

Lesson 09 Demo 01

Automating EC2 Instance Shutdown with CloudWatch Alarm

Objective: To create a CloudWatch alarm to automatically stop an instance based on CPU utilization

Tools required: AWS Management Console

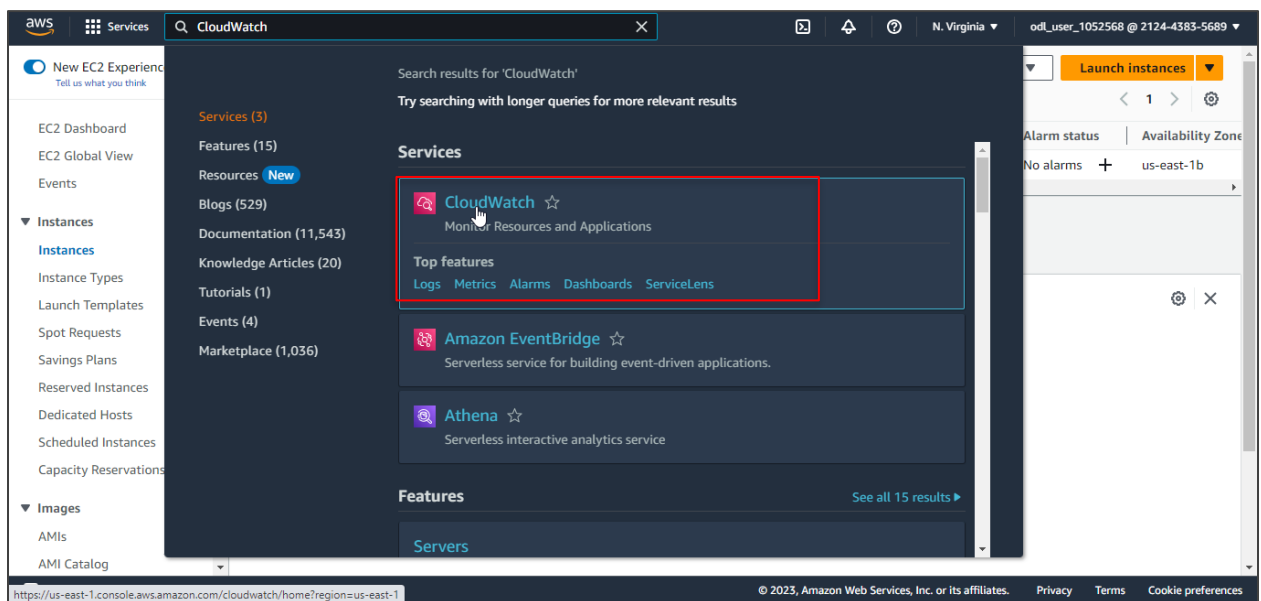
Prerequisites: An active EC2 instance

Steps to be followed:

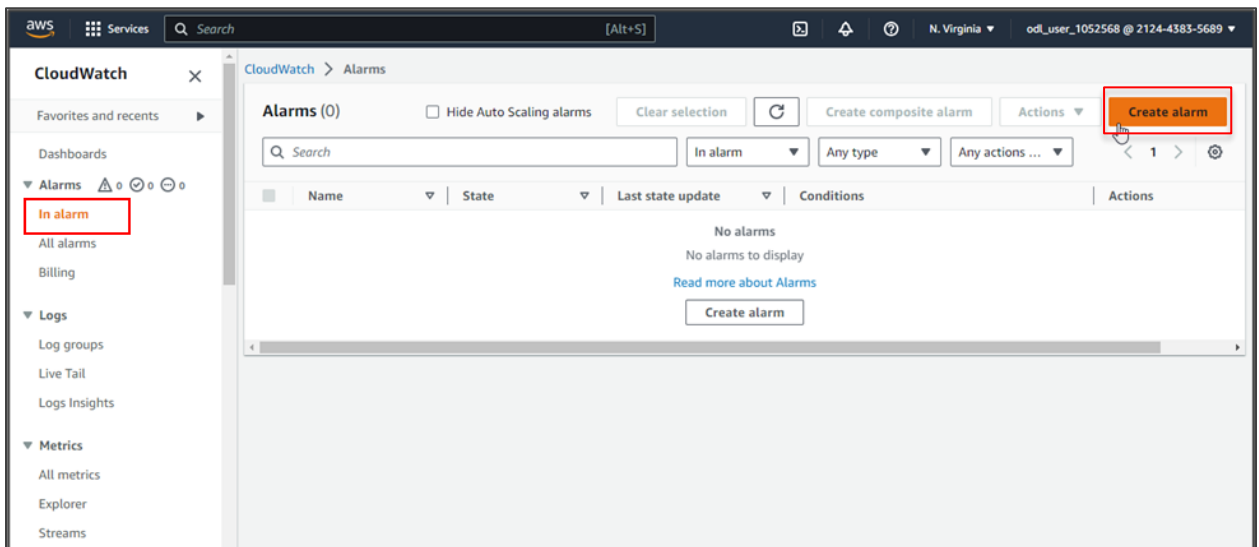
1. Create a CloudWatch alarm

Step 1: Create a CloudWatch alarm

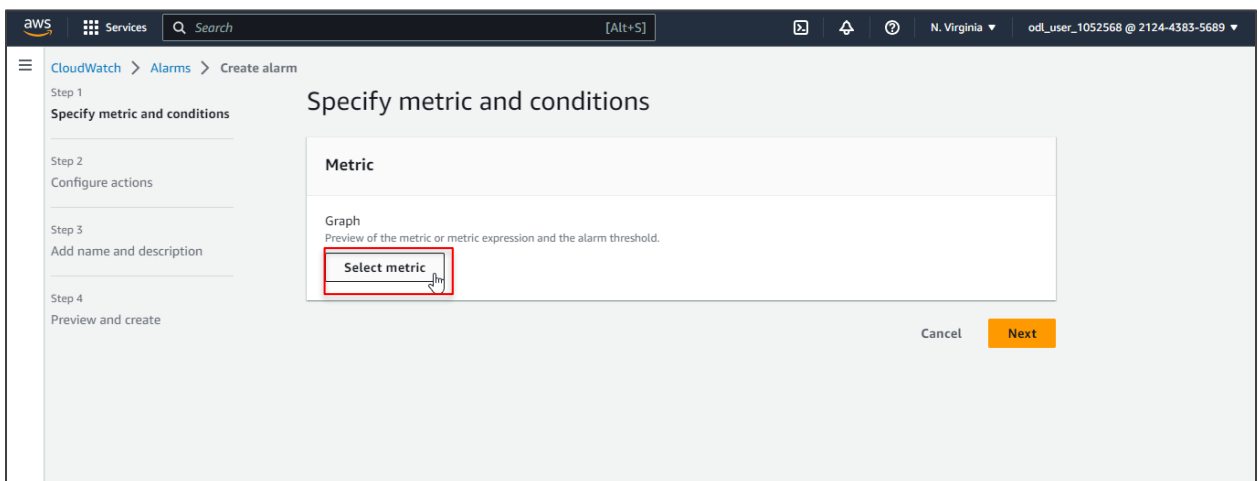
1.1 Open the AWS Management Console, search for and click the **CloudWatch** service



1.2 Go to the **Alarms** section of the left navigation pane and click **Create alarm** under the **In alarm** option



1.3 Click on **Select metric**



1.4 Under the Browse tab, choose **EC2** in the Metrics section, and then select **Per-Instance Metrics**

Select metric

0 04:15 04:30 04:45 05:00 05:15 05:30 05:45 06:00 06:15 06:30 06:45 07:00

Browse Multi source query Graphed metrics Options Source = Add math Add query

Metrics (408) Alarm recommendations Graph with SQL Graph search

N. Virginia Search for any metric, dimension, resource id or account id

EBS 9 **EC2** 17 Events 1 Firehose 2

Cancel Select a single metric to continue

Select metric

0.5 0 16:00 16:15 16:30 16:45 17:00 17:15 17:30 17:45 18:00 18:15 18:30 18:45

Select some metrics to appear here.

Browse Multi source query Graphed metrics Options Source = Add math Add query

Metrics (323) Alarm recommendations Graph with SQL Graph search

N. Virginia All > EC2 Search for any metric, dimension, resource id or account id

By Auto Scaling Group 34 **Per-Instance Metrics** 289

Cancel Select a single metric to continue

1.5 Select the **CPUUtilization** metric of your EC2 instance and click **Select metric**

Select metric

42 — 16:15 16:30 16:45 17:00 17:15 17:30 17:45 18:00 18:15 18:30 18:45 19:00

● CPUUtilization

Browse Multi source query Graphed metrics (1) Options Source = Add math Add query

<input checked="" type="checkbox"/>	My_server_1	i-02333c502c40f5529	CPUUtilization ⓘ	No alarms
<input type="checkbox"/>	My_server_1	i-02333c502c40f5529	DiskReadOps ⓘ	No alarms
<input type="checkbox"/>	My_server_1	i-02333c502c40f5529	NetworkPacketsOut ⓘ	No alarms
<input type="checkbox"/>	My_server_1	i-02333c502c40f5529	MetadataNoToken ⓘ	No alarms
<input type="checkbox"/>	My_server_1	i-02333c502c40f5529	NetworkIn ⓘ	No alarms

Cancel **Select metric**

1.6 In the Conditions section, select **Static** for the Threshold type, set **Lower** for CPUUtilization, enter **30** as the threshold value, and click **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

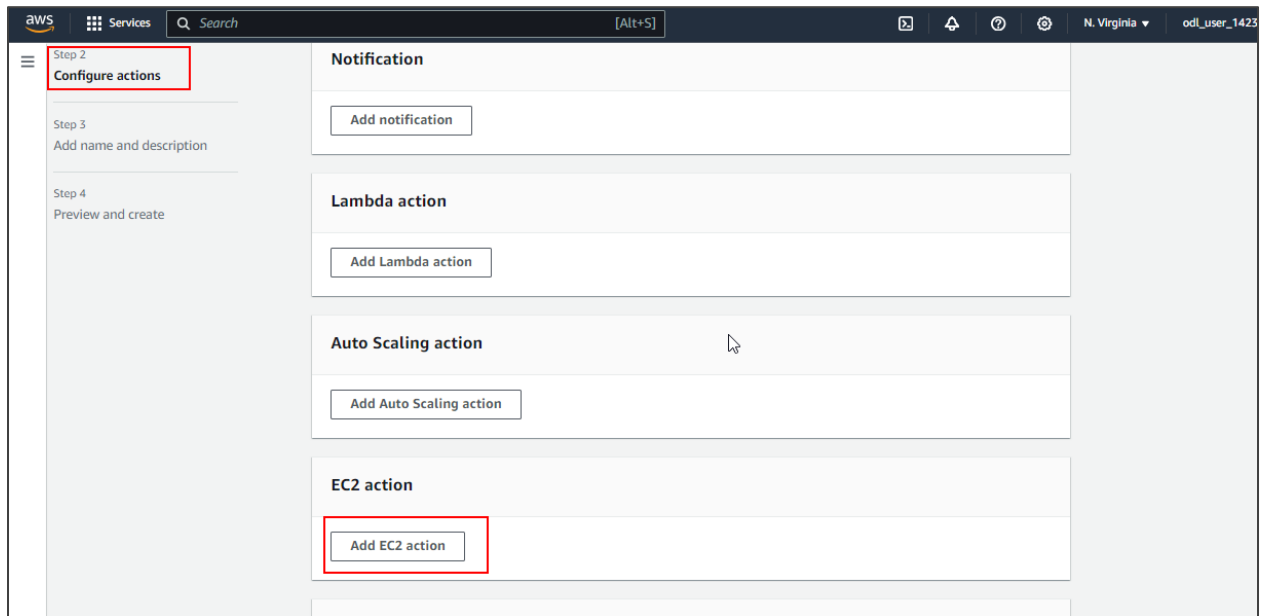
30

Must be a number

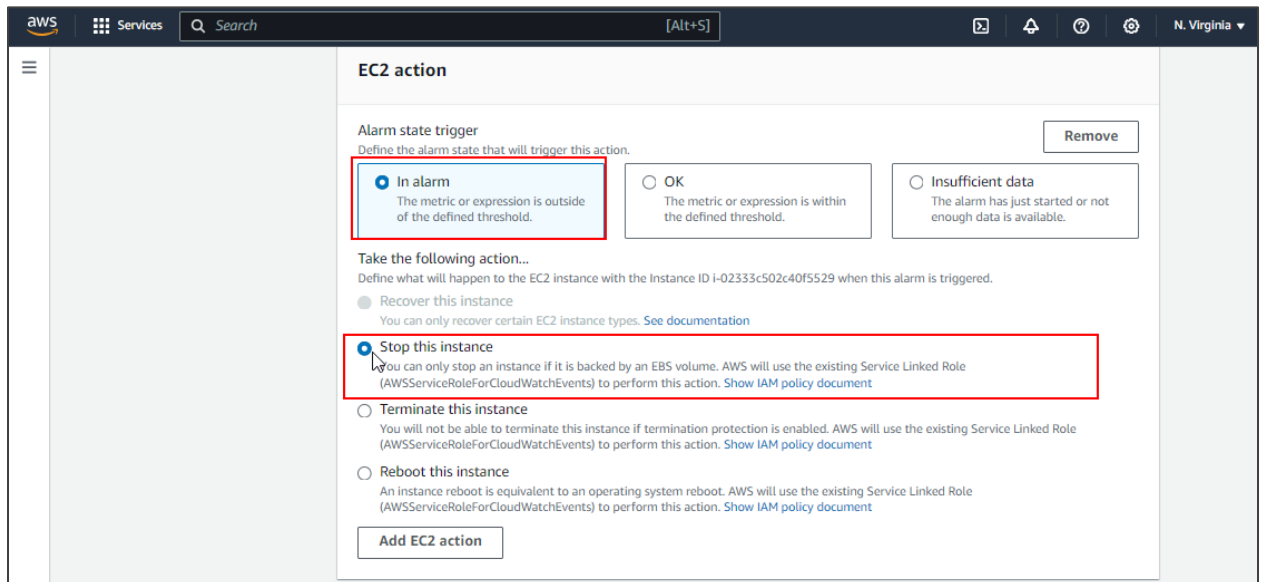
► Additional configuration

Cancel **Next**

1.7 In the Configure actions page, click on **Add EC2 action**



1.8 Select the **In alarm** and **Stop this instance** options



1.9 Click **Next**

☒ **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

Systems Manager action [Info](#) [🔗](#)

This action will create an Incident or Opsitem in Systems Manager when the alarm is **In alarm** state.

Add Systems Manager action

Cancel Previous **Next**

1.10 In the **Add name and description** section, assign a name to your alarm, and click **Next**

aws Services Search [Alt+S] N. Virginia odl_user_1423851 @ 34

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
EC2_CPU_Alert

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

Edit Preview

This is an H1
double asterisks will produce strong character
This is [an example](https://example.com/) inline link.

Up to 1024 characters (0/1024)

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**

1.11 Preview the alarm configurations and click **Create alarm**

Step 3: Add name and description Edit

Name and description

Name
EC2_CPU_Alert

Description
-

Cancel Previous Create alarm

CloudWatch × Successfully created alarm EC2_CPU_Alert. View alarm ×

CloudWatch > Alarms

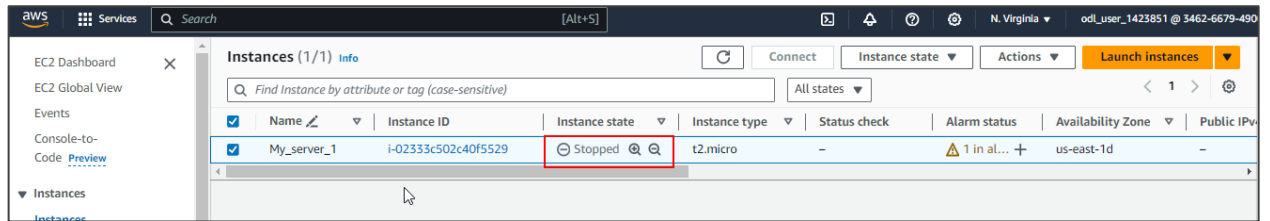
Alarms (1) ☐ Hide Auto Scaling alarms Clear selection ↺ Create composite alarm Actions Create alarm

Search Alarm state: Any Alarm type: Any Actions status: Any < 1 > ⚙

<input type="checkbox"/>	Name	State	Last state update (UTC)	Conditions	Actions
<input type="checkbox"/>	EC2_CPU_Alert	Insufficient data	2024-08-12 19:22:23	CPUUtilization < 30 for 1 datapoints within 5 minutes	Actions enabled

Once the alarm is created, it will be visible on the Alarms dashboard.

1.12 Navigate to the EC2 dashboard to check the instance state and alarm status. After a while, you will notice that the alarm triggers, causing the instance to stop.



By following these steps, you have successfully created a CloudWatch alarm that monitors the CPU utilization of your EC2 instance. The instance is automatically stopped when the CPU utilization falls below the specified threshold of 30%.