

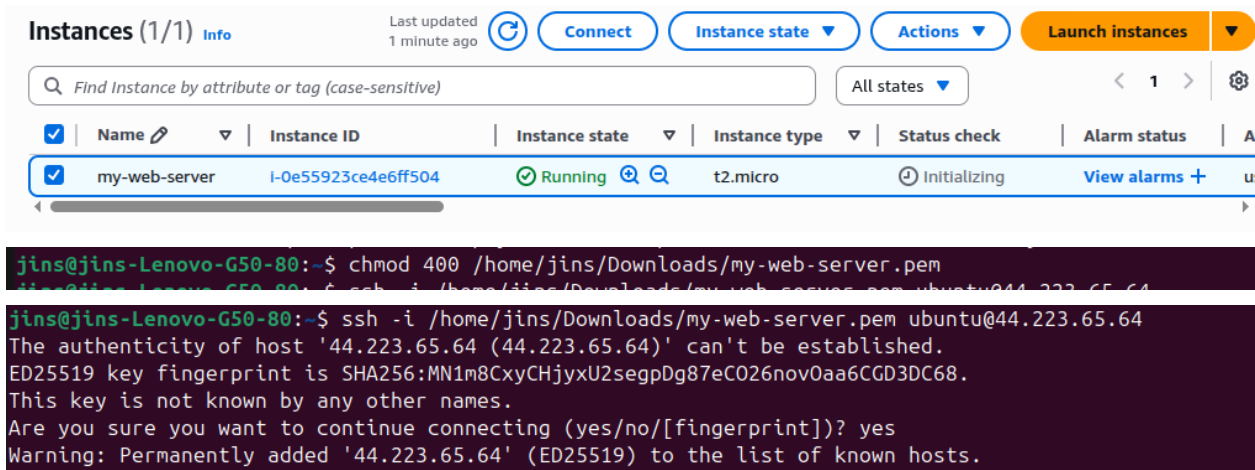
# Implementation of CloudWatch CPU Alarm and SNS Email Notification

## Overview

Amazon CloudWatch alarm that monitors the CPU usage of an EC2 instance. When the CPU exceeds a threshold (80%), the alarm triggers an SNS email notification so the support team can take action.

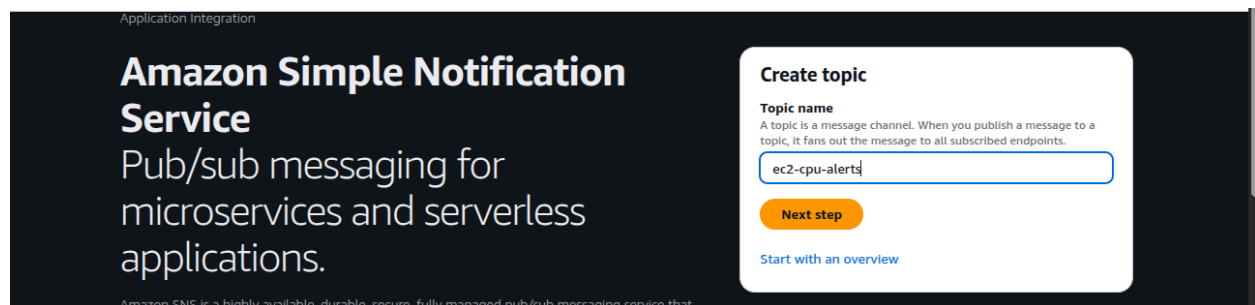
## Prerequisites

- 1.AWS EC2 instance (t2.micro) with port 22 allowed (allow only your IP) for ssh access and port 80 allowed (for http traffic from the internet).
2. Select OS ubuntu 24.04
- 3.Create key value pair for ssh access change permission to 400 for maintaining best security practice.



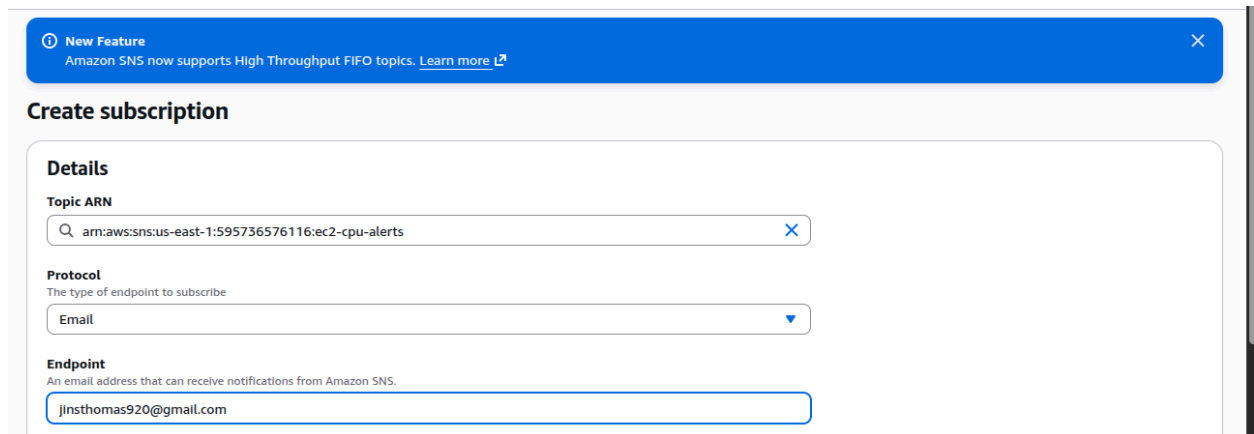
The screenshot displays the AWS Management Console for EC2 instances. The instance 'my-web-server' (ID: i-0e55923ce4e6ff504) is shown as 'Running'. Below the console, a terminal window shows the command 'chmod 400 /home/jins/Downloads/my-web-server.pem' and the execution of 'ssh -i /home/jins/Downloads/my-web-server.pem ubuntu@44.223.65.64'. The terminal output shows a warning about the host's authenticity and a confirmation to add it to the list of known hosts.

## 1.Create SNS topic using console and create subscription



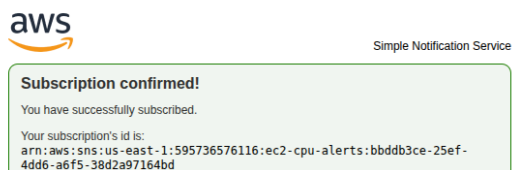
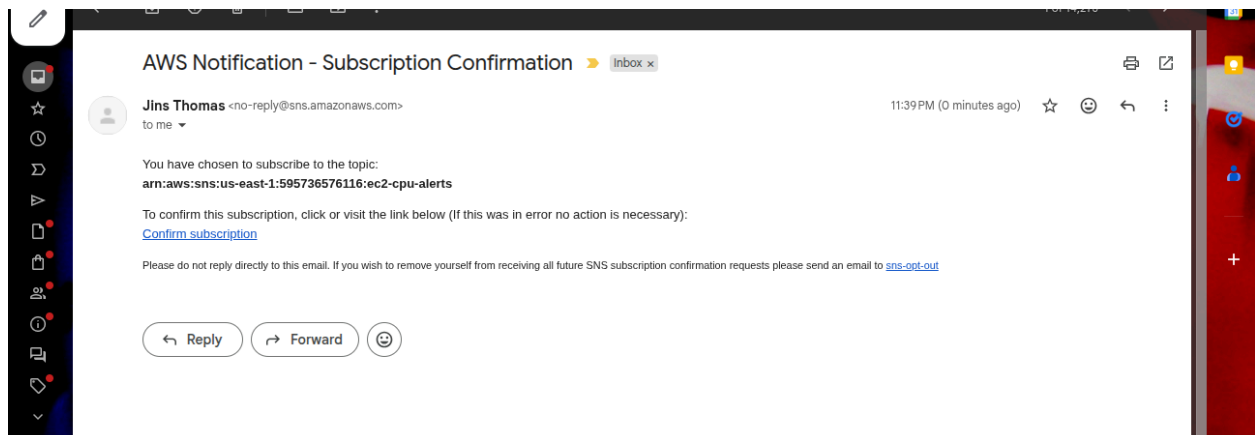
The screenshot shows the Amazon Simple Notification Service (SNS) console. The 'Create topic' form is displayed, with the topic name 'ec2-cpu-alerts' entered. The 'Next step' button is highlighted.

While creating a subscription select protocol i.e the communication method. I have selected email and provided my email as the endpoint.



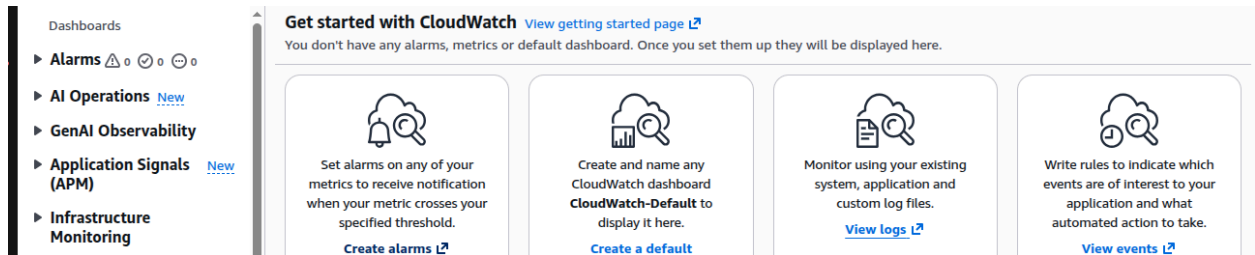
The screenshot shows the 'Create subscription' page in the AWS Management Console. At the top, there is a blue banner with a 'New Feature' icon and text: 'Amazon SNS now supports High Throughput FIFO topics. [Learn more](#)'. Below this, the page title is 'Create subscription'. Under the 'Details' section, there are three fields: 'Topic ARN' with the value 'arn:aws:sns:us-east-1:595736576116:ec2-cpu-alerts', 'Protocol' set to 'Email' (with a dropdown arrow), and 'Endpoint' with the value 'jinsthomas920@gmail.com'.

Once the subscription is created make sure you confirm the subscription received in your mail. Else AWS will not send the alert notification.

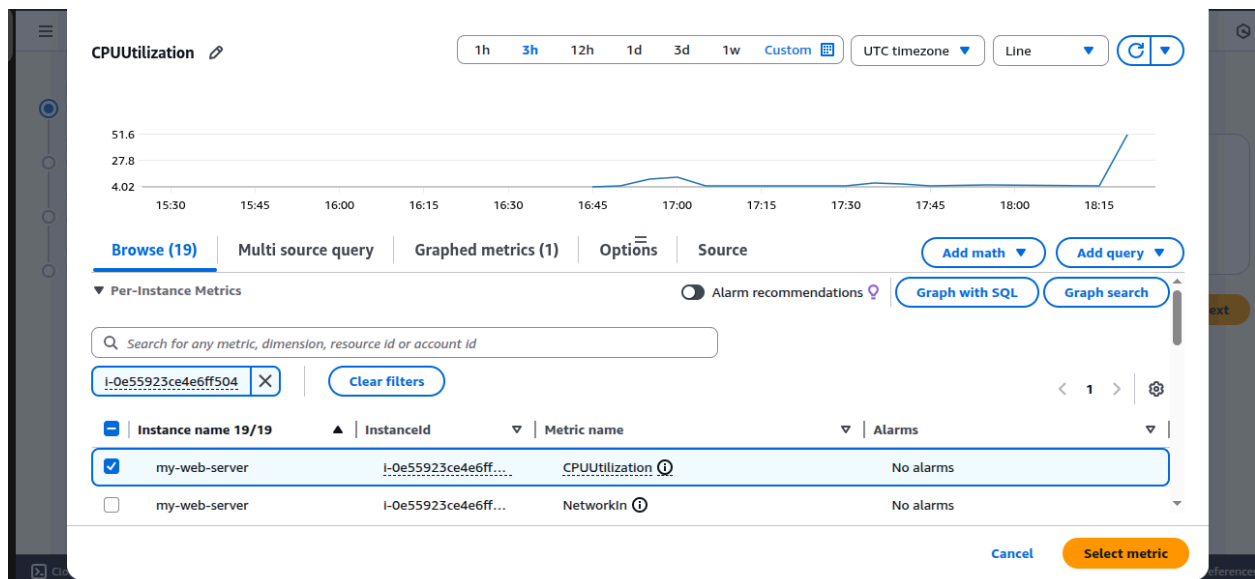


## 2. Creating the cloudwatch alarm

Select create alarm



Select the EC2 instance by identifying the instance id and select the desired metric. In this case i have selected cpu utilization of my EC2 instance(my-web-server).



Next we will specify the metric condition. The metric name in our case will be CPU utilization select statistics as average.

The screenshot shows the 'Specify metric and conditions' step of the AWS CloudWatch 'Create alarm' wizard. The left sidebar shows a progress bar with four steps: 'Specify metric and conditions' (selected), 'Configure actions', 'Add alarm details', and 'Preview and create'. The main content area is titled 'Specify metric and conditions' and contains a 'Metric' section. The 'Metric' section includes a 'Graph' showing CPU utilization over time, with a blue line representing the metric and a red line indicating the threshold. The graph shows a peak in CPU utilization around 16:30. The 'Metric' section also includes a 'Namespace' dropdown set to 'AWS/EC2', a 'Metric name' input field containing 'CPUUtilization', an 'Instanceid' input field containing 'i-0e55923ce4e6ff504', an 'Instance name' input field containing 'my-web-server', and a 'Statistic' dropdown set to 'Average'. An 'Edit' button is located in the top right corner of the 'Metric' section.

Now we will define the conditions for firing our alarm to receive an email notification on our endpoint which is my email w.r.t high CPU utilization for our configured instance. so in our case it will be fired when the cpu utilization is  $\geq 50$  (this condition is just for hands-on practice and exploring the service).

The screenshot shows the 'Conditions' step of the AWS CloudWatch 'Create alarm' wizard. The 'Threshold type' section has two options: 'Static' (selected) and 'Anomaly detection'. The 'Whenever CPUUtilization is...' section has four options: 'Greater', 'Greater/Equal' (selected), 'Lower/Equal', and 'Lower'. The 'than...' section has a text input field containing '50'. The 'Additional configuration' section is currently collapsed. 'Cancel' and 'Next' buttons are located at the bottom right of the form.

Once we are done with alarm creation we will now configure the actions for our alarm. So in the notification part in alarm we select an SNS topic that we have already created by the name ec2-cpu-alerts.

The screenshot shows the 'Configure actions' step in the AWS CloudWatch console. On the left, a sidebar lists four steps: 'Specify metric and conditions', 'Configure actions' (selected), 'Add alarm details', and 'Preview and create'. The main content area is titled 'Configure actions' and contains a 'Notification' section. Under 'Alarm state trigger', three radio buttons are visible: 'In alarm' (selected), 'OK', and 'Insufficient data'. Below this, the 'Send a notification to the following SNS topic' section has three radio buttons: 'Select an existing SNS topic' (selected), 'Create new topic', and 'Use topic ARN to notify other accounts'. A search bar shows 'ec2-cpu-alerts' as the selected topic. At the bottom, it lists an email endpoint 'jinsthomas920@gmail.com' with a link to 'View in SNS Console'.

Next we will provide the name and description of the alarm. I have provided High CPU Utilization and a short description that we can see in the below screenshot. After this we have successfully created the alarm.

The screenshot shows the 'Add alarm details' step in the AWS CloudWatch console. A blue banner at the top states 'Alarm recommendations available'. The sidebar on the left shows four steps: 'Specify metric and conditions', 'Configure actions', 'Add alarm details' (selected), and 'Preview and create'. The main content area is titled 'Add alarm details' and contains a 'Name and description' section. The 'Alarm name' field contains 'High CPU Utilization'. The 'Alarm description - optional' field is active, showing a preview of the text: 'you insatce has reached the threshold value of maximum CPU usage that is 50'. The description field has a character count of 'Up to 1024 characters (74/1024)'.

The screenshot shows the AWS CloudWatch Alarms console. A green banner at the top states "Successfully created alarm High CPU Utilization." with a "View alarm" button. Below this, the "Alarms (1)" section is visible, with a checkbox for "Hide Auto Scaling alarms". A search bar and filters for "Alarm state: Any", "Alarm type: Any", and "Actions status: Any" are present. A table lists the alarm:

Name	State	Last state update (UTC)	Conditions
High CPU Utilization	Insufficient data	2025-12-02 18:31:56	CPUUtilization >= 50 for 1 datapoints within 5 minutes

### 3. Testing the alarm by creating stress

```
ubuntu@ip-172-31-29-208:~$ sudo apt install -y stress
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
 stress
0 upgraded, 1 newly installed, 0 to remove and 0 not installed.
```

### Final output

The screenshot shows the AWS CloudWatch Alarms console with the "High CPU Utilization" alarm in the "In alarm" state. The "Services" section shows EC2 with a red bar indicating it is in alarm. The "Recent alarms" section shows a graph of CPU utilization over time, with a peak reaching 51.6%.

The screenshot shows an email notification from Jins Thomas (no-reply@sns.amazonaws.com) regarding the "High CPU Utilization" alarm in the US East (N. Virginia) region. The email states that the alarm has entered the ALARM state because the threshold was crossed. The alarm details are as follows:

- Name: High CPU Utilization
- Description: you insatce has reached the threshold value of maxium CPU usage that is 50
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [98.33213803225645 (02/12/25 18:31:00)] was greater than or equal to the threshold (50.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Tuesday 02 December, 2025 18:36:10 UTC
- AWS Account: 595736576116
- Alarm Arn: arn:aws:cloudwatch:us-east-1:595736576116:alarm:High CPU Utilization

### Note

For faster results and testing purpose keep the threshold value less and the time period less.



