

## Table of contents

- [Table of contents](#)
- [Introduction to Cloudwatch :](#)
  - [Cloudwatch metrics](#)
  - [SNS](#)
    - [Creating a topic in SNS](#)
  - [Cloudwatch alarms](#)
    - [Creating cloudwatch alarms](#)
  - [Cloudwatch Events and rules](#)
    - [Creating cloudwatch rule](#)
  - [Cloudwatch Logs](#)
    - [Creating cloudwatch log group](#)
    - [Insights](#)
  - [Cloudwatch dashboard](#)
  - [FAQ](#)

## Introduction to Cloudwatch :

- Cloudwatch is the go to monitoring service for AWS .
- It integrates with almost every service and pushes put metrics which gives us insight on how the resource is performing
- Cloudwatch has pre-built metrics from we can monitor key performance attribute like cpu utilization etc

### Cloudwatch metrics

- All the graphs and charts we see for the resources in aws are cloudwatch metrics
- In order to see the metric , we can navigate to the cloudwatch service. On the left index select metrics.
- In search we can enter reources for ex instance id , and we can can get the graphs related to it
- Cloudwatch basic metrics include these graphs getting refreshed every 5 minutes . This refresh point is what we refer as datapoints
- With cloudwatch enhanced monitoring , these graphs get refreshed after every 1 minute.

### SNS

- SNS is simple notification service which helps send notifications to bunch of subscribers
- In SNS we need to create a topic to which notifications are sent
- Whoever subscribes to that topic will receive notifications
- SNS works on Pub-sub model

### Creating a topic in SNS

- Go to SNS , Click on create Topic
- Enter a name and dscription , keep all the settings as default . And click on create topic
- Once a topic is created , we need to add subscribers
- Click on subscriptions , and click on create subscriptions

- Select the topic we just created
- Select the protocol as email, and enter the email id we wish to send notification to
- Click on create subscription
- The email owner needs to confirm the subscription from the inbox . Once confirmed , all the notifications sent to the topic will be received by the subscriber

## Cloudwatch alarms

- Cloudwatch alarms are a way of triggering an alert or small actions based on a cloudwatch metric .
- These alarms are useful for monitoring where we cant continuously observe a metric

## Creating cloudwatch alarms

- Navigate to cloudwatch service and click on alarms and create new alarm
- Select a metric based on which we want to create an alarm . In this instance we will select CPU utilization

CloudWatch > Alarms > Create alarm

Step 1  
Specify metric and conditions

Step 2  
Configure actions

Step 3  
Add name and description

Step 4  
Preview and create

### Specify metric and conditions

**Metric** Edit

**Graph**  
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.

1  
0.8  
0.6  
0.4  
0.2  
0

17:30 18:30 19:30

■ CPUUtilization

Namespace  
AWS/EC2

Metric name  
CPUUtilization

InstanceId  
i-0f84e387686d585df

Instance name  
No name specified

Statistic  
Average

Period

- Under the conditions , let us keep the threshold type as static . Below that "Whenever CPUUtilization is " , to be kept at the default value i.e. greater than
- Let us keep the threshold value as 5 since we want to test it .
- The additional configuration states how many datapoints need to be breached in order for alarm to trigger
- Click Next
- For notification settings , we'll choose SNS , which is a notification service by AWS . Let us select the topic we had created earlier
- Alarm state trigger we'll keep it as In alarm
- Auto scaling action we'll be seeing in upcoming sessions . Let us skip it for now
- EC2 Action is again optional , where we have option of stopping terminating or rebooting the instance once the alarm is triggered . Let us select the stop the instance action

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.

Take the following action...

Define what will happen to the EC2 instance with the Instance ID i-0f84e387686d585df 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](#)

 Failed to check if the instance is recoverable

Add EC2 action

Cancel

Previous

Next

- Let us add name and description for the alarm and create it.
- Once created keep observing the metric , whenever it goes beyond 5 , you should be receiving the alert and the instance should automatically stop

## Cloudwatch Events and rules

- Cloudwatch event is basically any change in AWS resource . for ex- starting an EC2 instance
- Cloudwatch Rule lets us define an automated action on the said event . Cloudwatch rule is similar to alarms but is not dependent on any graph to trigger also it integrates with other services for its actions like lambda ,ssm etc .
- Common use case for cloudwatch rule is to create notifications or actions for any api call made to the aws .

## Creating cloudwatch rule

- Click on create rule
- For a rule , there are two types of Event sources . Event pattern and Schedule
- Event pattern is based on any api call that is made in the aws wherein schedule lets us define cron expression (we'll see it in upcoming linuxsession ) which schedules it for a specific time period for ex . occurs every monday , or every day etc
- For testing we'll go with event pattern
- In service name let us select EC2
- in Event type "EBS snapshot notification "
- Keep other settings as shown in snippet below

Build or customize an Event Pattern or set a Schedule to invoke Targets.

☒ Event Pattern ⓘ ☐ Schedule ⓘ

**Build event pattern to match events by service**

Service Name: EC2

Event Type: EBS Snapshot Notification

☐ Any event ☒ Specific event(s)

☒ createSnapshot

☐ Any result ☒ Specific result(s)

☒ succeeded

☒ Any source ☐ Specific source(s)

☒ Any snapshot ID ☐ Specific snapshot ID(s) by ARN

Select Target to invoke when an event matches your Event Pattern or when schedule is triggered.

**Add target\***

- Observe that based on the settings you chose there is automatically a json created

```
{
  "source": [
    "aws.ec2"
  ],
  "detail-type": [
    "EBS Snapshot Notification"
  ],
  "detail": {
    "event": [
      "createSnapshot"
    ],
    "result": [
      "succeeded"
    ]
  }
}
```

- Now that we have the event pattern ready let us configure the right hand side i.e. target
- This target can be services like lambda or ssm which helps us automate things . As of now let us select SNS
- Select the topic and click on configure details
- Grant it an appropriate name and click on create
- Now test the rule by creating an EBS snapshot . We should ideally be receiving an notification once the snapshot succeeds

## Cloudwatch Logs

- Till now we have seen many services needs to store logs and S3 is the most common option
- However cloudwatch logs is also an option , even though it is less widely used as compared to S3
- Services like Lambda, VPC flowlogs , cloudtrail can send logs to cloudwatch logs .

- These logs can later be queried from cloudwatch insights
- Logs are sent to logical folder called log group
- The flow of logs sent inside the log group is termed as log stream
- The option to send the logs to cloudwatch logs can be seen while enabling VPC flowlogs or creating a trail in cloudtrail

### Creating cloudwatch log group

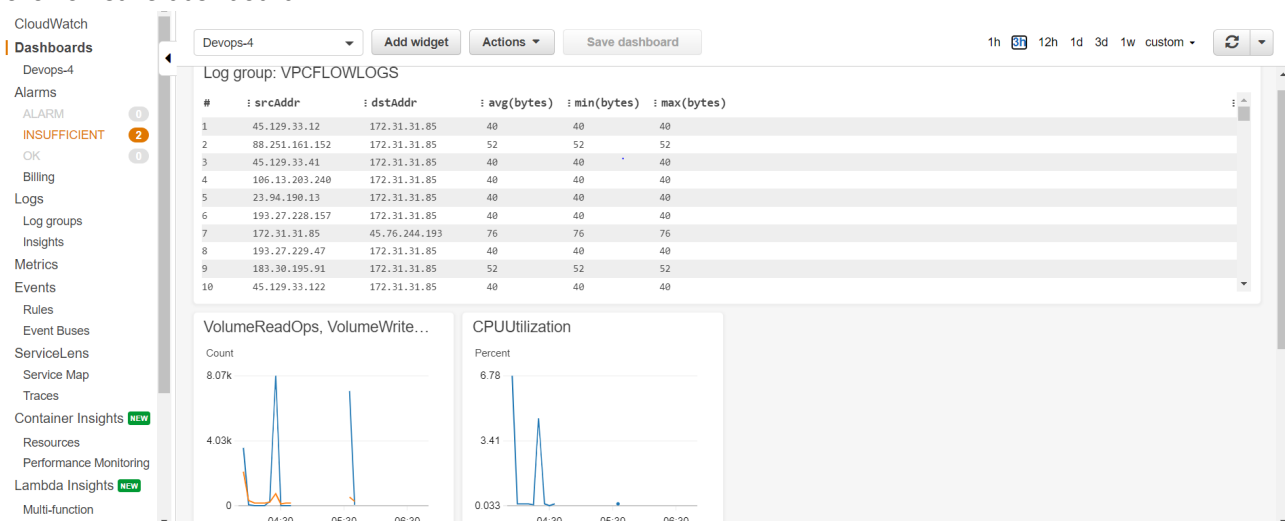
- We can create our own cloudwatch log group and chose to push log streams such as vpc flowlogs in that .
- Go to log groups and click on create log-group .
- Give it an appropriate name and create the log group
- Navigate to VPC flowlogs , follow the usual steps to enable it . Instead of S3 however , enter the cloudwatch log option and select the group we created
- This will need IAM roles , which could be done from "setup permissions tab"

### Insights

- Just like athena helps us query data in S3 , insights works the same for cloudwatch logs
- Once you go to insights , select the log group you wish to query . for ex vpc flowlogs log group that we had created earlier
- on the right panel , we can see a query tab where we will be able to see sample query tab
- Copy any of the sample queries and run it on the console . for ex : "stats avg(bytes), min(bytes), max(bytes) by srcAddr, dstAddr "

### Cloudwatch dashboard

- Cloudwatch dashboard is way of centralized monitoring of all the important resource metrics . This can also include results of the insights queries as well
- The dashboard can be created from the console itself where you need to navigate to dashboard and click on create dashboard
- Give it a name and click on create dashboard
- once created , click on dashboard and lclick on add widget . From widget add the required graph and click on save dashboard



- Similarly when you get the results of an insights queries , these can be directly added to the dashboard from the console

## FAQ

- It is very easy to get confused with cloudtrail and cloudwatch . Remember cloudtrail is for auditing and cloudwatch is meant for monitoring purpose
- Cloudwatch rules is one of the primary ways of automating actions on aws