

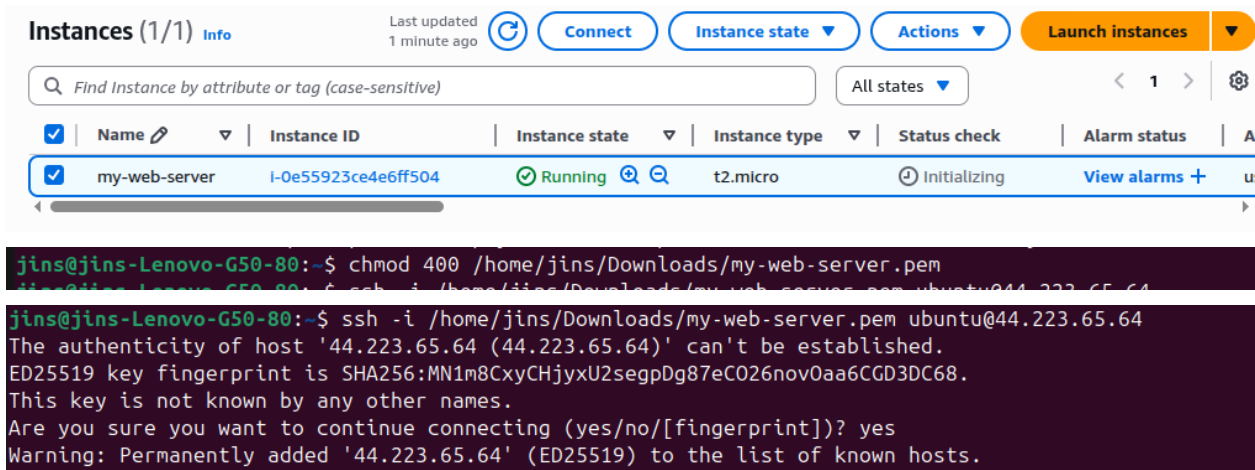
Implementation of CloudWatch CPU Alarm and SNS Email Notification

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

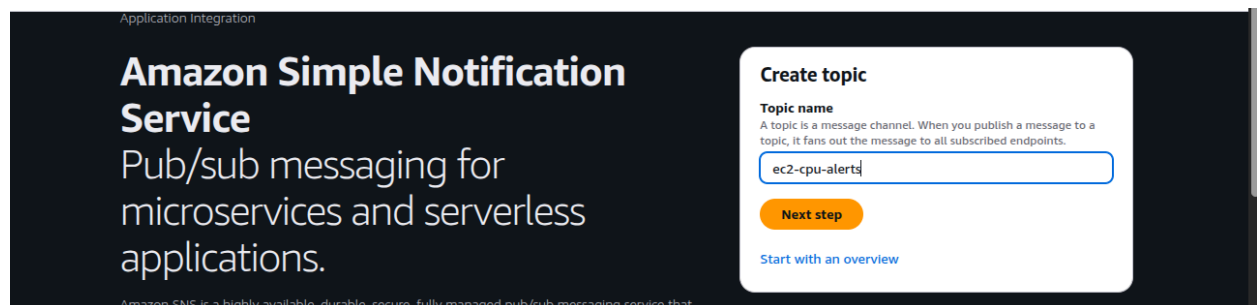
- 1.AWS EC2 instance (t2.micro) with port 22 allowed (allow only your IP) for ssh access and port 80 allowed (for http traffic from the internet).
2. Select OS ubuntu 24.04
- 3.Create key value pair for ssh access change permission to 400 for maintaining best security practice.



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 following commands and output:

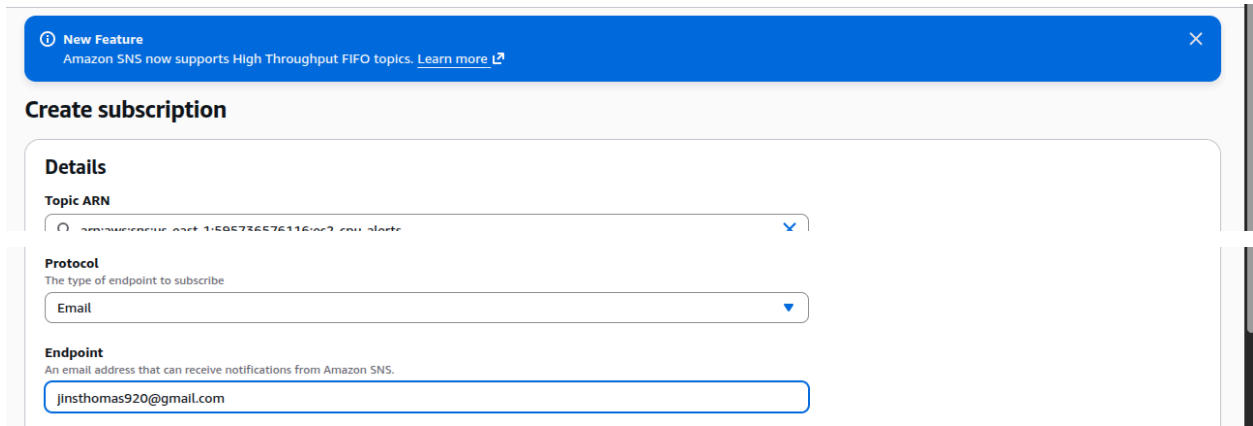
```
jins@jins-Lenovo-G50-80:~$ chmod 400 /home/jins/Downloads/my-web-server.pem
jins@jins-Lenovo-G50-80:~$ ssh -i /home/jins/Downloads/my-web-server.pem ubuntu@44.223.65.64
The authenticity of host '44.223.65.64 (44.223.65.64)' can't be established.
ED25519 key fingerprint is SHA256:MN1m8CxyCHjyxU2seggDg87eC026nov0aa6CGD3DC68.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '44.223.65.64' (ED25519) 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' step is active, with the topic name 'ec2-cpu-alerts' entered. The 'Next step' button is highlighted in orange.

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



Create subscription

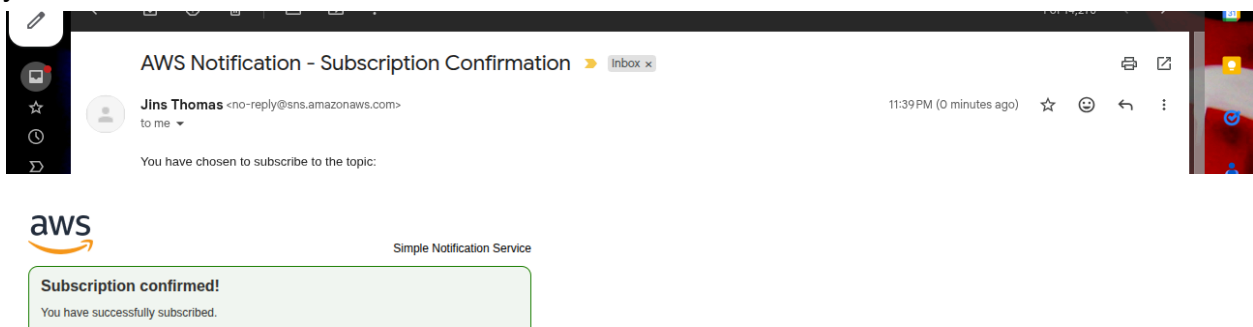
Details

Topic ARN
arn:aws:sns:us-east-1:585735575115:us-east-1-alerts

Protocol
The type of endpoint to subscribe
Email

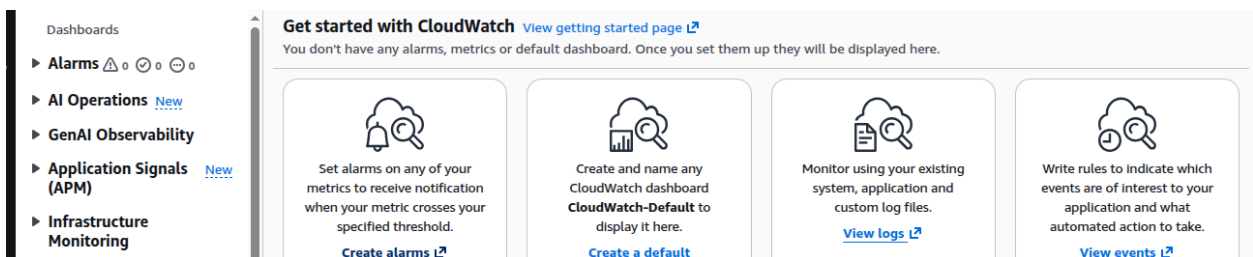
Endpoint
An email address that can receive notifications from Amazon SNS.
jinsthomas920@gmail.com

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

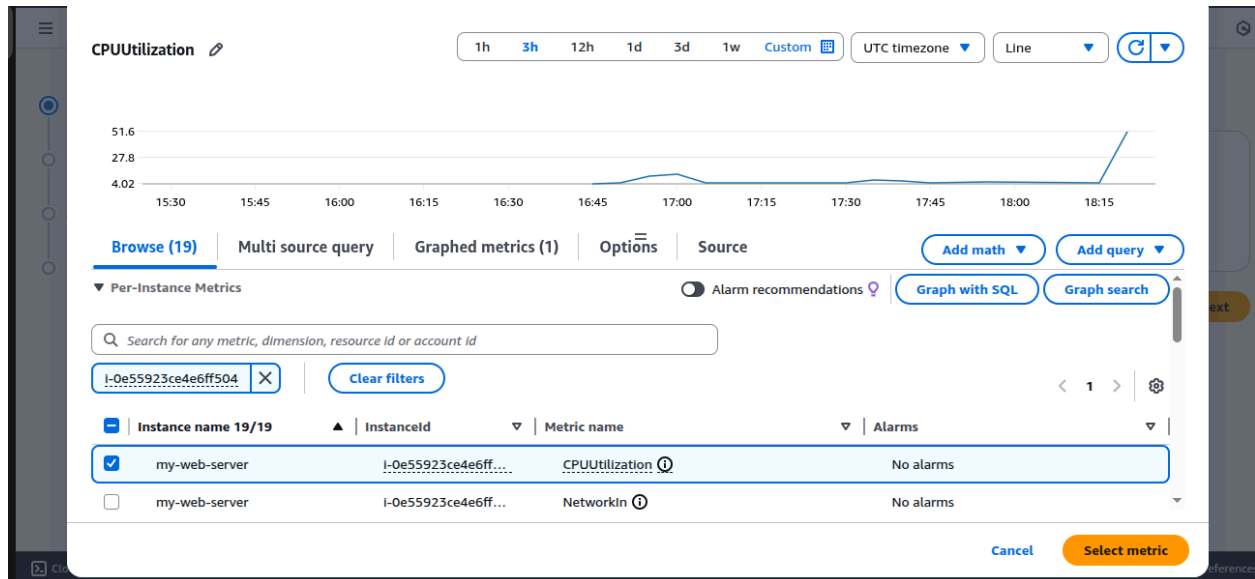


2.Creating the cloudwatch alarm

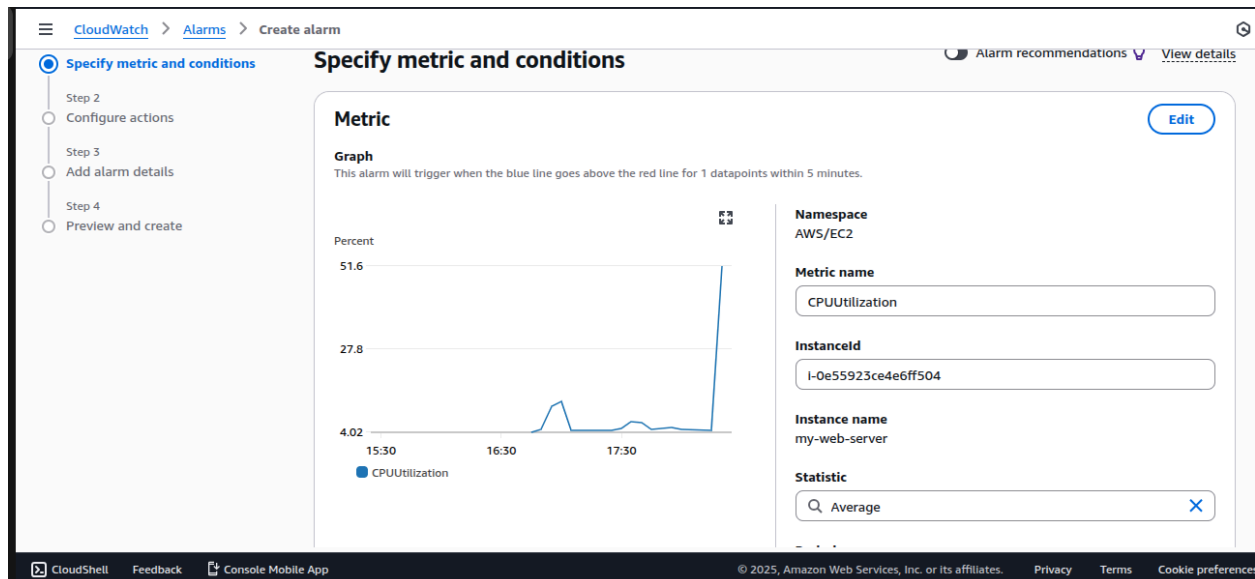
Select create alarm



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.



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 ≥ 50 (this condition is just for hands-on practice and exploring the service).

The screenshot shows the 'Create alarm' page in the AWS CloudWatch console. The 'Conditions' section is active. Under 'Threshold type', 'Static' is selected with the subtext 'Use a value as a threshold'. Under 'Whenever CPU utilization is...', 'Greater/Equal' is selected with the subtext '>= threshold'. The 'than...' section has a text input field containing '50' with the subtext 'Define the threshold value.' and 'Must be a number.' Below this is an 'Additional configuration' section. At the bottom right are 'Cancel' and 'Next' buttons. The footer includes 'CloudShell', 'Feedback', 'Console Mobile App', and copyright information for Amazon Web Services, Inc. or its affiliates.

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' section of the 'Create alarm' page. On the left is a sidebar with steps: 'Specify metric and conditions', 'Configure actions' (selected), 'Add alarm details', and 'Preview and create'. The 'Notification' section is active. Under 'Alarm state trigger', 'In alarm' is selected with the subtext 'The metric or expression is outside of the defined threshold.' Below this, 'Send a notification to the following SNS topic' is selected with the subtext 'Define the SNS (Simple Notification Service) topic that will receive the notification.' The 'Send a notification to...' search box contains 'ec2-cpu-alerts'. Below the search box, it says 'Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.' At the bottom, it shows 'Email (endpoints)' with the email 'jinsthomas920@gmail.com' and a link to 'View in SNS Console'. At the bottom right is a 'Remove' button. The footer is the same as the previous screenshot.

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.

CloudWatch > Alarms > Create alarm

Alarm recommendations available
Turn on Recommendations to pre-populate the wizard with the recommended alarms.

Step 1: Specify metric and conditions
Step 2: Configure actions
Step 3: Add alarm details
Step 4: Preview and create

Add alarm details

Name and description

Alarm name
High CPU Utilization

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

[Edit](#) [Preview](#)

you insatce has reached the threshold value of maximum CPU usage that is 50

Up to 1024 characters (74/1024)

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudWatch > Alarms

CloudWatch

Favorites and recents

Dashboards

▼ **Alarms** 0 0 0 0

In alarm

All alarms

Billing

Successfully created alarm High CPU Utilization. [View alarm](#)

Alarms (1) ☐ Hide Auto Scaling alarms [Clear selection](#) [Create composite alarm](#) [Actions](#) [Create alarm](#)

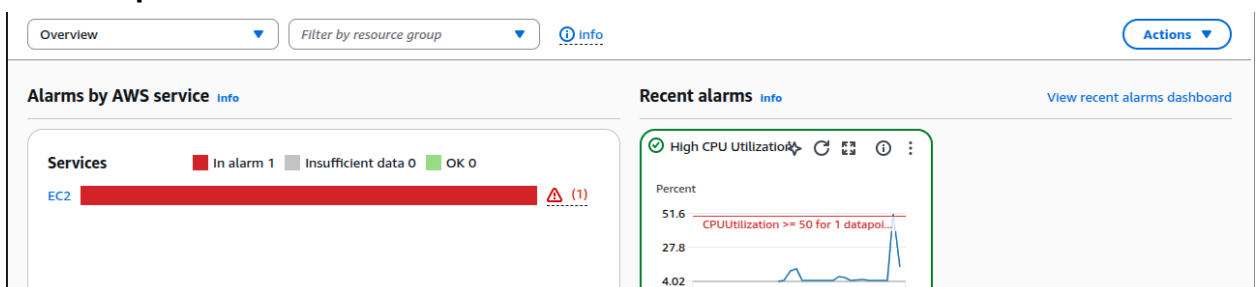
Alarm state: Any Alarm type: Any Actions status: Any

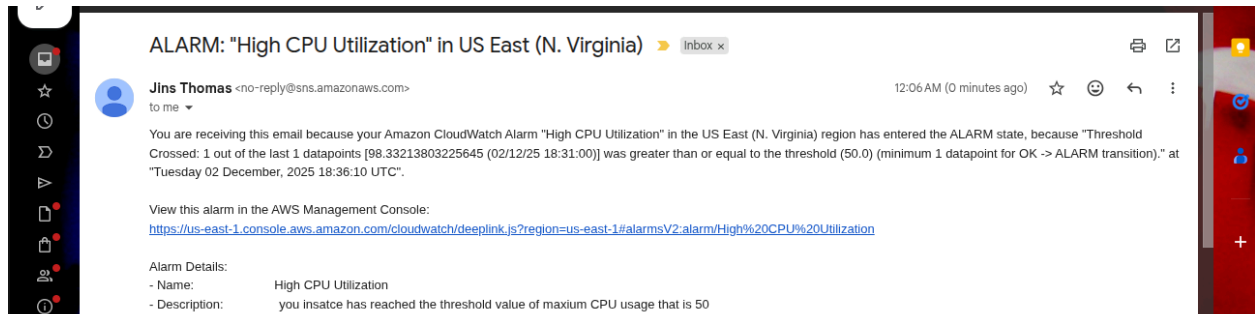
<input type="checkbox"/>	Name	State	Last state update (UTC)	Conditions
<input type="checkbox"/>	High CPU Utilization	Insufficient data	2025-12-02 18:31:56	CPUUtilization >= 50 for 1 datapoints within 5 minutes

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





Note

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

