**CloudWatch Agent**

**Standard Operating Procedures**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Topic No.** | **Name of Topic** | **Page**  **No.** |
| **1** | Introduction | **3** |
| **2** | Steps before initiating the CloudWatch Agent | **4** |
| **3** | CloudWatch Agent Plan | **5** |
| **4** | Considerations before executing this plan | **6** |
| **5** | Steps to execute the CloudWatch Agent | **7** |
| **6** | CloudWatch Agent Plan Activated on | **14** |
| **7** | References | **15** |
| **8** | Download the CloudWatch agent package on different O.S. | **16** |

# Introduction

The CloudWatch agent is a tool used to monitor and collect metrics and log data on Amazon EC2 instances, on-premises servers, and other cloud resources, such as Amazon ECS or AWS Lambda functions. The agent runs as a daemon or Windows service on your instances and can be configured to collect both system-level metrics and custom application logs.

The CloudWatch agent is installed on the instances or resources that you want to monitor and are configured to collect desired data. It then seamlessly sends this data to CloudWatch, where you can view and analyze it using CloudWatch console, CloudWatch metrics API, CloudWatch alarms, and CloudWatch dashboards.

Using the CloudWatch agent, you can monitor a variety of resources, including CPU usage, memory utilization, disk space, network traffic, custom application metrics, service logs, operating system logs, and more. You can also configure CloudWatch alarms to notify you if certain metrics exceed pre-defined thresholds.

Once you have collected your metrics and logs, you can use the CloudWatch console to view and analyze the data, generate alerts, or create custom dashboards that visualize important metrics.

The CloudWatch agent is a versatile and powerful tool that can help you effectively monitor your resources in the cloud and on-premises. By using the agent, you can gain insights into the performance and health of your resources, and respond to issues in a timely and efficient manner before they impact users.

# Steps before initiating the CloudWatch Agent

Before initializing the CloudWatch agent, there are several steps you should take to ensure that your AWS environment is properly configured for monitoring:

Create an IAM role for the CloudWatch agent: You will need to create an IAM role that allows the agent to access your Amazon EC2 instances and publish metrics and logs to CloudWatch.

Install the CloudWatch agent on the EC2 instances: Install the CloudWatch agent on each EC2 instance that you want to monitor.

Configure the CloudWatch agent: Configure the CloudWatch agent to collect the metrics and logs that you want to monitor. This can include system metrics, custom metrics, and logs.

Set up CloudWatch alarms: Create CloudWatch alarms based on the metrics that you are collecting to notify you when specific thresholds are reached.

Set up CloudWatch dashboards: Create CloudWatch dashboards to visualize and monitor the metrics that you are collecting. You can use the dashboard to monitor the overall health of your systems and to identify issues before they become critical.

Verify CloudWatch agent metrics and logs: Verify that the CloudWatch agent is collecting the metrics and logs that you want. You can view the metrics and logs in the CloudWatch console or by using the CloudWatch API.

Monitor CloudWatch agent performance: Monitor the performance of the CloudWatch agent to ensure that it's collecting the right data and that it's working correctly. You should also periodically update the CloudWatch agent to stay up-to-date with the latest features and bug fixes.

Troubleshoot: Troubleshoot issues by analyzing the logs collected by the agent and use the data to resolve any issues.

Once the agent is initialized, you can collect and monitor metrics and logs to gain insights into your systems and identify potential issues before they impact your users.

# CloudWatch Agent Plans

The CloudWatch agent is a software tool that allows you to monitor system-level metrics and collect log files from Amazon EC2 instances, on-premises servers, and other cloud resources. The agent runs as a daemon or Windows service on your instances and is designed to collect both custom application logs and system-level metrics.

The CloudWatch agent allows you to monitor a wide range of metrics, including CPU usage, memory utilization, network traffic, and disk usage, as well as custom application metrics. With the CloudWatch agent, you can also collect log data from your applications and services, including web servers, application servers, and databases.

You can configure the CloudWatch agent to collect metrics and logs at different intervals, and it can also send alarms and notifications when certain metrics exceed pre-defined thresholds. The agent also supports the collection of metrics and logs from sources outside of EC2 instances, including on-premises servers, AWS Lambda functions, and containers running on Amazon ECS.

The CloudWatch agent can be used to monitor the performance and health of your resources in real-time, allowing you to identify potential issues before they impact your applications or services. You can use the CloudWatch console to view and analyze the data collected by the agent, create custom dashboards to track specific metrics, and generate alarms and notifications when specific thresholds are exceeded.

In summary, the CloudWatch agent is a powerful tool that allows you to monitor system-level metrics and collect log files from Amazon EC2 instances, on-premises servers, and other cloud resources. It provides real-time visibility into the performance and health of your resources, enabling you to identify issues and troubleshoot in a timely and efficient manner.

# Consideration before executing this plan

Before executing the CloudWatch agent, there are several considerations to keep in mind to ensure successful deployment and monitoring of your resources. Some of these considerations are

**Compatibility**:

Ensure that the CloudWatch agent is compatible with the operating system of the instances that you want to monitor.

**Resource allocation:**

Consider if monitoring instances at a high rate will impact the performance of your instances, and adjust monitoring frequency and logging levels accordingly.

**Configuration:**

Plan ahead and configure the CloudWatch agent to monitor only the resources that need to be monitored, as well as tailor the configuration to your specific needs. This will avoid unnecessary data collection that could increase costs or cause unnecessary clutter.

**Data retention:**

Determine the amount of time you will need to retain your CloudWatch logs, and select an appropriate retention policy. Also, ensure that you have enough S3 storage space if you plan to store logs for an extended time.

**Security**:

Configure the agent to encrypt data in transport and at rest to ensure data security. If logging sensitive information, enable VPC flow logs and AWS Config for additional security monitoring.

**Network connectivity:**

Ensure that your instances have network connectivity to the appropriate ports to allow communication with CloudWatch.

**Cost:**

Take into account the cost implications of monitoring your resources and adjust your policy and monitoring frequency to suit your budget

# 5.Steps to execute the CloudWatch Agent: (Please refer Figures for more details)

Identify the server you need to execute CloudWatch Agent verify if it falls under server information table for particular project

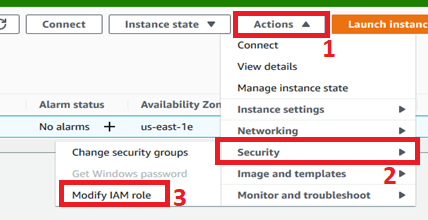
# **Step 1**: **Open the AWS Console and go to EC2 Service and Launch an EC2 with Ubuntu AMI 20.04**

# Step 2: **Attach an IAM role with EC2 Instance**

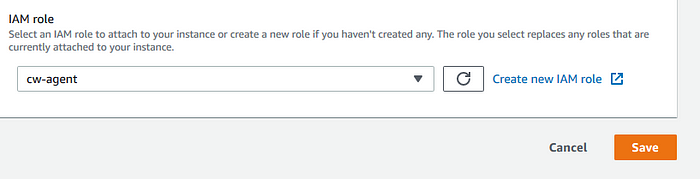
Create an IAM role. In the list of policies, select **CloudWatchAgentServerPolicy &** create an IAM role.

****

Now attach an IAM role with EC2 Instance. Go to EC2 — Select Instance — Click on Action — Security — Modify IAM role

****

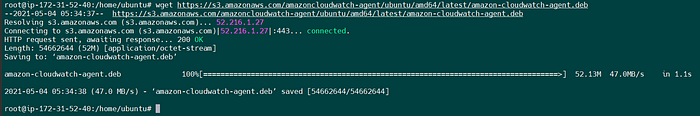
Choose create IAM role & save it.



# Step 3**: Download the CloudWatch Agent Package apt install amazon-**CloudWatch**-agent**

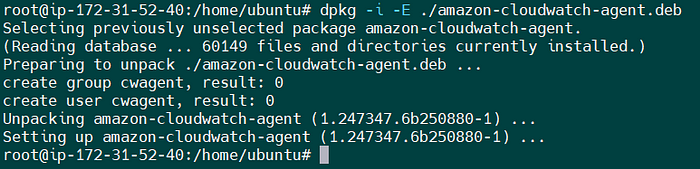
***apt-get install amazon-CloudWatch-agent***

***wget*** [*https://s3.amazonaws.com/amazonCloudWatch-agent/ubuntu/amd64/latest/amazon-CloudWatch-agent.deb*](https://s3.amazonaws.com/amazoncloudwatch-agent/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb)



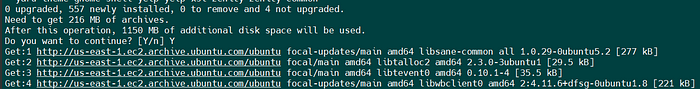
**Install the package:**

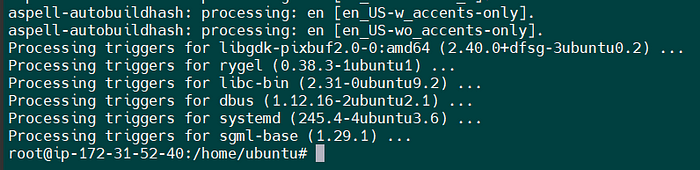
dpkg ***-i -E ./amazon-CloudWatch-agent.deb***



**Update Packages: (**This will take a few minutes if you haven’t updated your available updates prior**)**

***apt-get update***





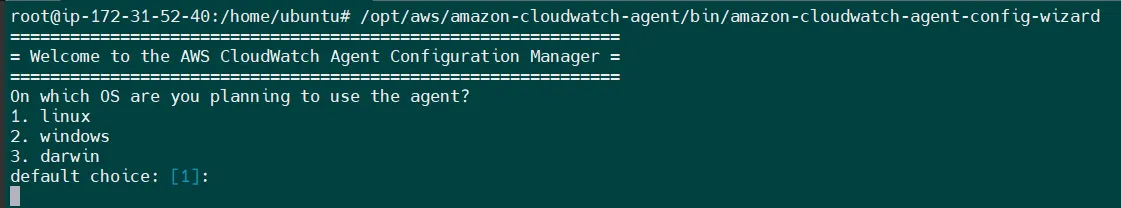
# Step 4**: Create the CloudWatch Agent Configuration File**

Before running the CloudWatch agent on any servers, you must create a CloudWatch agent configuration file. The agent configuration file is a JSON file that specifies the metrics and logs that the agent is to collect, including custom metrics

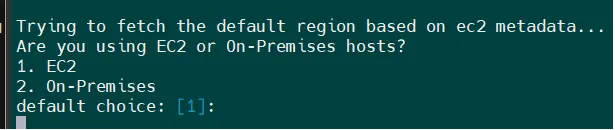
The agent configuration file wizard

amazon-CloudWatch-agent-config-wizard

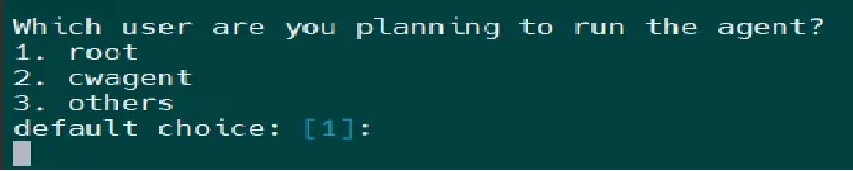
**/opt/aws/amazon-CloudWatch-agent/bin/amazon-CloudWatch-agent-config-wizard**



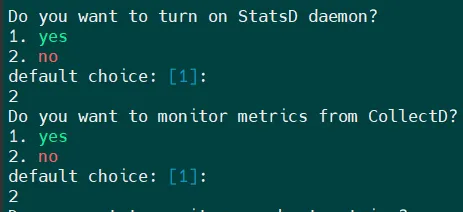
Using EC2 Instance so Option 1



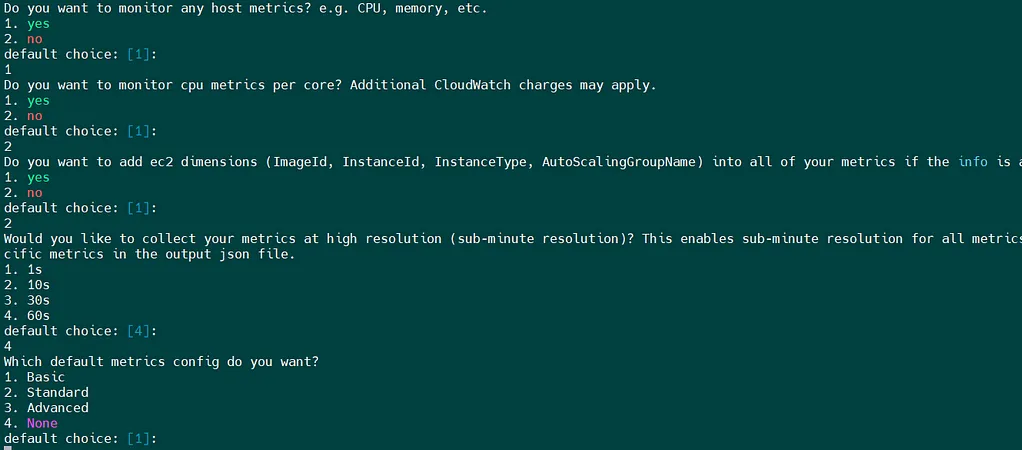
Select an user



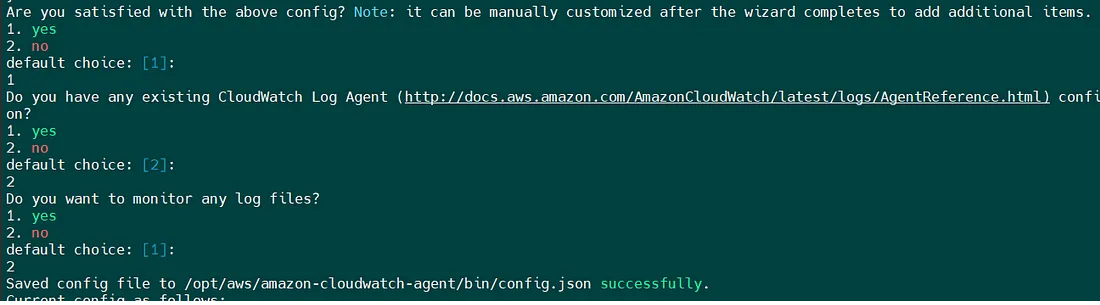
We can skip this two option for memory metric.



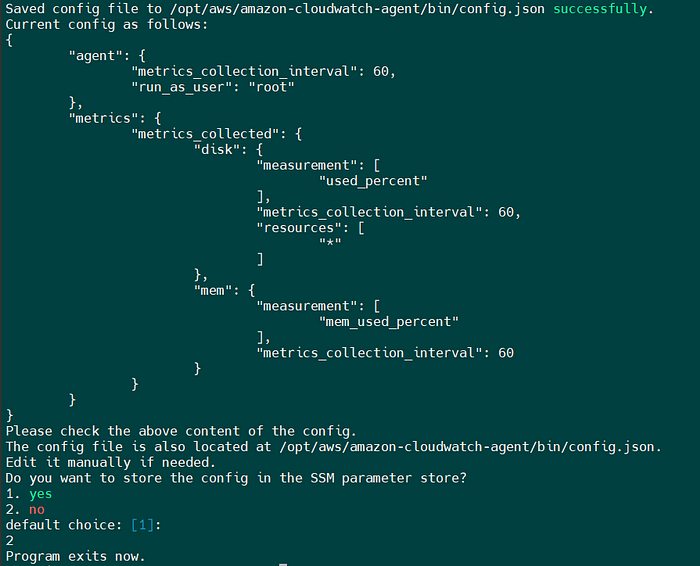
We can choose below options according to our requirements.



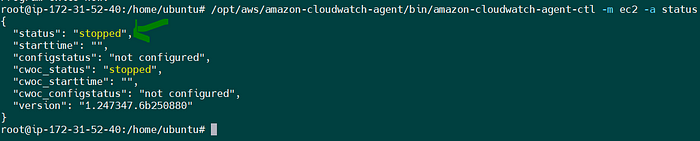
Provide some declarations for config files.



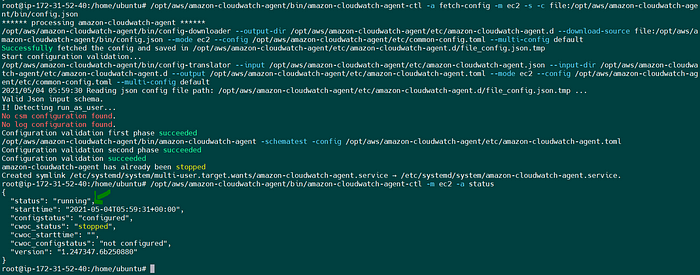
We can check json file under /opt/aws/amazon-CloudWatch-agent/bin/config.json



Now check the status of CW agent. ***/opt/aws/amazon-CloudWatch-agent/bin/amazon-CloudWatch-agent-ctl -m ec2 -a status***

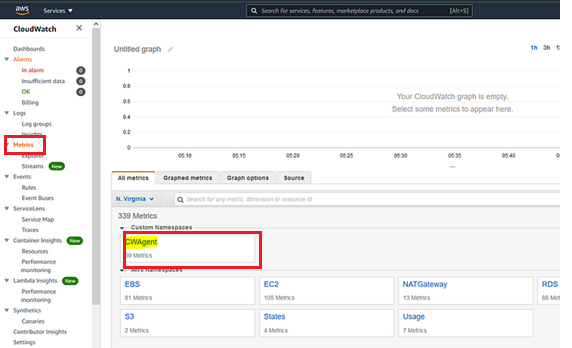


Status is stopped now, so start it & check status after that. ***/opt/aws/amazon-CloudWatch-agent/bin/amazon-CloudWatch-agent-ctl -a fetch-config -m ec2 -s -c file:/opt/aws/amazon-CloudWatch-agent/bin/config.json***

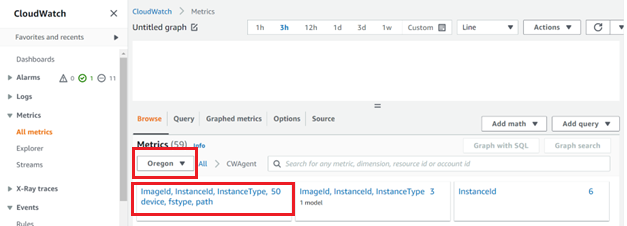


# Step 5**: Check Custom Metrics in AWS CloudWatch:**

Custom Matrix

****

Check inside for memory metrics:

****

# List of Instances with CloudWatch agent

|  |  |  |
| --- | --- | --- |
| **Instance ID** | **Server name** | **Activation** |
| i-095bee6b84f361255 | Beta Grocerybabu larvael | yes |

# Reference Link

For more details refer the link:

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/install-CloudWatch-Agent-on-EC2-Instance.html>

[https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/create-CloudWatch-agent-configuration-file.html](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/create-cloudwatch-agent-configuration-file.html)

[https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/create-CloudWatch-agent-configuration-file-wizard.html](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/create-cloudwatch-agent-configuration-file-wizard.html)

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/install-CloudWatch-Agent-on-EC2-Instance-fleet.html#start-CloudWatch-Agent-EC2-fleet>

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/troubleshooting-CloudWatch-Agent.html>

# Download the CloudWatch agent package on different OS

sudo yum install amazon-CloudWatch-agent

Amazon Linux and Amazon Linux 2

[https://s3.amazonaws.com/amazonCloudWatch-agent/amazon\_linux/amd64/latest/amazon-CloudWatch-agent.rpm](https://s3.amazonaws.com/amazoncloudwatch-agent/amazon_linux/amd64/latest/amazon-cloudwatch-agent.rpm)

Redhat

[https://s3.amazonaws.com/amazonCloudWatch-agent/redhat/amd64/latest/amazon-CloudWatch-agent.rpm](https://s3.amazonaws.com/amazoncloudwatch-agent/redhat/amd64/latest/amazon-cloudwatch-agent.rpm)

macOS

[https://s3.amazonaws.com/amazonCloudWatch-agent/darwin/amd64/latest/amazon-CloudWatch-agent.pkg](https://s3.amazonaws.com/amazoncloudwatch-agent/darwin/amd64/latest/amazon-cloudwatch-agent.pkg)

Windows

[https://s3.amazonaws.com/amazonCloudWatch-agent/windows/amd64/latest/amazon-CloudWatch-agent.msi](https://s3.amazonaws.com/amazoncloudwatch-agent/windows/amd64/latest/amazon-cloudwatch-agent.msi)

THANK YOU