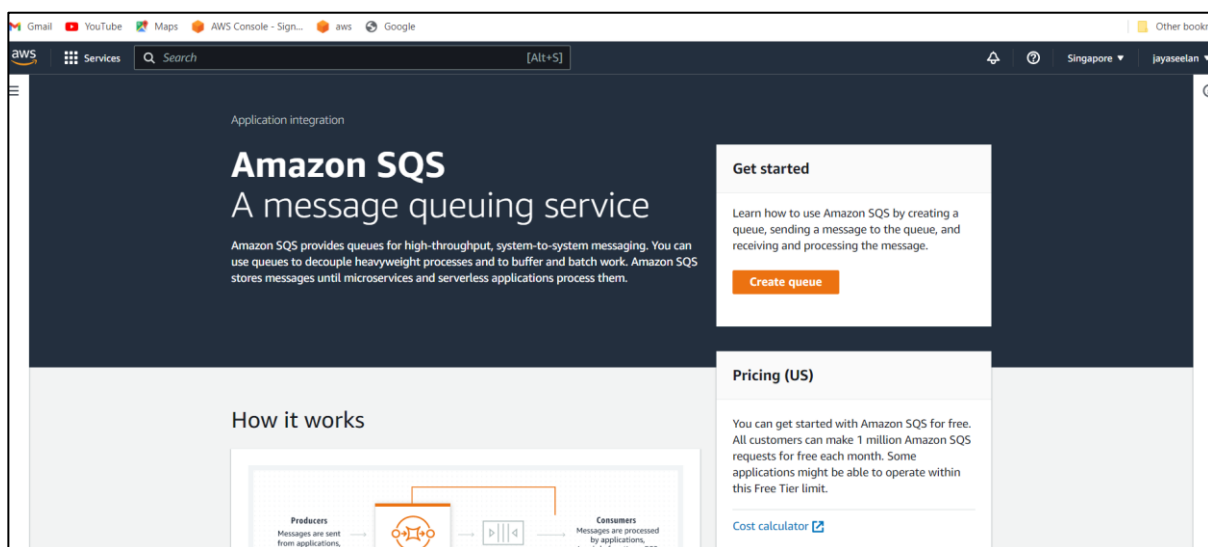
SQS

Simple queue service

Amazon Simple Queue Service (SQS) lets you send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available.

Steps to create SQS

Step1.1:create queue



Step1.2:select type---->standard---->name(any)

Type

Choose the queue type for your application or cloud infrastructure.

You can't change the queue type after you create a queue.

☒ **Standard** Info
At-least-once delivery, message ordering isn't preserved

- At-least once delivery
- Best-effort ordering

☐ **FIFO** Info
First-in-first-out delivery, message ordering is preserved

- First-in-first-out delivery
- Exactly-once processing

Name

mysqls

A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).

Step1.3:configuration

Configuration

Set the maximum message size, visibility to other consumers, and message retention. [Info](#)

Visibility timeout [Info](#)

Seconds ▼

Should be between 0 seconds and 12 hours.

Message retention period [Info](#)

Days ▼

Should be between 1 minute and 14 days.

Delivery delay [Info](#)

Seconds ▼

Should be between 0 seconds and 15 minutes.

Maximum message size [Info](#)

KB

Should be between 1 KB and 256 KB.

Receive message wait time [Info](#)

Seconds

Should be between 0 and 20 seconds.

Step1.4:create queue

1:795071115491:

► Redrive allow policy - *Optional*

Identify which source queues can use this queue as the dead-letter queue. [Info](#)

► Dead-letter queue - *Optional*

Send undeliverable messages to a dead-letter queue. [Info](#)

► Tags - *Optional*

A tag is a label assigned to an AWS resource. Use tags to search and filter your resources or track your AWS costs. [Learn more](#)

Cancel

Create queue

aws Services Search [Alt+S] Singapore jayaseelan

Amazon SQS > Queues

Queues (1)

Search queues by prefix

< 1 >

Name	Type	Created	Messages available	Messages in flight	Encryption	Content-based deduplication
mysqs	Standard	Jan 04, 2023, 12:35:58 GMT+5:30	0	0	Amazon SQS key (SSE-SQS)	-

Edit Delete Send and receive messages Actions

Create queue

queue created...

Step2:send and received message---->message body(any type)---->send message

Send message [Info](#) Clear content Send message

Message body
Enter the message to send to the queue.

hi this is jayasellan

Delivery delay [Info](#)

0 Seconds ▼

Should be between 0 seconds and 15 minutes.

message has been sent

Send message [Info](#) Clear content Send message

✓ Your message has been sent and is ready to be received. View details ×

Message body
Enter the message to send to the queue.

hi this is jayasellan

Delivery delay [Info](#)

Step2.1: Poll For Message Click--->Message Will Received

Receive messages [Info](#) Edit poll settings Stop polling Poll for messages

Messages available: 0 | Polling duration: 30 | Maximum message count: 10 | Polling progress: 1 receives/second

Messages (1) View details Delete

Search messages

<input type="checkbox"/>	ID	Sent	Size	Receive count
<input type="checkbox"/>	216a1444-b3b7-4a1d-8f8d-4e3158361dc0	Jan 04, 2023, 12:49:30 GMT+5:30	21 bytes	1

Looking for language selection? Find it in the new [Unified Settings](#)

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Click id --->shown all details

Message: 216a1444-b3b7-4a1d-8f8d-4e3158361dc0 ×

Details **Body** **Attributes**

hi this is jayaseelan

Done

SNS

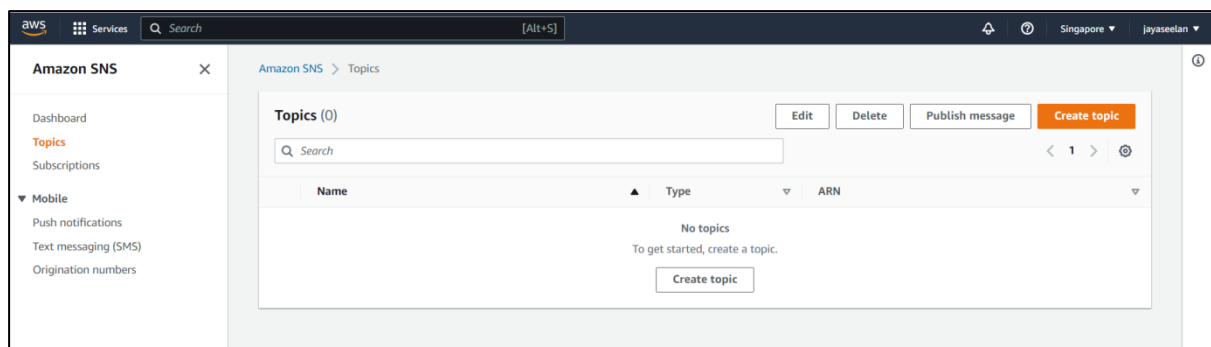
Simple Notification Service

Amazon Simple Notification Service (Amazon SNS) is a managed service that provides message delivery from publishers to subscribers (also known as *producers* and *consumers*).

Publishers communicate asynchronously with subscribers by sending messages to a *topic*, which is a logical access point and communication channel.

Clients can subscribe to the SNS topic and receive published messages using a supported endpoint type, such as Amazon Kinesis Data Firehose, Amazon SQS, AWS Lambda, HTTP, email, mobile push notifications, and mobile text messages (SMS).

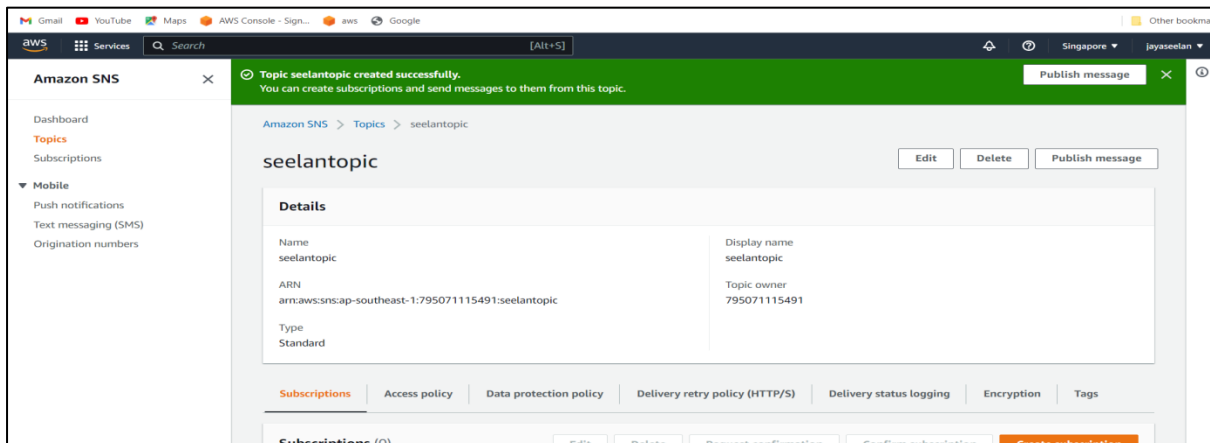
Step1:Create Topic(Group)



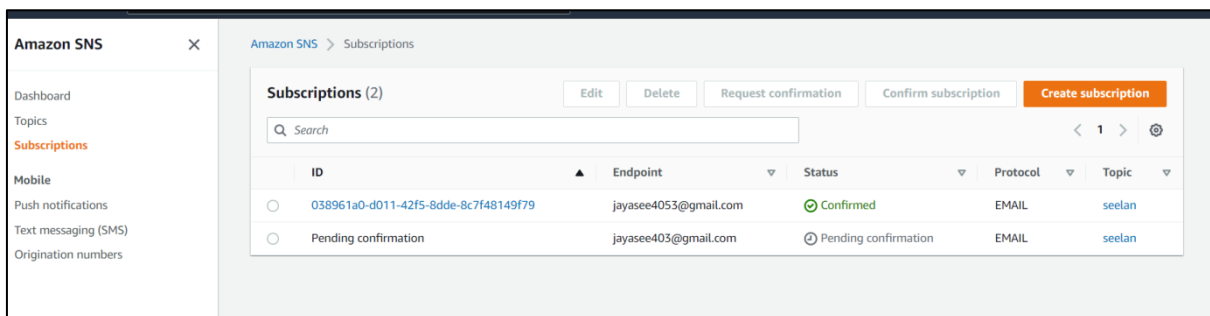
Step1.1:details----->select type(standard)--->name(any)--->display name(any)----->create topic

A screenshot of the 'Create topic' form in the Amazon SNS console. The breadcrumb trail at the top reads 'Amazon SNS > Topics > Create topic'. The main heading is 'Create topic'. Under the 'Details' section, there are two tabs: 'Type' and 'Info'. The 'Type' tab is selected. Below it, a note states 'Topic type cannot be modified after topic is created'. There are two radio button options: 'FIFO (first-in, first-out)' and 'Standard'. The 'Standard' option is selected. The 'FIFO' option includes bullet points: 'Strictly-preserved message ordering', 'Exactly-once message delivery', 'High throughput, up to 300 publishes/second', and 'Subscription protocols: SQS'. The 'Standard' option includes bullet points: 'Best-effort message ordering', 'At-least once message delivery', 'Highest throughput in publishes/second', and 'Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints'. Below the type selection, there is a 'Name' field with the text 'seelan topic' and a note: 'Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).'. At the bottom, there is a 'Display name - optional' field with the text 'seelan topic' and a note: 'To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message. Maximum 100 characters.'

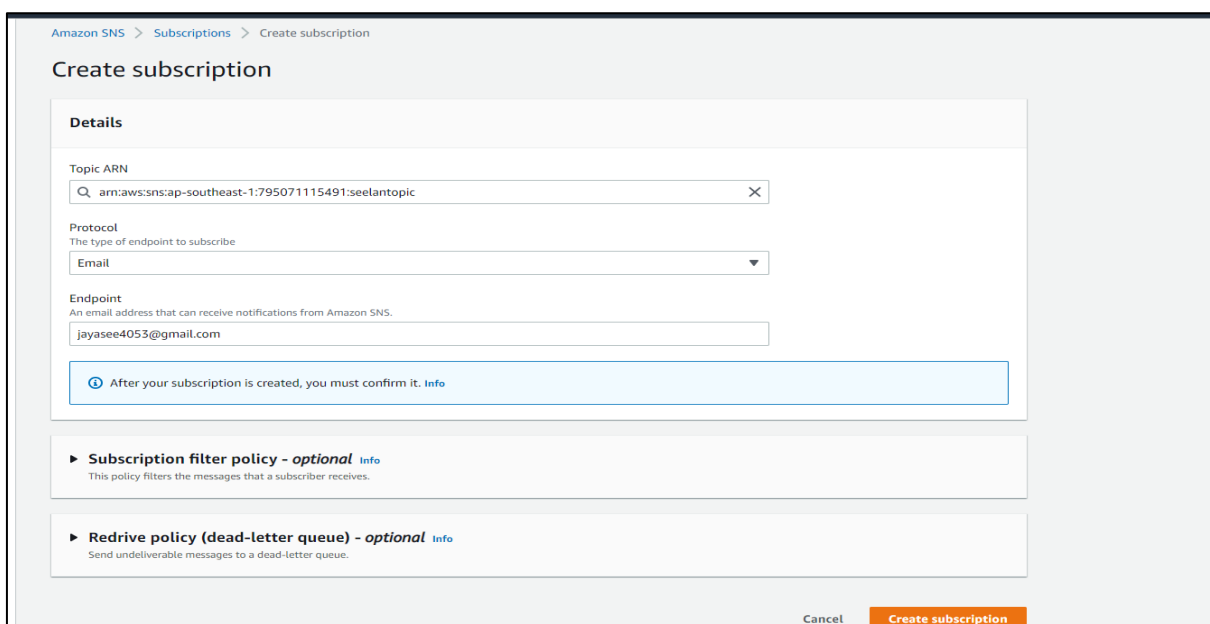
Topic created..



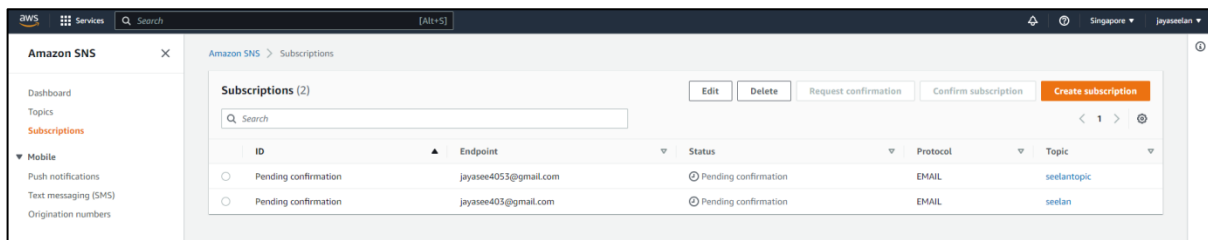
Step2:Create Subscribtion(members)



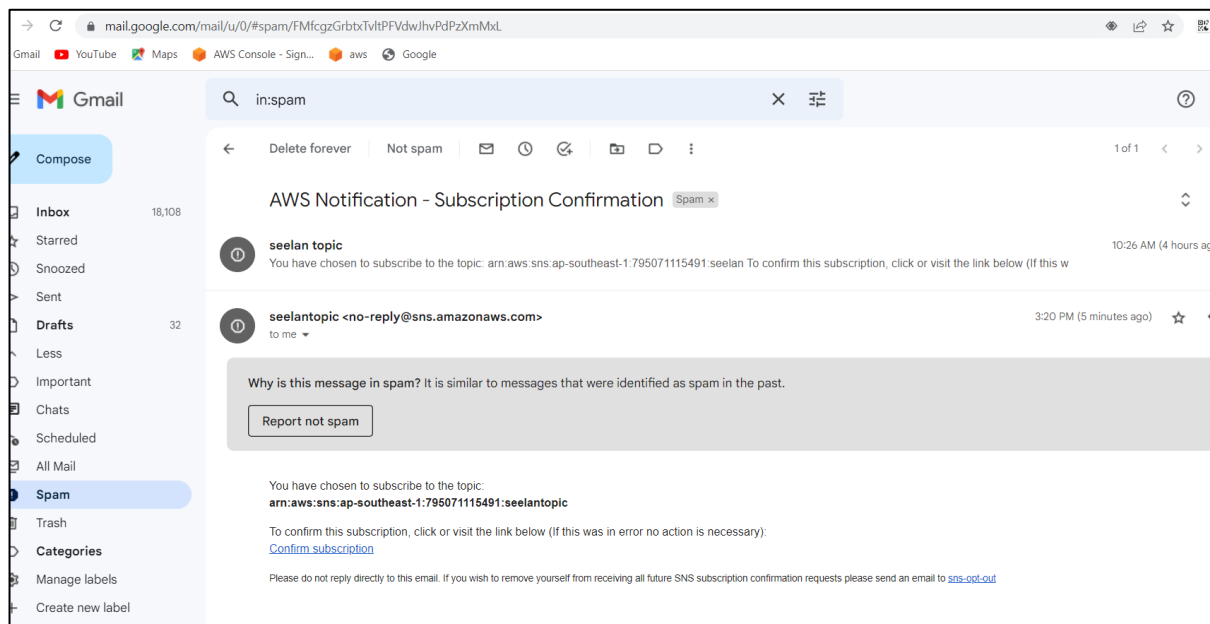
Step2.1:select topic---->protocol(email)----->endpoint(mail id)---->create subscription



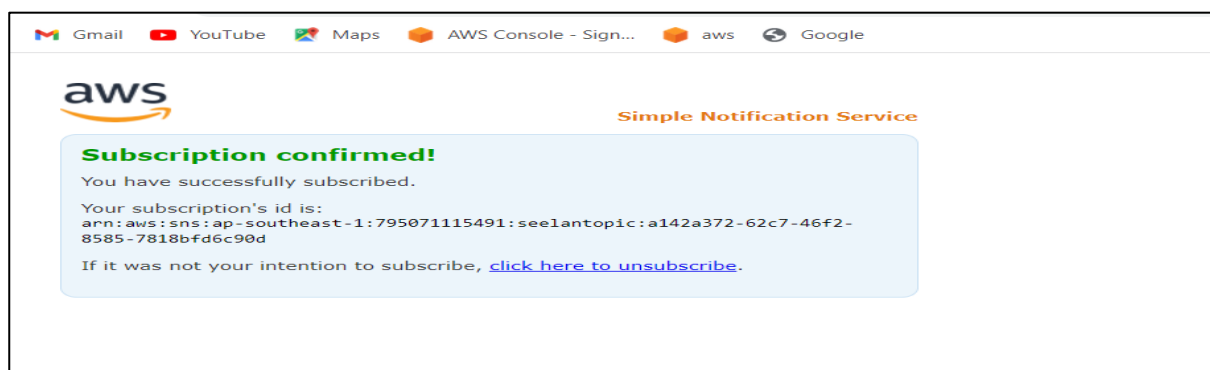
Subscription created..



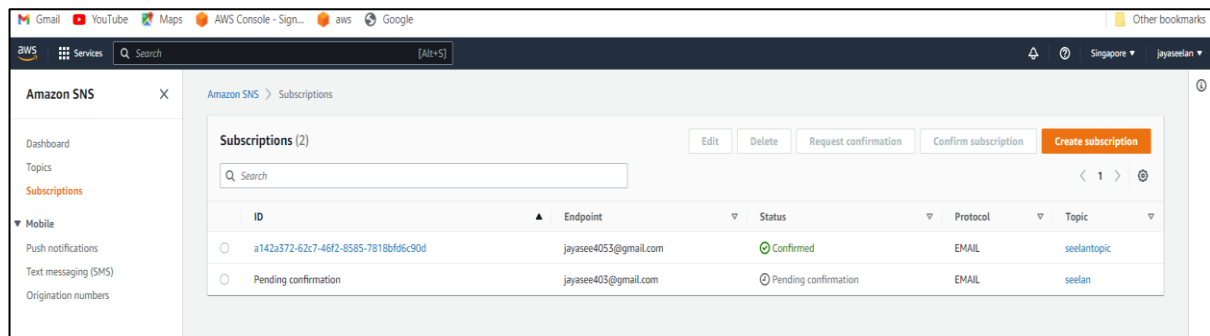
Step2.2:confirmation subscription---->subscription status(change for confirmed)----->given mail go---->spam msg---->click --->aws msg received(click)----->confirm subsribtion--->status will change confirmed.



Subscription confirmed....



Now check subscription status(confirmed..)

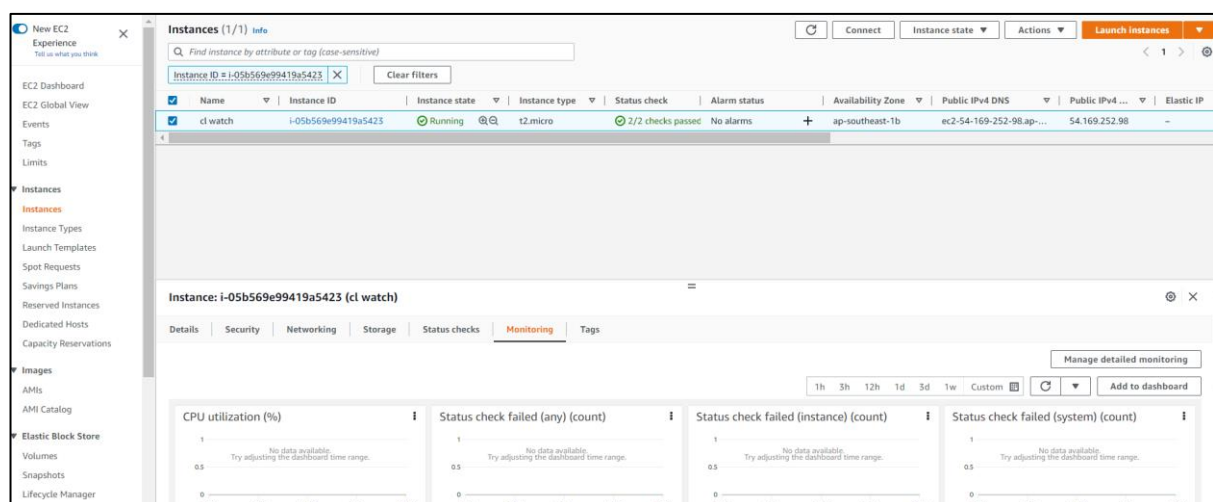


CLOUDWATCH

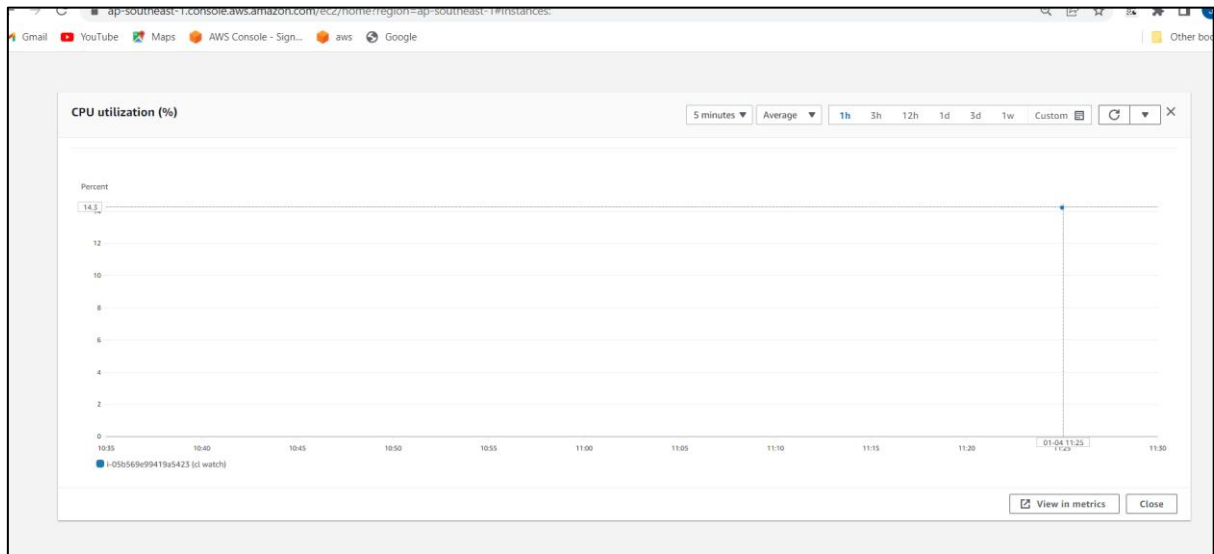
CloudWatch enables you to monitor your complete stack (applications, infrastructure, and services) and use alarms, logs, and events data to take automated actions and reduce mean time to resolution (MTTR).

This frees up important resources and allows you to focus on building applications and business value.

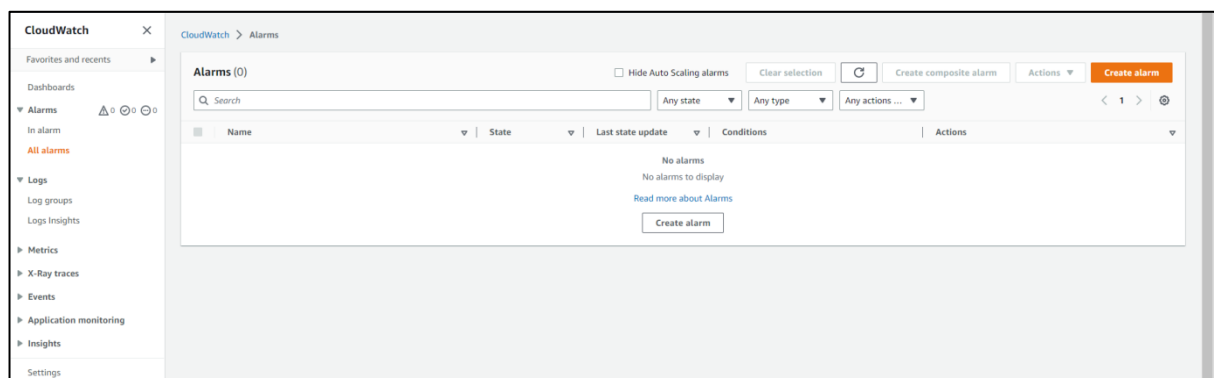
Step1:Create Instance---->cpu utilization--->check cpu utilization ratio



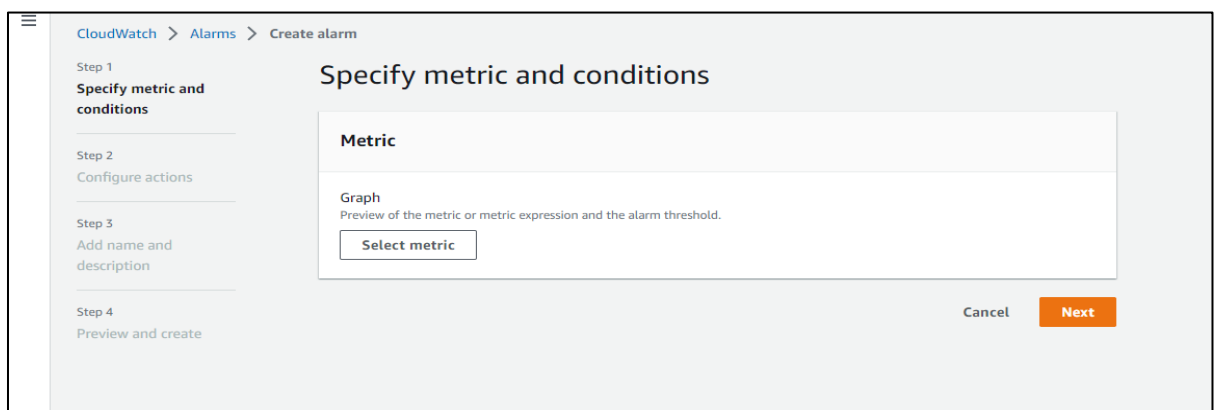
cpu utilization ratio=14.2



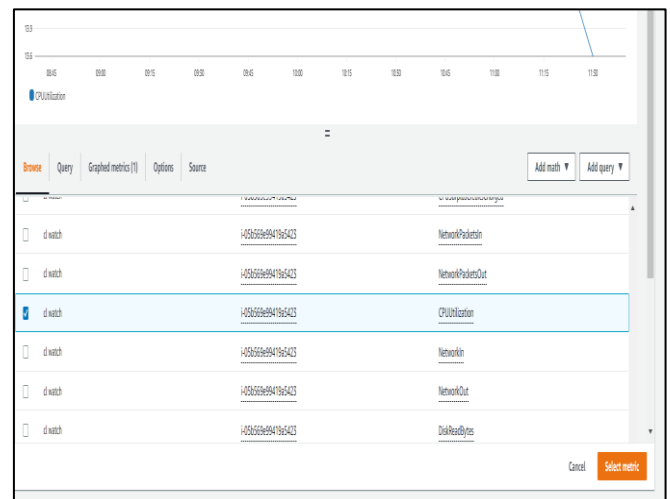
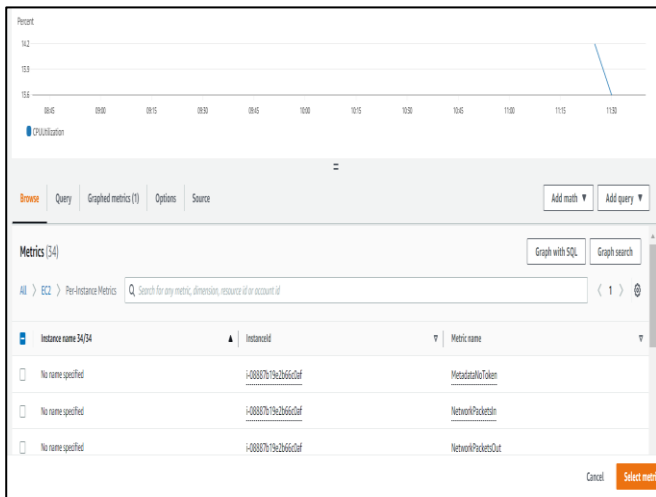
Step2:Cloud Watch--->All Alarms---->All Alarms--->Create Alarm



Step2.1:select metric--->ec2---> pre-instance metrics



select instance cpu utilization---->select metrix



Step2.2:Condition---->Threshold Type(Static)---->Lower/Equal---->Than(Cpu Utilization 14 So Give 10)---->Next

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.

10

Must be a number

► Additional configuration

Cancel Next

Step2.3:Configuration Action---->Notification---->In Alarm---->Select Existing Sns Topic--->Select Topic

Configure actions

Notification

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.

[Remove](#)

Send a notification to the following 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 to notify other accounts

Send a notification to...

Only email lists for this account are available.

Email (endpoints)
jayasee4053@gmail.com - [View in SNS Console](#)

[Add notification](#)

Step2.4:Ec2 Action---->In Alaram--->Take Following Action(Any One Select)--->Stop This Instance--->Next.

EC2 action

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.

[Remove](#)

Take the following action...
Define what will happen to the EC2 instance with the Instance ID i-05b569e99419a5423 when this alarm is triggered.

☐ Recover this instance
You can only recover certain EC2 instance types. [See documentation](#)

☒ **Stop this instance**
You can only stop an instance if it is backed by an EBS volume. AWS will use the existing Service Linked Role (AWSServiceRoleForCloudWatchEvents) to perform this action. [Show IAM policy document](#)

☐ Terminate this instance
You will not be able to terminate this instance if termination protection is enabled. AWS will use the existing Service Linked Role (AWSServiceRoleForCloudWatchEvents) to perform this action. [Show IAM policy document](#)

☐ Reboot this instance
An instance reboot is equivalent to an operating system reboot. AWS will use the existing Service Linked Role (AWSServiceRoleForCloudWatchEvents) to perform this action. [Show IAM policy document](#)

[Add EC2 action](#)

Step2.5:Add Name & Description---->Alaram Name(Any)---->Alaram Description(Any Content Type)---->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
my alarammmm

Alarm description - optional [View formatting guidelines](#)

Edit **Preview**

this cpu will increase so instal stop

Up to 1024 characters (37/1024)

ⓘ 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** **Next**

Step2.6:preview and create---->create 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

Preview and create

Step 1: Specify metric and conditions **Edit**

Metric

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

Percent

90

80

70

60

50

40

30

20

10

00:00 10:00 11:00

CPUUtilization

Namespace: AWS/EC2
Metric name: CPUUtilization
InstanceId: i-05a569a994119a5423
Instance name: ci-ec2-01
Statistic: Average
Period: 5 minutes

Conditions

Threshold type: Static
Whenever CPUUtilization is Lower/Equal (<=) than... 90

Additional configuration

Step 2: Configure actions **Edit**

Actions

Notification
When in alarm, send a notification to "awslogs-01"

EC2 action
When in alarm, stop this instance (Instance ID: i-05a569a994119a5423)

Step 3: Add name and description **Edit**

Name and description

Name: my alarammmm
Description: this cpu will increase so instal stop

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

Cancel **Previous** **Create alarm**

Alaram create ---->wait for state(insufficient data to alarm)

aws Services Search [Alt+S] Singapore jaysseelan

CloudWatch X Successfully created alarm my alarammmm. View alarm X

CloudWatch > Alarms

Alarms (1/1) ☐ Hide Auto Scaling alarms Clear selection Create composite alarm Actions Create alarm

Search Any state Any type Any actions ...

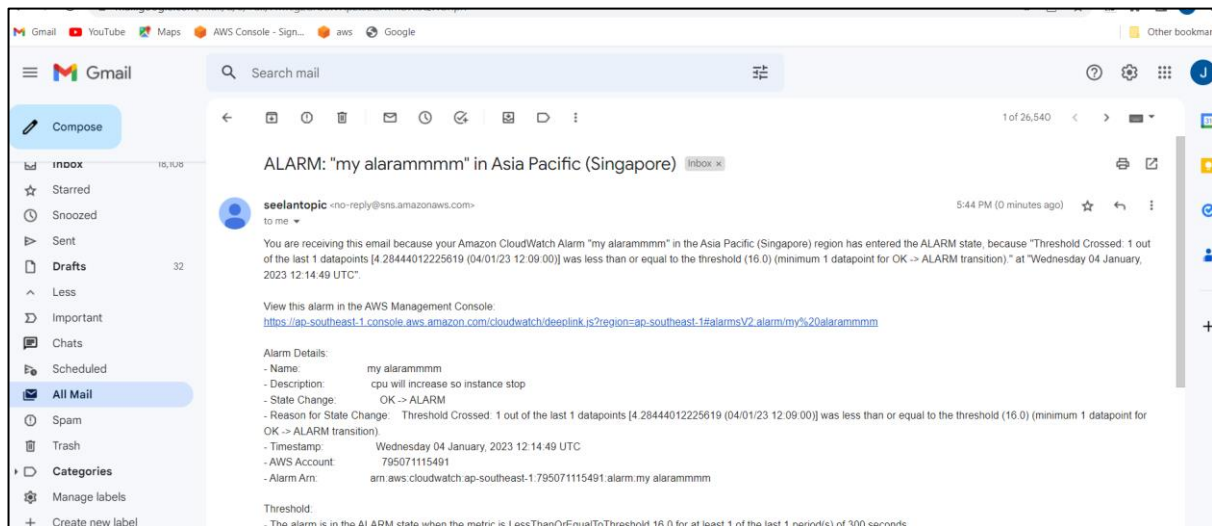
<input checked="" type="checkbox"/>	Name	State	Last state update	Conditions	Actions
<input checked="" type="checkbox"/>	my alarammmm	Insufficient data	2023-01-04 17:42:30	CPUUtilization <= 16 for 1 datapoints within 5 minutes	Actions enabled

CloudWatch > Alarms

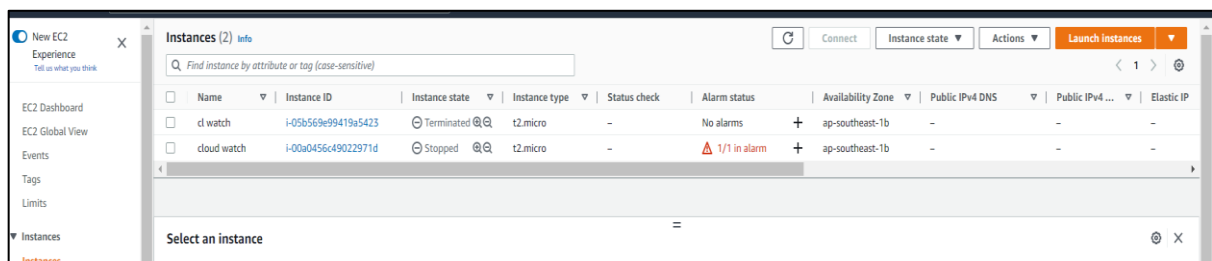
Alarms (1/1)

Search Any state Any type Any actions ...

<input checked="" type="checkbox"/>	Name	State	Last state update	Conditions	Actions
<input checked="" type="checkbox"/>	my alarammmm	Insufficient data	2023-01-04 17:42:30	CPUUtilization <= 16 for 1 datapoints within 5 minutes	Actions enabled



instance will stoped..

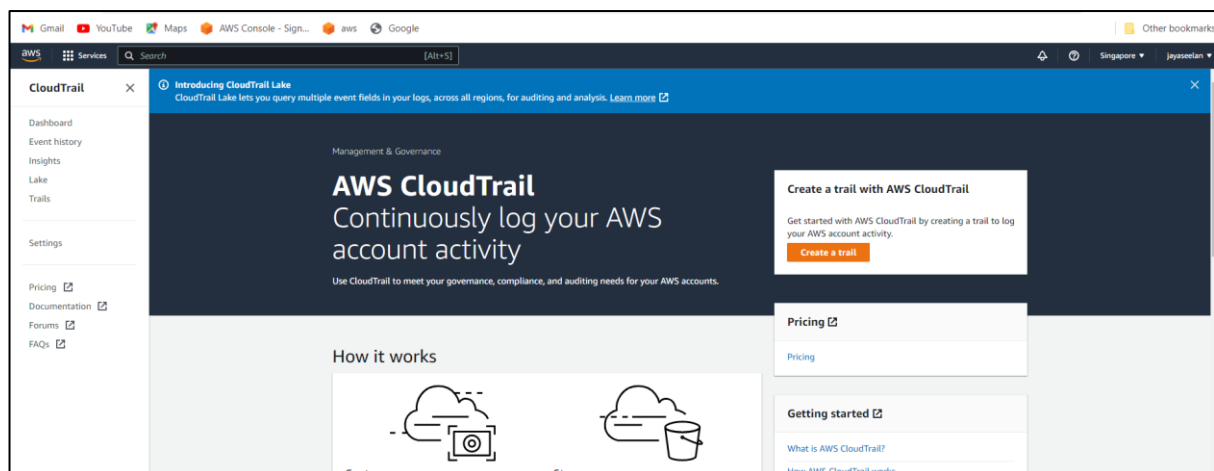


conditions apply ok...

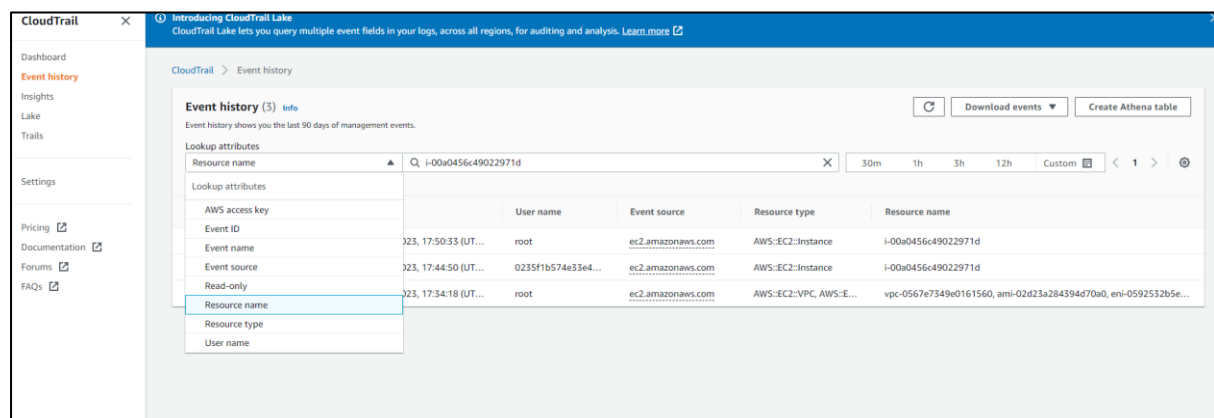
CLOUD TRIAL:

CloudTrail enables auditing, security monitoring, and operational troubleshooting by tracking user activity and API usage.

CloudTrail logs, continuously monitors, and retains account activity related to actions across your AWS infrastructure, giving you control over storage, analysis, and remediation actions.



Step1:Event History--->Resource Name--->Instance Ip Copy And Put---->It Show Log History



Detail saw histry click--->history name--->show full history