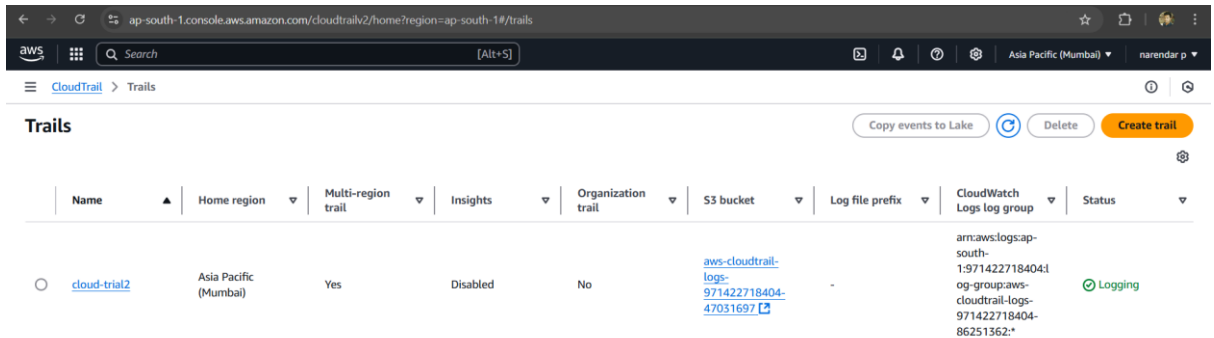


TASK ON MONITORING

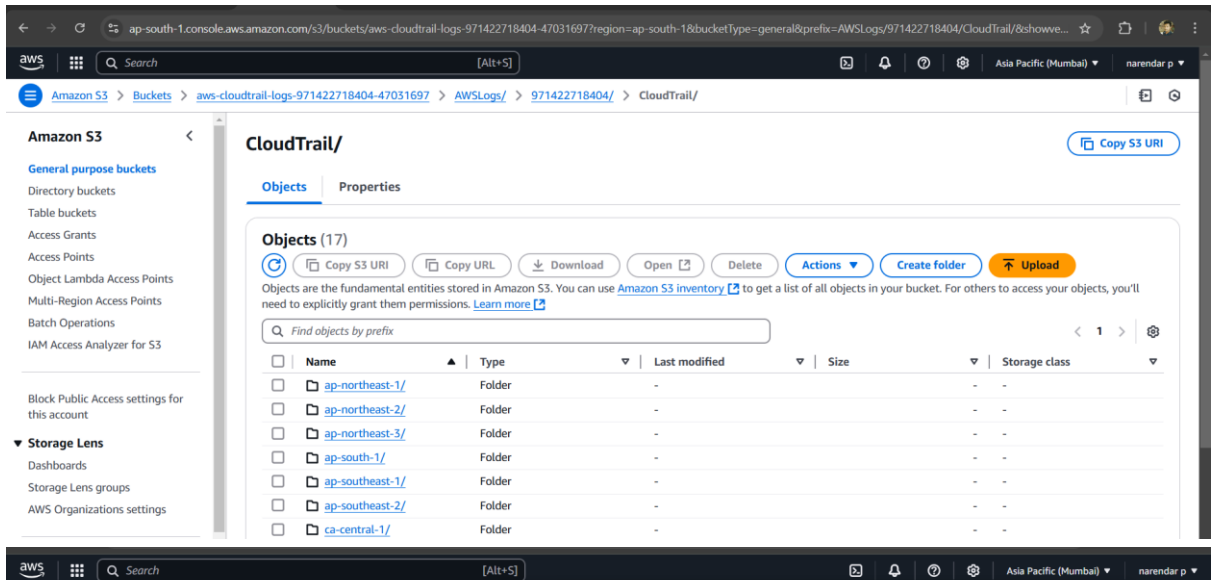
1) Enable cloud trail monitoring and store the events in s3 and cloud watch log events:

Stored cloud trail logs in s3 bucket:



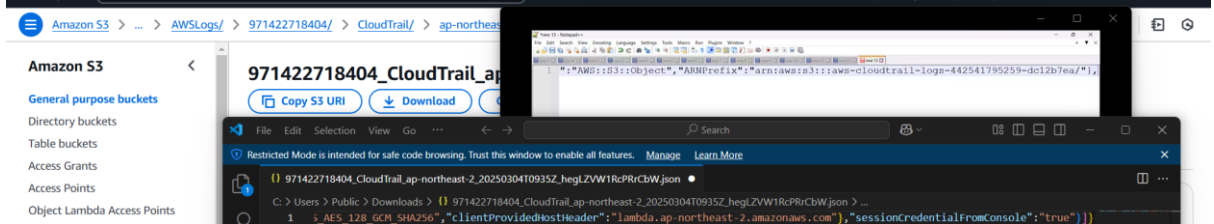
The screenshot shows the AWS CloudTrail console. At the top, there are buttons for 'Copy events to Lake', 'Delete', and 'Create trail'. Below this is a table of trails. The table has columns for Name, Home region, Multi-region trail, Insights, Organization trail, S3 bucket, Log file prefix, CloudWatch Logs log group, and Status. One trail is listed: 'cloud-trial2' in the Asia Pacific (Mumbai) region, which is a multi-region trail with insights disabled, no organization trail, and logs stored in the S3 bucket 'aws-cloudtrail-logs-971422718404-47031697'. The log file prefix is '-', and the CloudWatch Logs log group is 'arn:aws:logs:ap-south-1:971422718404:og-group:aws-cloudtrail-logs-971422718404-86251362:'. The status is 'Logging'.

| Name | Home region | Multi-region trail | Insights | Organization trail | S3 bucket | Log file prefix | CloudWatch Logs log group | Status |
|--------------|-----------------------|--------------------|----------|--------------------|---|-----------------|--|---------|
| cloud-trial2 | Asia Pacific (Mumbai) | Yes | Disabled | No | aws-cloudtrail-logs-971422718404-47031697 | - | arn:aws:logs:ap-south-1:971422718404:og-group:aws-cloudtrail-logs-971422718404-86251362: | Logging |



The screenshot shows the Amazon S3 console for the bucket 'aws-cloudtrail-logs-971422718404-47031697'. The 'Objects' tab is selected, showing a list of 17 objects. The objects are folders representing different AWS regions: 'ap-northeast-1/', 'ap-northeast-2/', 'ap-northeast-3/', 'ap-south-1/', 'ap-southeast-1/', 'ap-southeast-2/', and 'ca-central-1/'. Each folder has a size of 0 bytes and is stored in the 'Standard' storage class. The console also shows a search bar and various action buttons like 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'.

| Name | Type | Last modified | Size | Storage class |
|-----------------|--------|---------------|------|---------------|
| ap-northeast-1/ | Folder | - | - | - |
| ap-northeast-2/ | Folder | - | - | - |
| ap-northeast-3/ | Folder | - | - | - |
| ap-south-1/ | Folder | - | - | - |
| ap-southeast-1/ | Folder | - | - | - |
| ap-southeast-2/ | Folder | - | - | - |
| ca-central-1/ | Folder | - | - | - |



Enable cloud watch to store cloud trial logs:

cloud-trial2

General details

Trail logging
Logging

Trail name
cloud-trial2

Multi-region trail
Yes

Apply trail to my organization
Not enabled

Trail log location
aws-cloudtrail-logs-971422718404-47031697/AWSLogs/971422718404

Last log file delivered
March 04, 2025, 15:17:57 (UTC+05:30)

Log file SSE-KMS encryption
Not enabled

Log file validation
Enabled

Last file validation delivered
-

SNS notification delivery
arn:aws:sns:ap-south-1:971422718404:alert

Last SNS notification
March 04, 2025, 15:17:57 (UTC+05:30)

CloudWatch Logs

Log group
aws-cloudtrail-logs-971422718404-86251362

IAM Role
arn:aws:iam::971422718404:role/service-role/cloudtrial-event-role

Cloud trail logs stored in cloud watch:

CloudWatch

Log groups (2)

By default, we only load up to 10000 log groups.

Filter log groups or try prefix search

Exact match

| Log group | Log class | Anomaly d... | Data pr... | Sensitiv... | Retention | Metric ... |
|---|-----------|--------------|------------|-------------|--------------|------------|
| aws-cloudtrail-logs-971422718404-86251362 | Standard | Configure | - | - | Never expire | - |
| vpc-flow-log-cloudwatch | Standard | Configure | - | - | Never expire | - |

CloudWatch

Log groups

aws-cloudtrail-logs-971422718404-86251362

Creation time
14 minutes ago

Retention
Never expire

Stored bytes
-

KMS key ID
-

Anomaly detection
Configure

Transformer
Configure

Log streams (4)

Filter log streams or try prefix search

Exact match

Show expired

Info

| Log stream | Last event time |
|--------------------------------------|---------------------------|
| 971422718404_CloudTrail_ap-south-1 | 2025-03-04 10:02:29 (UTC) |
| 971422718404_CloudTrail_ap-south-1_2 | 2025-03-04 10:02:14 (UTC) |
| 971422718404_CloudTrail_ap-south-1_4 | 2025-03-04 10:00:07 (UTC) |
| 971422718404_CloudTrail_ap-south-1_3 | 2025-03-04 09:58:15 (UTC) |

aws CloudWatch > Log groups > aws-cloudtrail-logs-971422718404-86251362 > 971422718404_CloudTrail_ap-south-1

CloudWatch

- Favorites and recents
- Dashboards
- Alarms
- Logs
 - Log groups
 - Log Anomalies
 - Live Tail
 - Logs Insights
 - Contributor Insights
- Metrics
- X-Ray traces

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search Clear 1m 30m 1h 12h Custom UTC timezone

| Timestamp | Message |
|---|---|
| No older events at this moment. Retry | |
| 2025-03-04T10:02:29.115Z | {"eventVersion":"1.10","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.115Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.11","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.11","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.10","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.10","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.11","userIdentity":{"type":"AssumedRole","principalId":"AROAGELKOLHCHUEZH2F55:CLOU","arn":"arn:aws:iam::971422718404:role/CloudTrail_ap-south-1"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |
| 2025-03-04T10:02:29.116Z | {"eventVersion":"1.10","userIdentity":{"type":"Root","principalId":"971422718404","arn":"arn:aws:iam::971422718404:root"},"eventSource":"aws:cloudtrail","eventName":"CreateTrail","requestParameters":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"responseElements":{"trailArn":"arn:aws:cloudtrail:ap-south-1:971422718404:trail/CloudTrail_ap-south-1"},"eventType":"AwsApiCall","readOnlyFields":["aws:sourceip","aws:userid","aws:username"],"details":{"trailName":"CloudTrail_ap-south-1","cloudWatchLogsGroupArn":"arn:aws:logs:ap-south-1:971422718404:log-group:/aws-logs-971422718404-86251362-1:aws-logs-971422718404-86251362-1-1"},"eventTime":"2025-03-04T10:02:29.116Z"} |

2) Enable SNS for cloud trial to send alert on email:

→Enable sns:

aws CloudTrail > Trails > arn:aws:cloudtrail:ap-south-1:971422718404:trail/cloud-trial2 > Edit

Trail log bucket name

Enter a new S3 bucket name and folder (prefix) to store your logs. Bucket names must be globally unique.

aws-cloudtrail-logs-971422718404-47031697 Browse

Prefix - optional
prefix
Logs will be stored in aws-cloudtrail-logs-971422718404-47031697/AWSLogs/971422718404

Log file SSE-KMS encryption [Info](#)
☐ Enabled

Additional settings

Log file validation [Info](#)
☒ Enabled

SNS notification delivery [Info](#)
☒ Enabled

Create a new SNS topic
☒ New
☐ Existing

SNS topic
aws-cloudtrail-logs-971422718404-822145fb

Cancel Save changes

→ create one alert:

The screenshot shows the Amazon SNS console interface. On the left, the navigation menu includes 'Amazon SNS', 'Dashboard', 'Topics', 'Subscriptions', and 'Mobile'. The main content area displays the 'alert' topic details. A green banner at the top indicates 'Topic alert created successfully. You can create subscriptions and send messages to them from this topic.' Below this, the 'Details' section shows the Name (alert), Display name (cloudtrial-alert), ARN (arn:aws:sns:ap-south-1:971422718404:alert), and Type (Standard). The 'Subscriptions' section shows 0 subscriptions with buttons for 'Edit', 'Delete', 'Request confirmation', 'Confirm subscription', and 'Create subscription'. The 'Access policy', 'Data protection policy', 'Delivery policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags' tabs are also visible.

→ in alert create subscription with email:

The screenshot shows the Amazon SNS console interface for a specific subscription. The breadcrumb trail is 'Amazon SNS > Topics > alert > Subscription: 041ace16-6c5f-4a5d-a48a-58cc97038330'. The 'Details' section shows the ARN (arn:aws:sns:ap-south-1:971422718404:alert:041ace16-6c5f-4a5d-a48a-58cc97038330), Endpoint (narenderpashamolla@gmail.com), Topic (alert), and Subscription Principal (arn:aws:iam:971422718404:root). The 'Status' is 'Pending confirmation' and the 'Protocol' is 'EMAIL'. The 'Subscription filter policy' section shows 'Redrive policy (dead-letter queue)' and a message stating 'No filter policy configured for this subscription. To apply a filter policy, edit this subscription.' with an 'Edit' button.

→ Confirmed subscription in gmail:

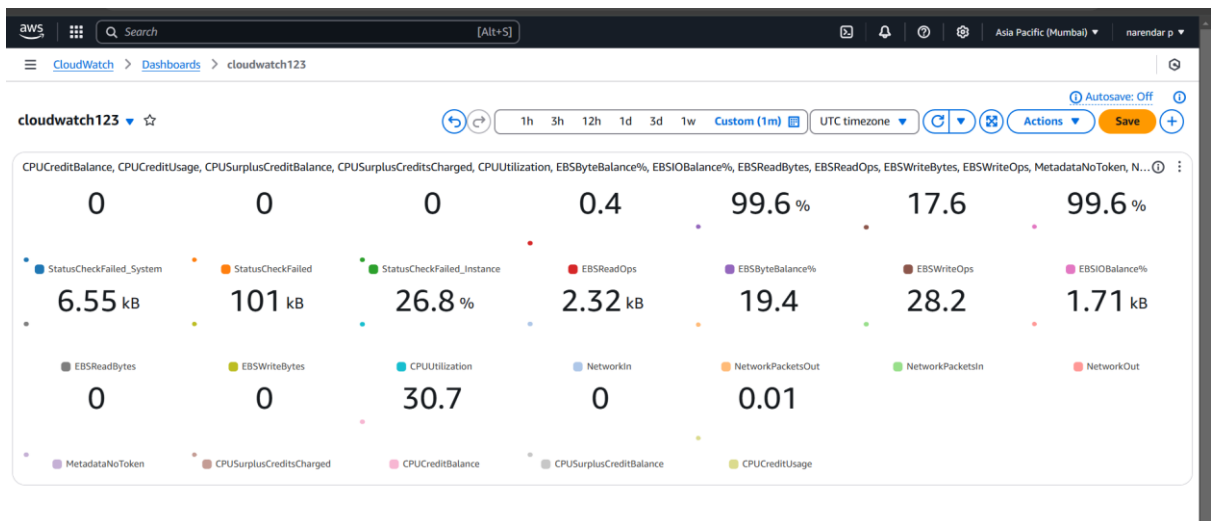
The screenshot shows a Gmail confirmation email from AWS SNS. The email header includes the AWS logo and 'Simple Notification Service'. The main body of the email states 'Subscription confirmed! You have successfully subscribed.' and provides the subscription ID: 'arn:aws:sns:ap-south-1:971422718404:alert:041ace16-6c5f-4a5d-a48a-58cc97038330'. It also includes a link to 'click here to unsubscribe'.

→ alert status -confirmed:

The screenshot shows the Amazon SNS console interface. On the left, there's a navigation menu with 'Amazon SNS', 'Dashboard', 'Topics', 'Subscriptions', and 'Mobile'. The main content area displays the details for a subscription with ID '041ace16-6c5f-4a5d-a48a-58cc97038330'. A blue banner at the top indicates a 'New Feature' about High Throughput FIFO topics. The subscription details include: ARN, Endpoint (narenderpashamolla@gmail.com), Topic (alert), and Subscription Principal. The status is 'Confirmed' with a green checkmark icon. There are 'Edit' and 'Delete' buttons at the top right of the details section.

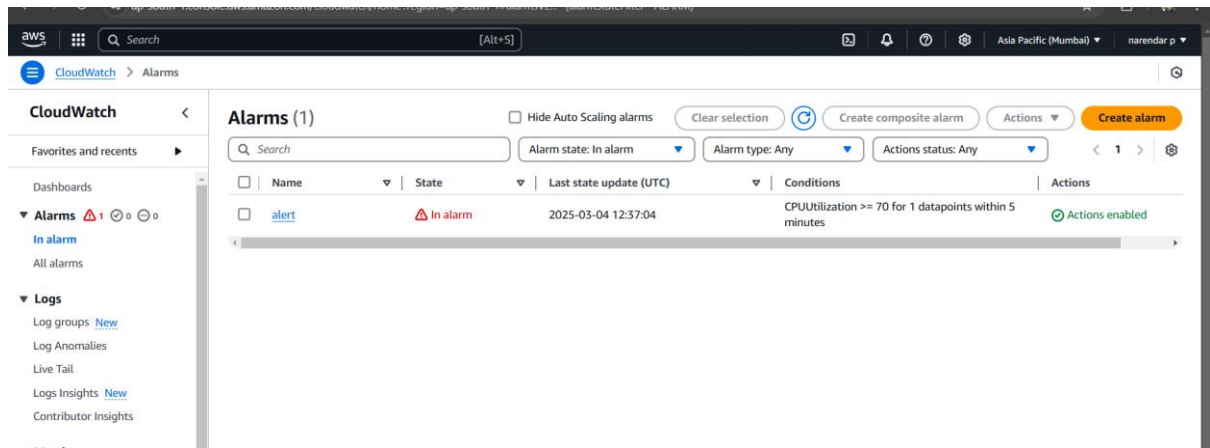
The screenshot shows a Gmail inbox with several 'AWS Notification Message' emails from 'cloudtrial-alert'. The messages are about CloudTrail validation and include unsubscribe links. The most recent message is dated 15:14 (21 minutes ago). The email content includes a JSON payload with AWSLogs information.

3)Configure cloud watch monitoring and record the cpu utilization and other metrics of ec2:

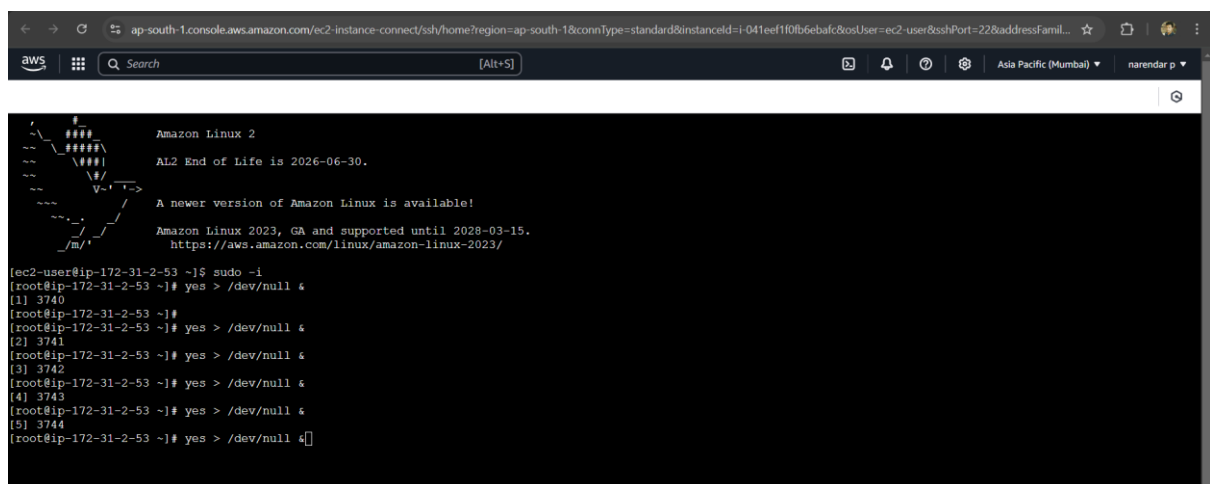


4) Create one alarm to send alert to email if the cpu utilization is more than 70 percent:

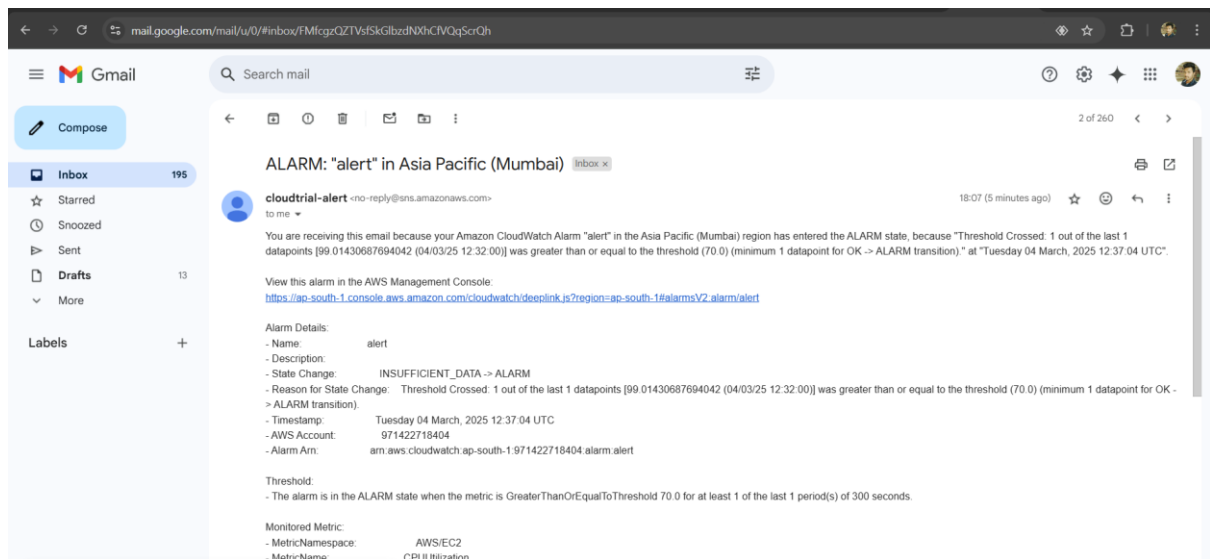
→ Created one alarm:



→ applied stress on instance:

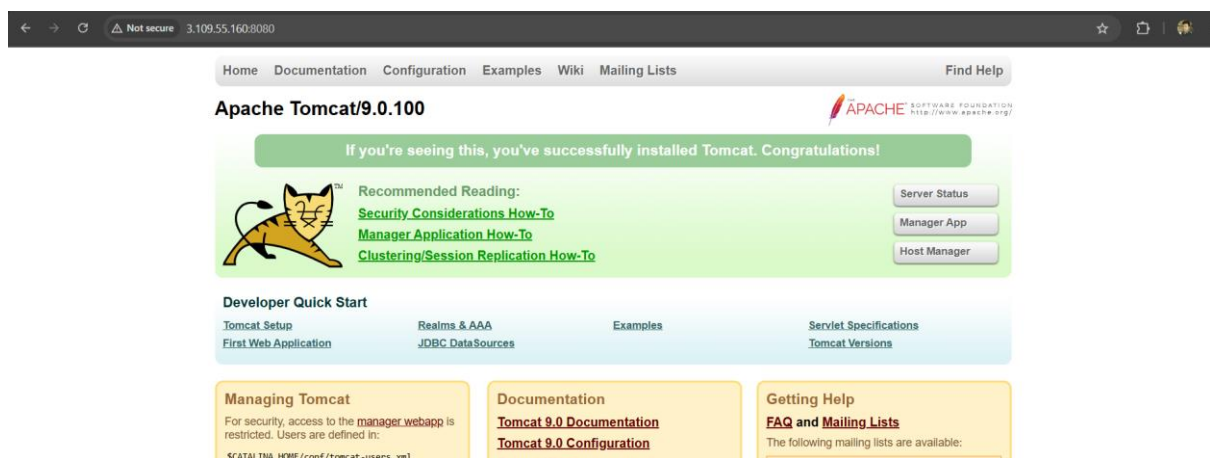


→Got email:



5) Create Dashboard and monitor tomcat service whether it is running or not and send the alert:

→Started apache tomcat:



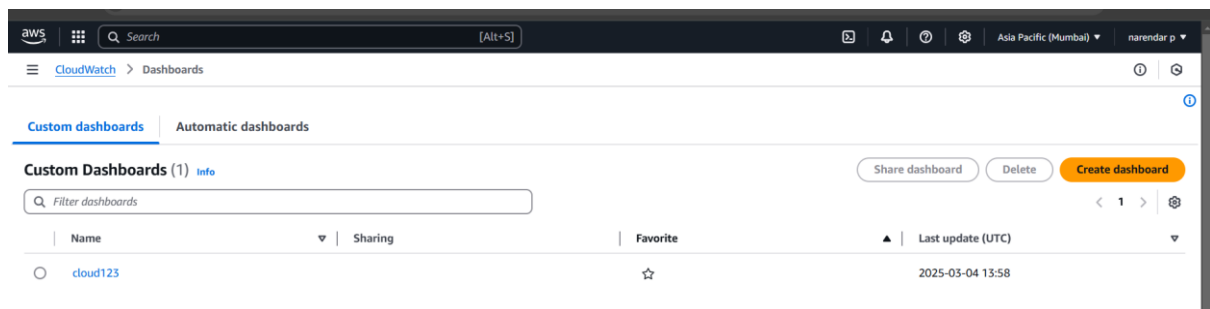
→ Bash Script to check tomcat's running status and send the status value to cloudwatch metric

```
#!/bin/bash

# Get the EC2 instance ID
INSTANCE_ID=$(/opt/aws/bin/ec2-metadata -i | cut -d " " -f2)

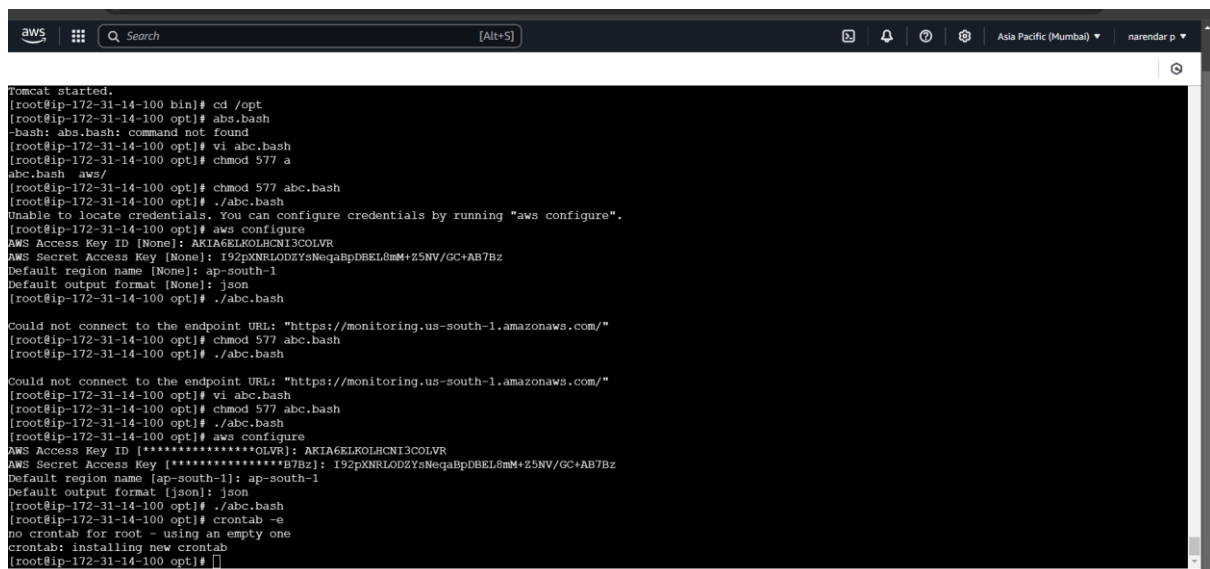
#checking tomcat is running or not
count=$(ps -ef | grep "/opt/apache-tomcat-9.0.100/" | grep -v grep | wc -l)
if [ $count -eq 0 ]
then
    echo "not running - sending 0 to cloudwatch"
    aws --region us-east-1 cloudwatch put-metric-data --metric-name tomcat --value 0 --namespace tomcat --dimensions InstanceId=$INSTANCE_ID
else
    echo "tomcat is running sending 1 to cloudwatch"
    aws --region us-east-1 cloudwatch put-metric-data --metric-name tomcat --value 1 --namespace tomcat --dimensions InstanceId=$INSTANCE_ID
fi
```

→ created dashboard:



The screenshot shows the AWS CloudWatch Dashboards interface. The top navigation bar includes the AWS logo, a search bar, and the user's name 'narendar p'. The main content area is titled 'Dashboards' and has tabs for 'Custom dashboards' and 'Automatic dashboards'. Under 'Custom dashboards', there is a list of one dashboard named 'cloud123'. The dashboard is marked as a favorite and was last updated on 2025-03-04 13:58. Buttons for 'Share dashboard', 'Delete', and 'Create dashboard' are visible.

→ Scheduled cron job tab to run script every minute:



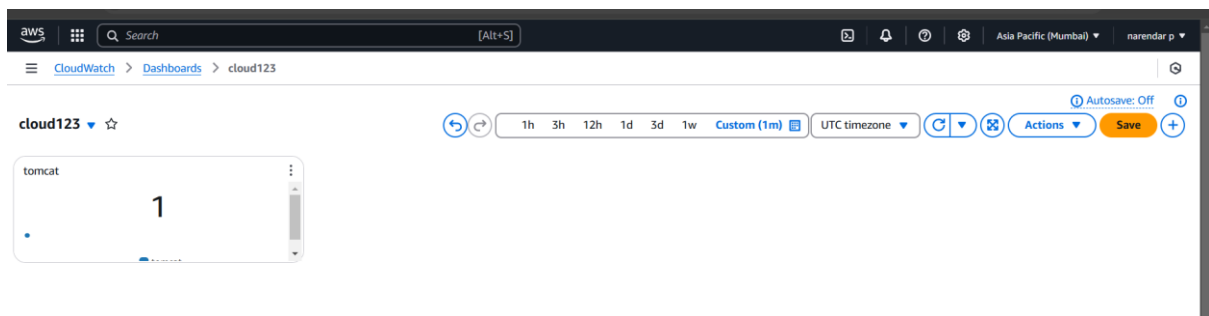
The screenshot shows a terminal window with the following commands and output:

```
Tomcat started.
[root@ip-172-31-14-100 bin]# cd /opt
[root@ip-172-31-14-100 opt]# abc.bash
-bash: abc.bash: command not found
[root@ip-172-31-14-100 opt]# vi abc.bash
[root@ip-172-31-14-100 opt]# chmod 577 a
abc.bash  aws/
[root@ip-172-31-14-100 opt]# chmod 577 abc.bash
[root@ip-172-31-14-100 opt]# ./abc.bash
Unable to locate credentials. You can configure credentials by running "aws configure".
[root@ip-172-31-14-100 opt]# aws configure
AWS Access Key ID [None]: AKIA6ELKOLHCN13COLVR
AWS Secret Access Key [None]: I92pXNRL0DZYsNegaBpDBEL8mM+Z5NV/GC+AB7Bz
Default region name [None]: ap-south-1
Default output format [None]: json
[root@ip-172-31-14-100 opt]# ./abc.bash

Could not connect to the endpoint URL: "https://monitoring.us-south-1.amazonaws.com/"
[root@ip-172-31-14-100 opt]# chmod 577 abc.bash
[root@ip-172-31-14-100 opt]# ./abc.bash

Could not connect to the endpoint URL: "https://monitoring.us-south-1.amazonaws.com/"
[root@ip-172-31-14-100 opt]# vi abc.bash
[root@ip-172-31-14-100 opt]# chmod 577 abc.bash
[root@ip-172-31-14-100 opt]# ./abc.bash
[root@ip-172-31-14-100 opt]# aws configure
AWS Access Key ID [*****OLVR]: AKIA6ELKOLHCN13COLVR
AWS Secret Access Key [*****B7Bz]: I92pXNRL0DZYsNegaBpDBEL8mM+Z5NV/GC+AB7Bz
Default region name [ap-south-1]: ap-south-1
Default output format [json]: json
[root@ip-172-31-14-100 opt]# ./abc.bash
[root@ip-172-31-14-100 opt]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@ip-172-31-14-100 opt]#
```


→ Monitoring tomcat running status through Cloudwatch dashboard



6) Create Dashboard and monitor nginx service to send the alert if nginx is not running:

→ Installed nginx and started it:

```
[root@ip-172-31-0-21 ~]# yum install nginx
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package 1:nginx-1.22.1-1.amzn2.0.4.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-0-21 ~]# systemctl start nginx
[root@ip-172-31-0-21 ~]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; vendor prese
  t: disabled)
   Active: active (running) since Wed 2025-03-05 09:54:36 UTC; 29s ago
     Process: 3447 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
     Process: 3443 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
     Process: 3442 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=
0/SUCCESS)
    Main PID: 3449 (nginx)
      CGroup: /system.slice/nginx.service
              └─3449 nginx: master process /usr/sbin/nginx
                  └─3450 nginx: worker process
```

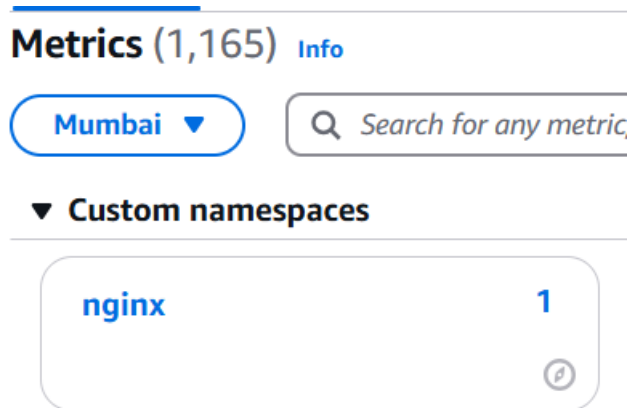
→ Script to check Nginx running status and send the status value to cloud watch metrics:

```
root@ip-172-31-0-21:~#
#!/bin/bash
# Get the EC2 instance ID
INSTANCE_ID=$(/opt/aws/bin/ec2-metadata -i | cut -d " " -f2)
# Checking if Nginx is running
count=$(ps -ef | grep "nginx" | grep -v grep | wc -l)
count=$((ps -C nginx --no-headers | wc -l))
if [ $count -eq 0 ]
then
    echo "Nginx not running - sending 0 to Cloudwatch"
    aws --region ap-south-1 cloudwatch put-metric-data --metric-name nginx --value 0 --namespace nginx --dimensions InstanceId=$INSTANCE_ID
else
    echo "Nginx is running - sending 1 to Cloudwatch"
    aws --region ap-south-1 cloudwatch put-metric-data --metric-name nginx --value 1 --namespace nginx --dimensions InstanceId=$INSTANCE_ID
fi
~
~
~
```

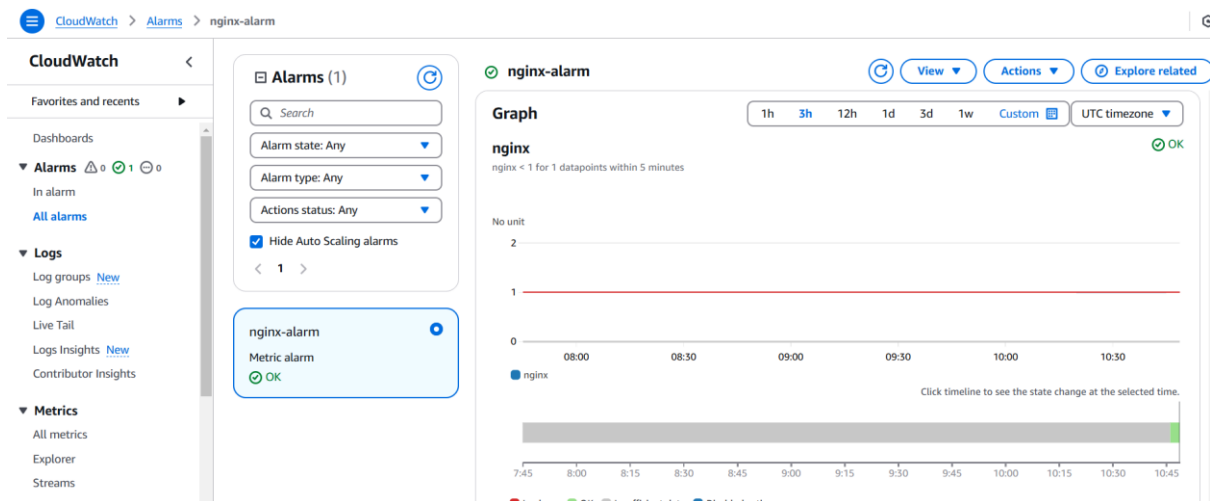
→ Scheduled a cron job to run above script every minute:

```
[root@ip-172-31-0-21 ~]# ./cloud.bash
Nginx not running - sending 0 to CloudWatch
[root@ip-172-31-0-21 ~]# pwd
/root
[root@ip-172-31-0-21 ~]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@ip-172-31-0-21 ~]# crontab -e
crontab: installing new crontab
[root@ip-172-31-0-21 ~]# crontab -e
```

→ Monitoring Nginx running status through Cloudwatch dashboard:



→ Alarm showing ok status as Nginx running smoothly:

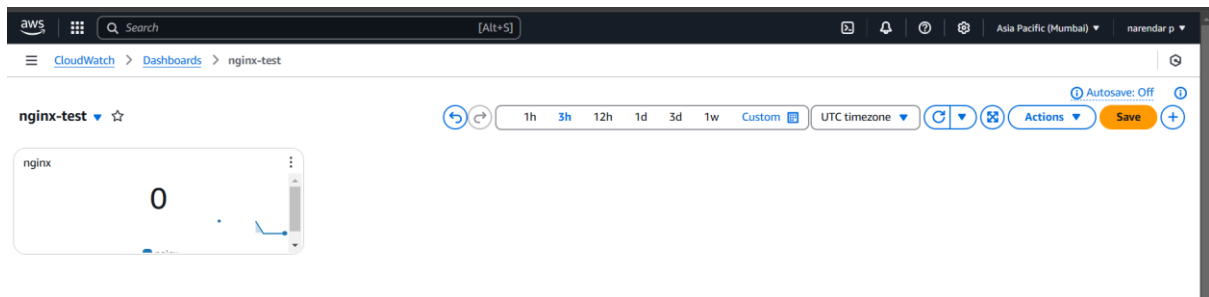


→ Stop Nginx:

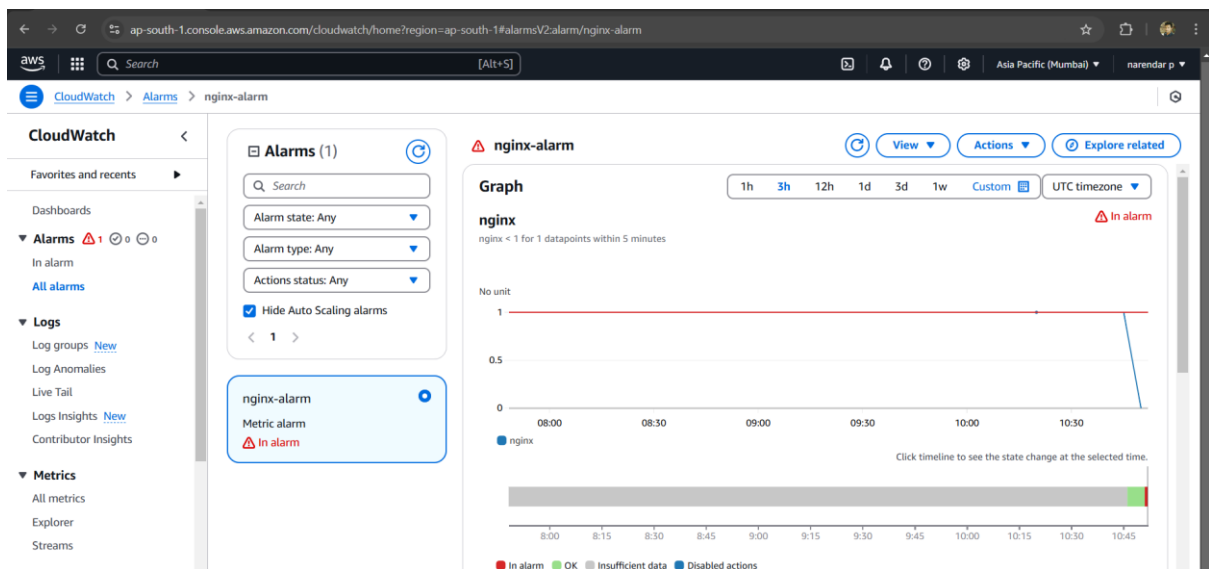
```
[root@ip-172-31-0-21 ~]# systemctl stop nginx
[root@ip-172-31-0-21 ~]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; vendor preset: disabled)
   Active: inactive (dead)

Mar 05 09:54:35 ip-172-31-0-21.ap-south-1.compute.internal systemd[1]: Starting The nginx HTTP and reverse proxy server...
Mar 05 09:54:35 ip-172-31-0-21.ap-south-1.compute.internal nginx[3443]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Mar 05 09:54:35 ip-172-31-0-21.ap-south-1.compute.internal nginx[3443]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Mar 05 09:54:36 ip-172-31-0-21.ap-south-1.compute.internal systemd[1]: Started The nginx HTTP and reverse proxy server.
Mar 05 10:49:53 ip-172-31-0-21.ap-south-1.compute.internal systemd[1]: Stopping The nginx HTTP and reverse proxy server...
Mar 05 10:49:53 ip-172-31-0-21.ap-south-1.compute.internal systemd[1]: Stopped The nginx HTTP and reverse proxy server.
[root@ip-172-31-0-21 ~]# ./cloud.bash
Nginx not running - sending 0 to CloudWatch
```

→ Cloudwatch dashboard reflecting 0 as Nginx is not running on EC2 server:



→Metric Alarm state changed from ok to Alarm:



→ Got an Email alert with the alarm status:

16:21 (11 minutes ago) ☆ 😊 ↩ ⋮

<https://ap-south-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-south-1#alarmsV2:alarm/nginx-alarm>

```
- Name: nginx-alarm
- Description:
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [0.0 (05/03/2025 05:11:10)] crossed the threshold 0.0
- Timestamp: Wednesday 05 March, 2025 10:51:10 UTC
- AWS Account: 971422718404
- Alarm Arn: arn:aws:cloudwatch:ap-south-1:971422718404:alarm:nginx-alarm
```

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