

✦ Simple Notification Service (Amazon SNS) is a web service that coordinates and manages the delivery or sending of messages to subscribing endpoints or clients.

#### ✕ Components

✦ EC2 Instance - An Amazon EC2 instance is a virtual server in Amazon's Elastic Compute Cloud (EC2) for running applications on the Amazon Web Services (AWS) infrastructure.

✦ - Amazon CloudWatch monitors your Amazon Web Services (AWS) resources and the applications you run on AWS in real time.

✕ In the below hands-on steps, we are going to create an ec2 instance, one cloudwatch alarm and one SNS topic . So here we want to show when the cpu utilization of the server goes beyond 40%, the ec2 instance should be stopped and we should receive an email in the registered email ID.

✕ Create an EC2 instance with all default settings except on Network settings. Follow below to setup network settings:

## ▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0119ae136d2e37942 (Default VPC)  
172.31.0.0/16

(default) ▼



Subnet [Info](#)

No preference ▼



[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable ▼

### Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - *required*

sns-sg

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ., -, /, !, @, #, \$, %, &, '.

Description - *required* [Info](#)

sns-sg

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type [Info](#)

ssh ▼

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere ▼

Source [Info](#)

Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - *optional* [Info](#)

e.g. SSH for admin desktop

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type [Info](#)

HTTP ▼

Protocol [Info](#)

TCP

Port range [Info](#)

80

Source type [Info](#)

Anywhere ▼

Source [Info](#)

Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - *optional* [Info](#)

e.g. SSH for admin desktop

▶ Security group rule 3 (ICMP, All)

Remove

▼ Security group rule 4 (ICMP, All, 0.0.0.0/0)

Remove

Type [Info](#)

All ICMP - IPv4 ▼

Protocol [Info](#)

ICMP

Port range [Info](#)

All

Source type [Info](#)

Anywhere ▼

Source [Info](#)

Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - *optional* [Info](#)

e.g. SSH for admin desktop

✗ Now we are going to create SNS. Search for Amazon Simple Notification Service and give a topic to the text area as below:

The screenshot shows the 'Create topic' page in the Amazon SNS console. On the left, there's a header for 'Amazon Simple Notification Service' with the tagline 'Pub/sub messaging for microservices and serverless applications.' Below this is a brief description of SNS. On the right, the 'Create topic' form is visible. It has a 'Topic name' field with the value 'ec2-server-alert' and a 'Next step' button. A link 'Start with an overview' is also present.

✗ Under Details, we're selecting standard type as FIFO is strictly for message delivering.

The screenshot shows the 'Details' section of the 'Create topic' page. It features two radio button options for 'Type': 'FIFO (first-in, first-out)' and 'Standard'. The 'Standard' option is selected. Below the type selection, there are two text input fields: 'Name' and 'Display name - optional'. Both fields contain the text 'ec2-server-alert'. The 'Name' field has a note: 'Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (\_).' The 'Display name' field has 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.'

✗ Delivery status logging - click on Create new service role and create new role. We will get a new tab for the IAM console.

▼ **Delivery status logging - optional** [Info](#)

These settings configure the logging of message delivery status to CloudWatch Logs.

Log delivery status for these protocols

- ☐ AWS Lambda
- ☐ Amazon SQS
- ☐ HTTP/S
- ☐ Platform application endpoint
- ☐ Amazon Kinesis Data Firehose

Success sample rate

The percentage of successful message deliveries to log.

%

**IAM roles**

Amazon SNS requires permission to write logs to CloudWatch Logs. You can use separate roles for successful and failed message deliveries.

Service role [Info](#)

☐ Use existing service role  
Choose an existing service role from your account.

☒ Create new service role  
Create a new service role in your account.

[Create new roles](#)

IAM role for successful deliveries

-

IAM role for failed deliveries

-

✗ IAM Management Console - we're going with the default settings and click on allow.

▼ Hide Details

Role Summary [?](#)

Role Description Provides write access to AWS Services and Resources

IAM Role [Create a new IAM Role](#) ▼

Role Name

▼ Hide Policy Document

[Edit](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:PutMetricFilter"
      ]
    }
  ]
}
```

Role Summary [?](#)

Role Description Provides write access to AWS Services and Resources

IAM Role [Create a new IAM Role](#) ▼

Role Name

► View Policy Document

✗ All the other settings are set be default in SNS.Click on create subscription

Amazon SNS > Topics > ec2-server-alert

## ec2-server-alert

Edit Delete Publish message

**Details**

Name ec2-server-alert	Display name ec2-server-alert
ARN arn:aws:sns:us-east-2:526211996142:ec2-server-alert	Topic owner 526211996142
Type Standard	

Subscriptions Access policy Data protection policy Delivery policy (HTTP/5) Delivery status logging Encryption Tags Integrations

**Subscriptions (0)** Edit Delete Request confirmation Confirm subscription **Create subscription**

Q Search

ID	Endpoint	Status	Protocol
No subscriptions found You don't have any subscriptions to this topic.			

Create subscription

< 1 >

✖ Now we need to enter the email ID which we need to receive notifications.

Amazon SNS > Subscriptions > Create subscription

## Create subscription

**Details**

**Topic ARN**  
Q arn:aws:sns:us-east-2:526211996142:ec2-server-alert X

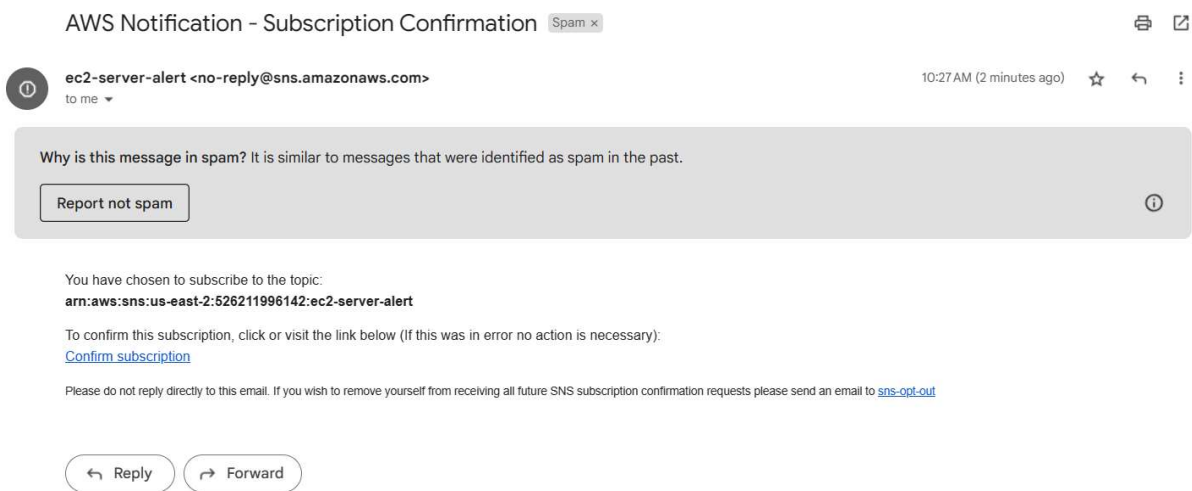
**Protocol**  
The type of endpoint to subscribe:  
Email ▼

**Endpoint**  
An email address that can receive notifications from Amazon SNS.  
exampleaws2023@gmail.com

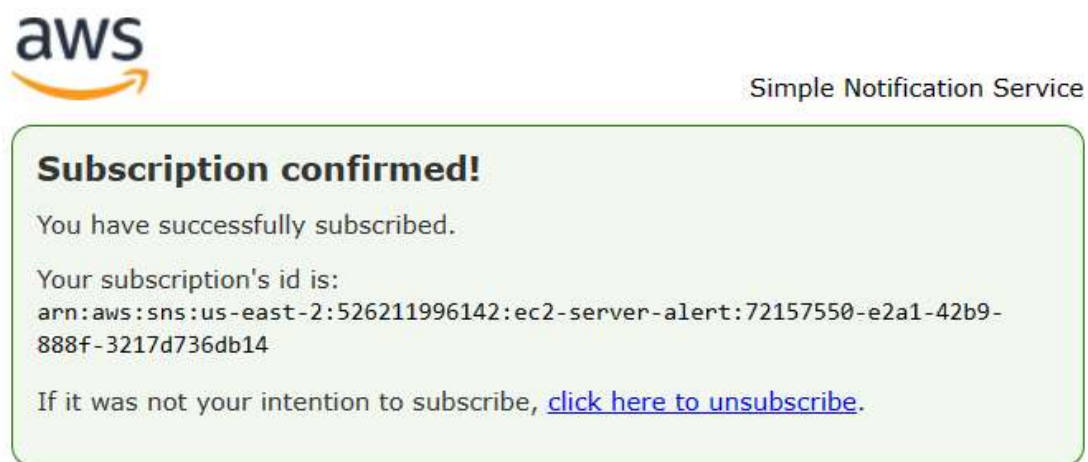
ⓘ After your subscription is created, you must confirm it. [Info](#)

✖ other settings are default and click on create subscription.

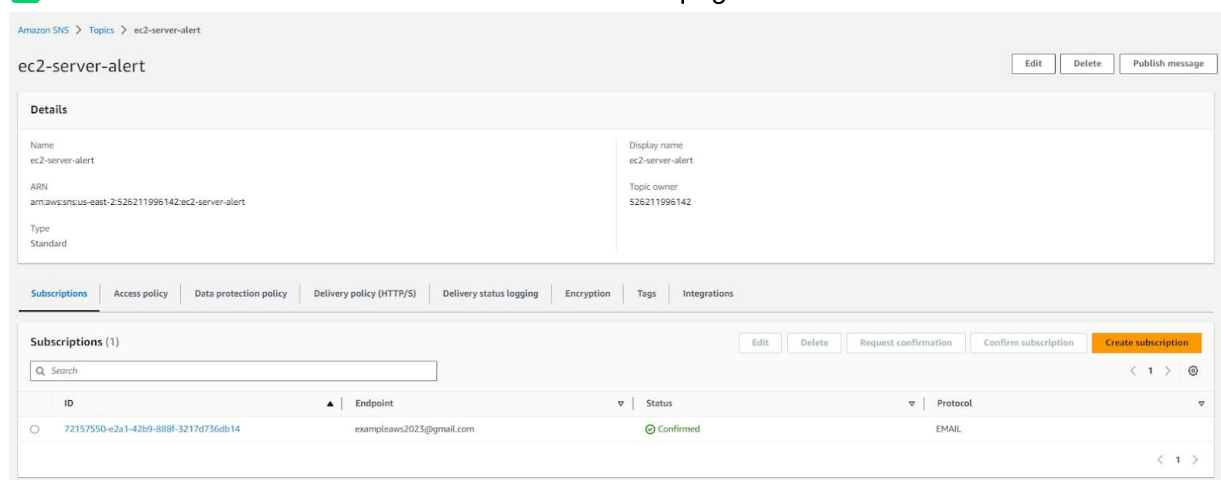
✖ We will receive an email to the above specified ID for the confirmation on subscription.  
(Please check spam folder also if did not received in Inbox)



✕ Click on confirm subscription



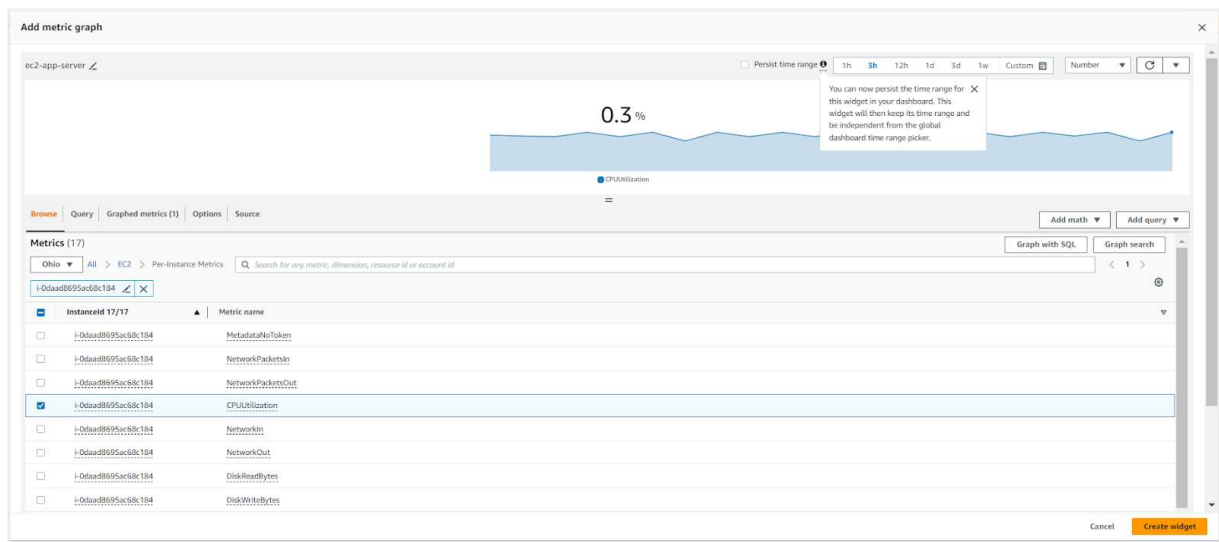
✕ We could see the email ID is confirmed from SNS page



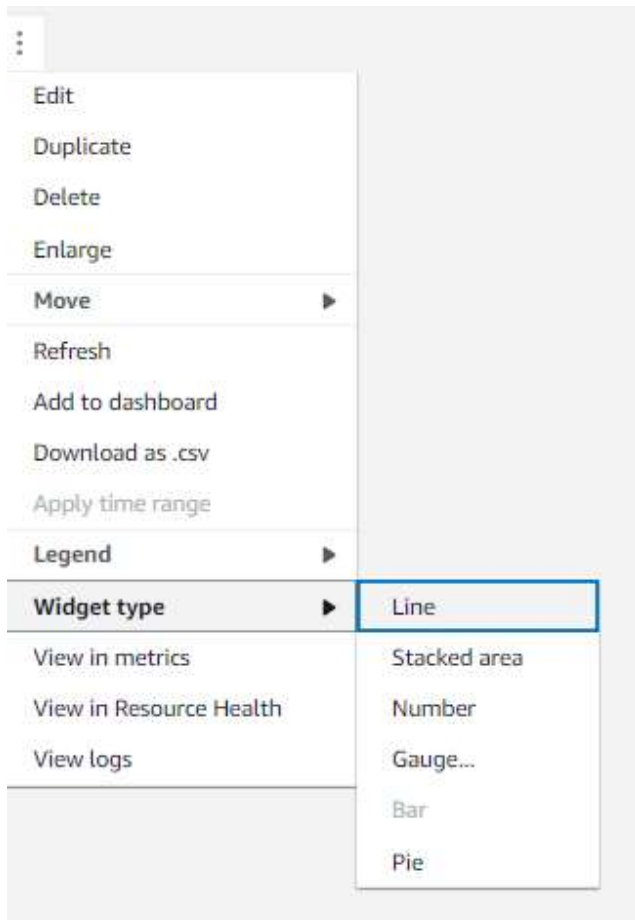
✖ we are going to create CloudWatch

✖ Goto Dashboards → Create Dashboard → give a name → Add Widget → select Number → Click on Next. Then give a graph name ( here : ec2-app-server) then select EC2 → Click on Per-Instance metrics .

✖ Now give instance ID from the ec2 instance we have created and select the metric name as per your wish. Here we're selecting CPU Utilization.



✖ Click on Create Widget. We can also create multiple widgets by clicking Duplicate and change widget type to desired type.



✔ Go to Alarms → In alarm → Create Alarm → Select Metric → click on EC2 → Per-Instance Metrics → select CPU Utilization → click on Select Metric

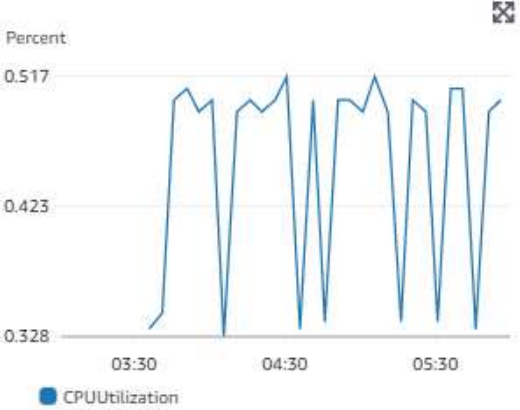


## Specify metric and conditions

**Metric**Edit

**Graph**

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



Percent

0.517

0.423

0.328

03:30 04:30 05:30

■ CPUUtilization

Namespace  
AWS/EC2

Metric name

InstanceId

Instance name  
No name specified

Statistic

Period

✕ Now we're going to give the condition that if utilization goes greater than 40% then we need to get notified.

### 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.

Must be a number

▼ Additional configuration

Datapoints to alarm  
Define the number of datapoints within the evaluation period that must be breaching to cause the alarm to go to ALARM state.

out of

Missing data treatment  
How to treat missing data when evaluating the alarm.

Treat missing data as missing

▼

Cancel

Next

✕ Then click on Next. Select the Dashboard name which we have created

## Configure actions

### Notification

#### 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.

#### 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...

ec2-server-alert



Only email lists for this account are available.

#### Email (endpoints)

exampleaws2023@gmail.com - [View in SNS Console](#)

Add notification

✖ specify what action needs to be taken if utilization goes beyond 40%. Here we're selecting to stop instances.

### 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-0daad8695ac68c184 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

✖ Give an alarm name and description and click next → Create Alarm

## Add name and description

### Name and description

Alarm name

ec2-server-monitoring

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

Edit

Preview

If CPU utilization goes greater than 40% , then the instance should be stopped.

Up to 1024 characters (80/1024)

 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

✗ to the server ssh connection (here I'm using MobaXterm) using public IP of the instance  
Give the below commands to make the cpu utilization is high.

- ☐ yum install  
[https://dl.fedoraproject.org/pub/epel/7Server/x86\\_64/Packages/e/epel-release-7-14.noarch.rpm](https://dl.fedoraproject.org/pub/epel/7Server/x86_64/Packages/e/epel-release-7-14.noarch.rpm)  
ch.rpm -y --skip-broken
- ☐ yum install stress -y
- ☐ stress --cpu 80 -----> we are giving cpu load 80

✗ Give the 'top' command in the duplicate session and see the cpu load is getting high.



✗ the instance got automatically stopped as it reach beyond 40%.

Instances (1) info										
<input type="text" value="Find instance by attribute or tag (case-sensitive)"/>										
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
<input type="checkbox"/>	-	i-0daad8695ac68c184	Stopped	t2.micro	-	1/1 has +	us-east-2b	-	-	-

✗ Also we will receive an email that cpu utilization threshold crossed.

ALARM: "ec2-server-monitoring" in US East (Ohio) [inbox x](#)

ec2-server-alert <no-reply@sns.amazonaws.com>  
to me

11:57 AM (4 minutes ago) ☆ ↶ ⋮

You are receiving this email because your Amazon CloudWatch Alarm "ec2-server-monitoring" in the US East (Ohio) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [100.0 (10/07/23 06:21:00)] was greater than the threshold (40.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Monday 10 July, 2023 06:21:11 UTC".

View this alarm in the AWS Management Console:  
<https://us-east-2.console.aws.amazon.com/cloudwatch/deeplink?region=us-east-2&alarms%2Falarm/ec2-server-monitoring>

Alarm Details:

- Name: ec2-server-monitoring
- Description: If CPU utilization goes greater than: 40%, then the instance should be stopped.
- State Change: INSUFFICIENT\_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [100.0 (10/07/23 06:21:00)] was greater than the threshold (40.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Monday 10 July, 2023 06:21:11 UTC
- AWS Account: 526211996142
- Alarm Arn: arn:aws:cloudwatch-us-east-2:526211996142:alarm:ec2-server-monitoring

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 40.0 for at least 1 of the last 1 period(s) of 60 seconds.

Monitored Metric:

- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-0daad8695ac68c184]
- Period: 60 seconds
- Statistic: Maximum
- Unit: not specified
- TreatMissingData: missing

State Change Actions:

- OK:
- ALARM: [arn:aws:sns-us-east-2:526211996142:ec2-server-alert] [arn:aws:swf-us-east-2:526211996142:action/actions/AWS\_EC2\_InstanceStop/1.0]
- INSUFFICIENT\_DATA:

-

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:  
<https://sns.us-east-2.amazonaws.com/unsubscribe?SubscriptionArn=arn:aws:sns-us-east-2:526211996142:ec2-server-alert:72157550-e2d1-4269-888f-3217d726db14&Endpoint=exampleaws2023@gmail.com>

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at <https://aws.amazon.com/support>

