Placement Empowerment Program

Cloud Computing and DevOps Centre

Set Up a Cloud-Based Monitoring Service: Enable basic cloud monitoring (e.g., CloudWatch on AWS). View metrics like CPU usage and disk I/O for your cloud VM.

Name: CHANDRU S

Department: INFORMATION TECHNOLOGY

St. JOSEPH'S COLLEGE OF ENGINEERING | St. JOSEPH'S INSTITUTE OF TECHNOLOGY

AUTONOMOUS INSTITUTIONS, AFFILIATED TO ANNA UNIVERSITY

Introduction

Effective monitoring is essential in cloud computing to ensure the performance, reliability, and availability of cloud resources. **AWS CloudWatch** offers a powerful monitoring solution that provides real-time insights into AWS services and infrastructure.

This **Proof of Concept (PoC)** explores how to use **CloudWatch** to monitor an **EC2 instance**, focusing on key performance metrics such as **CPU utilization** and **disk I/O**. The PoC demonstrates how to enable monitoring, access real-time data, and analyze system health, helping users optimize cloud-based virtual machines for better performance and reliability.

Overview

This Proof of Concept (PoC) demonstrates how to set up AWS CloudWatch for monitoring an EC2 instance, ensuring optimal performance and resource management. The key steps include:

1. Enabling basic monitoring for an EC2 instance using CloudWatch.
2. Viewing key performance metrics, such as CPU utilization and disk read/write operations, to assess instance health.
3. Utilizing real-time insights from CloudWatch to detect performance bottlenecks and potential issues before they impact services.

By completing this PoC, users will gain a practical understanding of how to integrate CloudWatch monitoring with EC2 instances, enabling proactive performance management and efficient cloud resource utilization.

Objectives

The primary objective of this Proof of Concept (PoC) is to enable AWS CloudWatch monitoring for an EC2 instance and analyze key performance metrics. Specific goals include:

1. **Activating CloudWatch Monitoring** – Enable monitoring for an EC2 instance to track resource utilization.
2. **Analyzing Key Metrics** – View and interpret **CPU usage** and **disk I/O** data to assess instance performance.
3. **Understanding Real-time Monitoring** – Utilize CloudWatch to gain visibility into cloud resource health and performance trends.

Importance

1. **Performance Monitoring** – CloudWatch provides insights into CPU usage, disk I/O, and network traffic, helping to identify and troubleshoot performance issues.

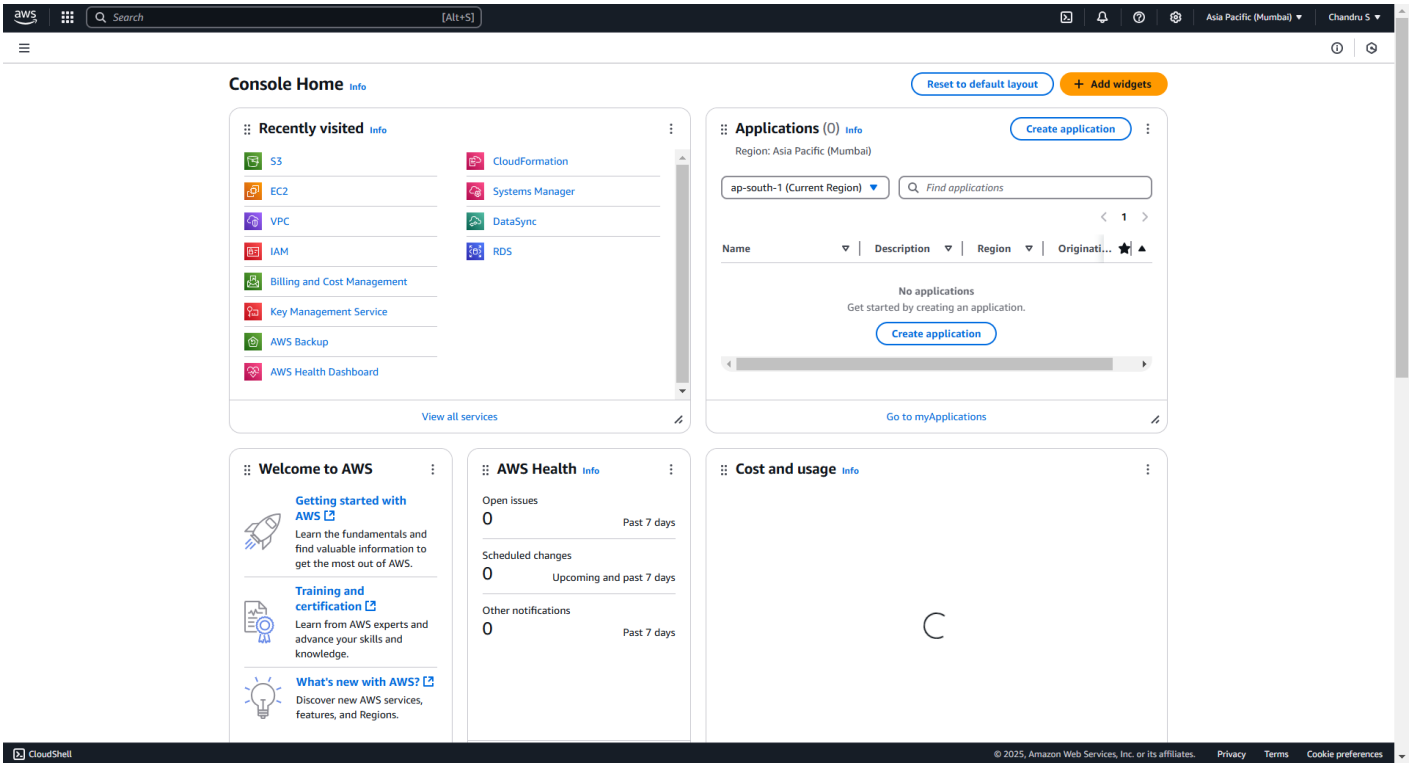
2. **Real-time Visibility** – Enables administrators to access live performance data, ensuring quick responses to fluctuations in resource consumption to prevent downtime.
3. **Efficient Resource Management** – Helps optimize instance capacity and resource allocation, leading to improved efficiency and potential cost savings.
4. **Proactive Issue Detection** – Monitors system behaviour and patterns, allowing early detection of anomalies and performance bottlenecks before they impact operations.

This PoC equips users with hands-on experience in integrating CloudWatch monitoring with EC2, fostering better cloud performance management and reliability.

Step-by-Step Overview

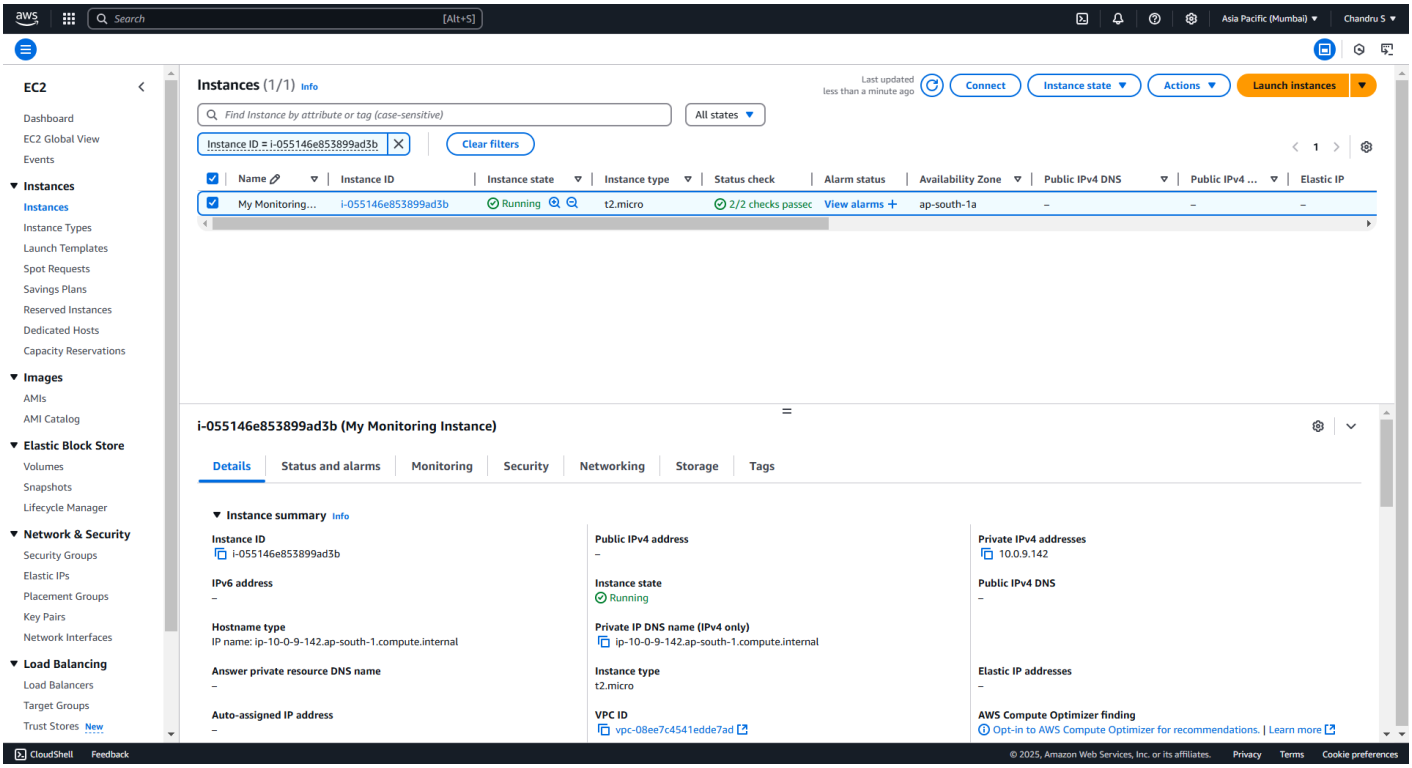
Step 1:

- Open the AWS Management Console.
- Enter your username and password to log in.



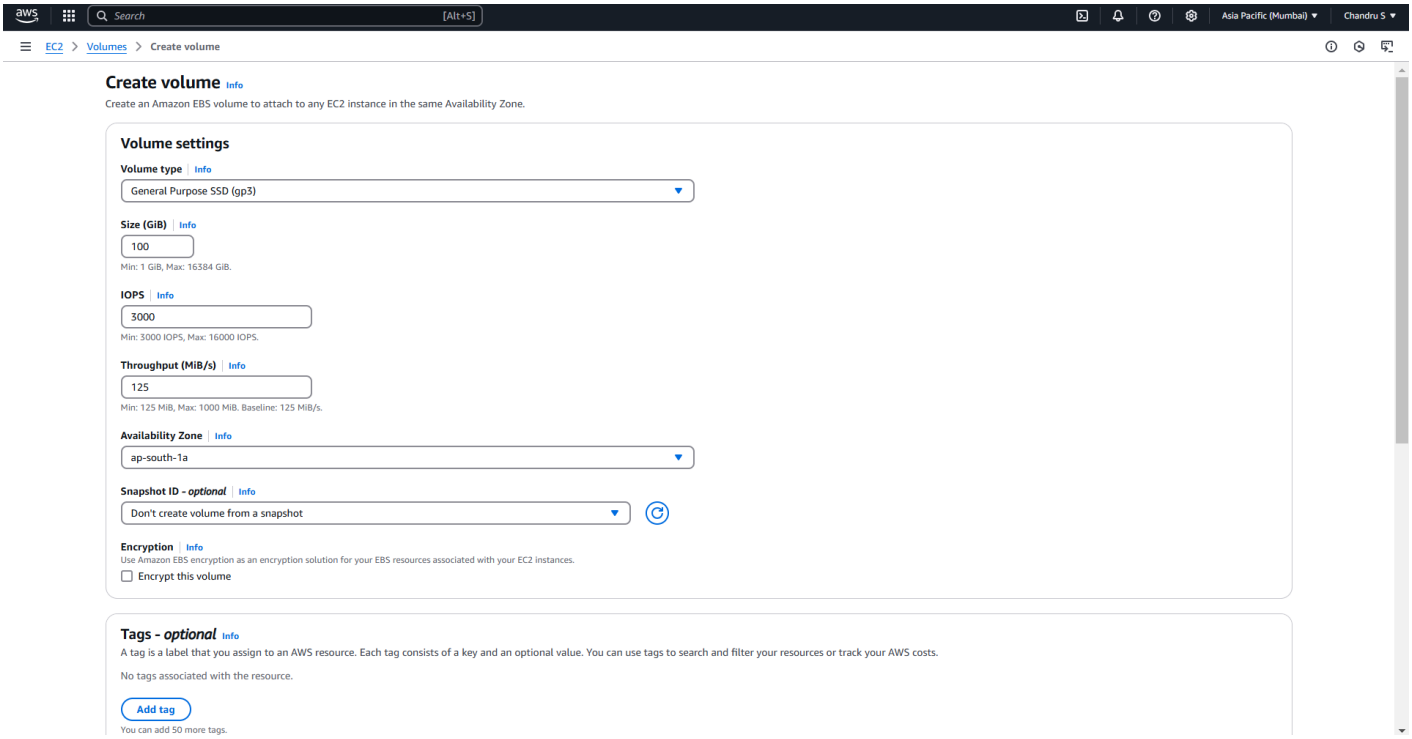
Step 2:

- Navigate to the EC2 Dashboard.
- Click Launch Instances and enter a name (e.g., *"My Monitoring Instance"*).
- Keep the default settings and click Launch Instance.



Step 3:

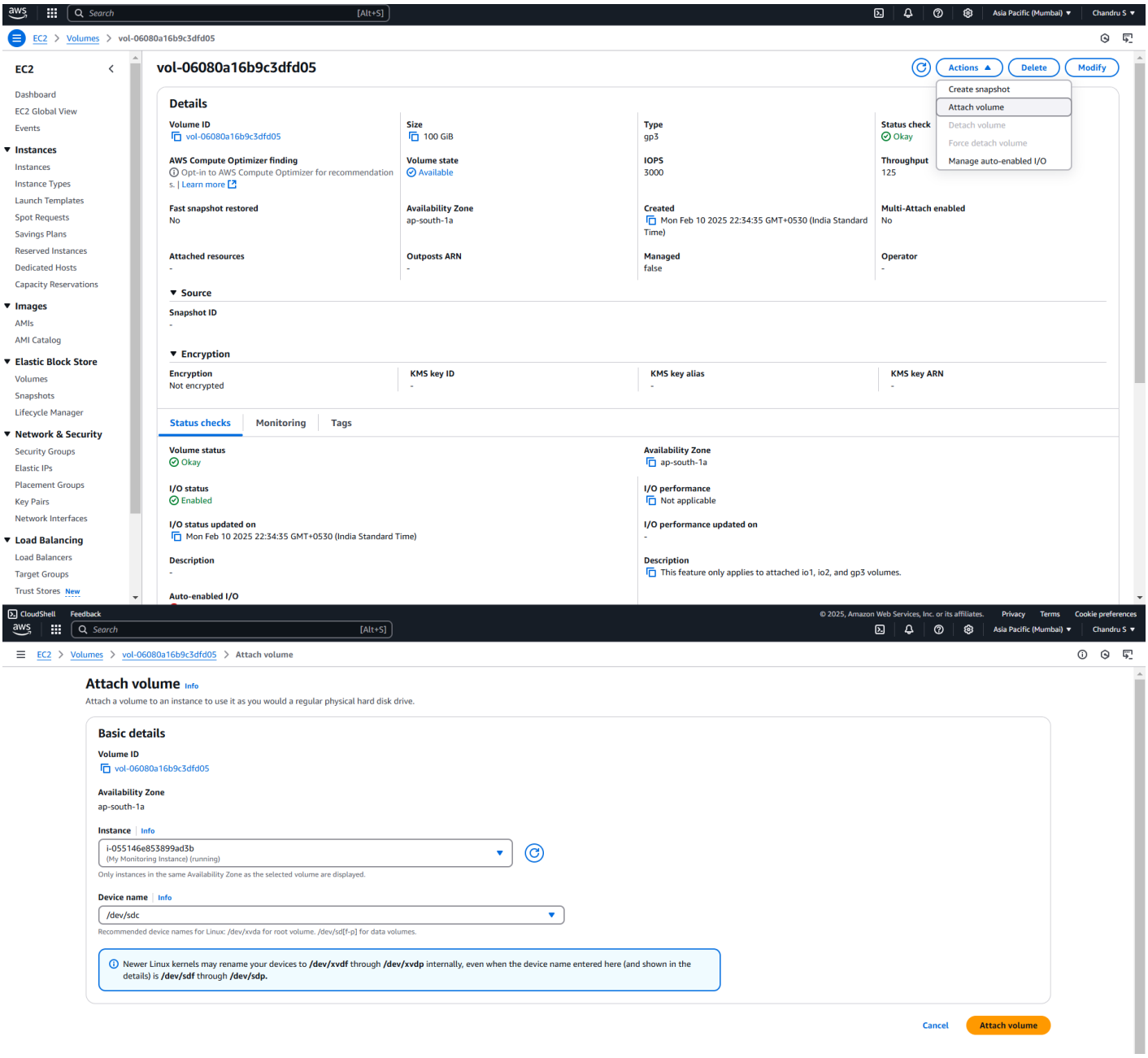
- In the EC2 Dashboard, go to the left menu and click Volumes under Elastic Block Store (EBS).
- Click Create Volume and configure the storage size and type as needed.
- Click Create to finalize the volume.



Step 4:

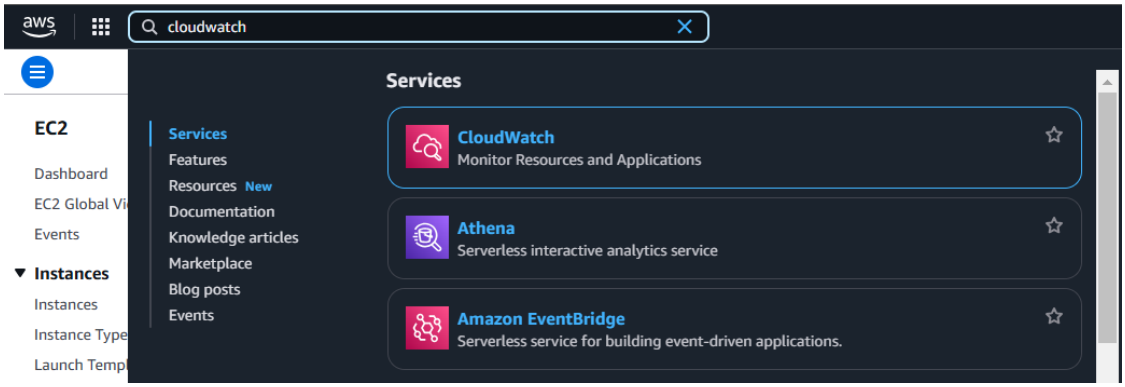
- Go to the Volumes list.
- Select the newly created EBS Volume.

- Click Actions > Attach Volume.
- Select the EC2 instance to which the volume should be attached.



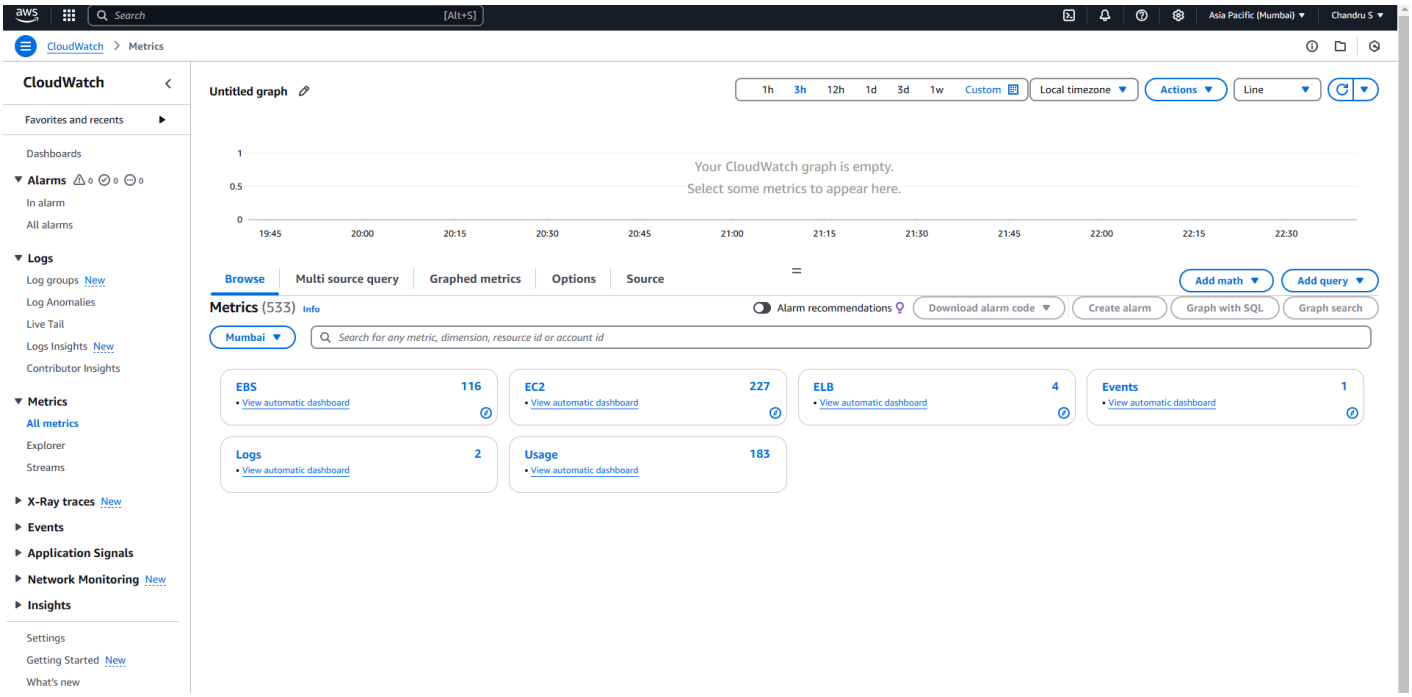
Step 5:

- Go to the AWS Console homepage.
- In the search bar at the top, type **CloudWatch** and press Enter.
- From the search results, click CloudWatch to open the monitoring dashboard.

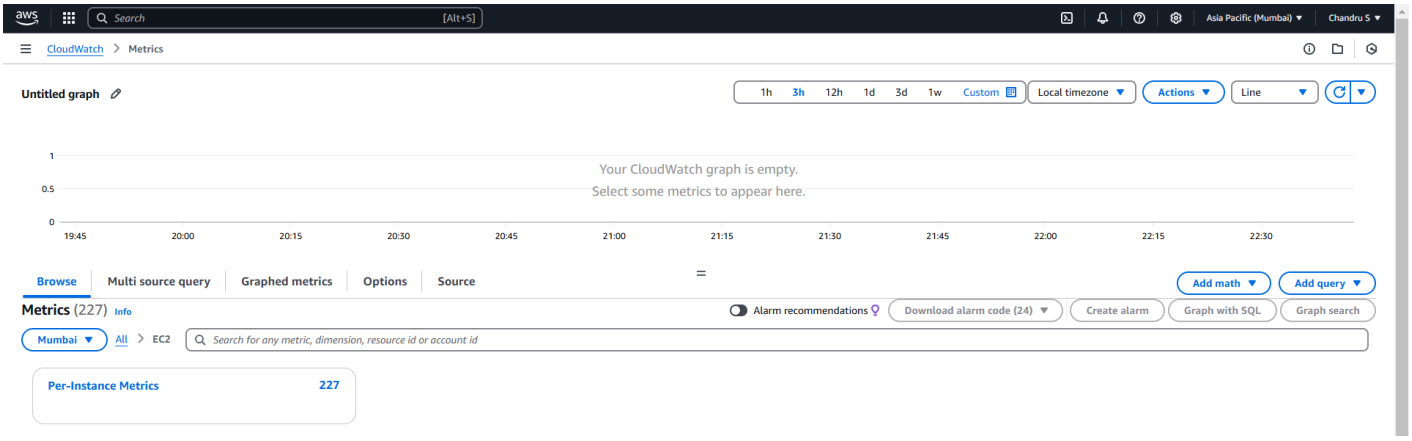


Step 6:

- In the CloudWatch Dashboard, look at the left-hand menu.
- Click **Metrics.**
- Under **Browse**, select **EC2**.

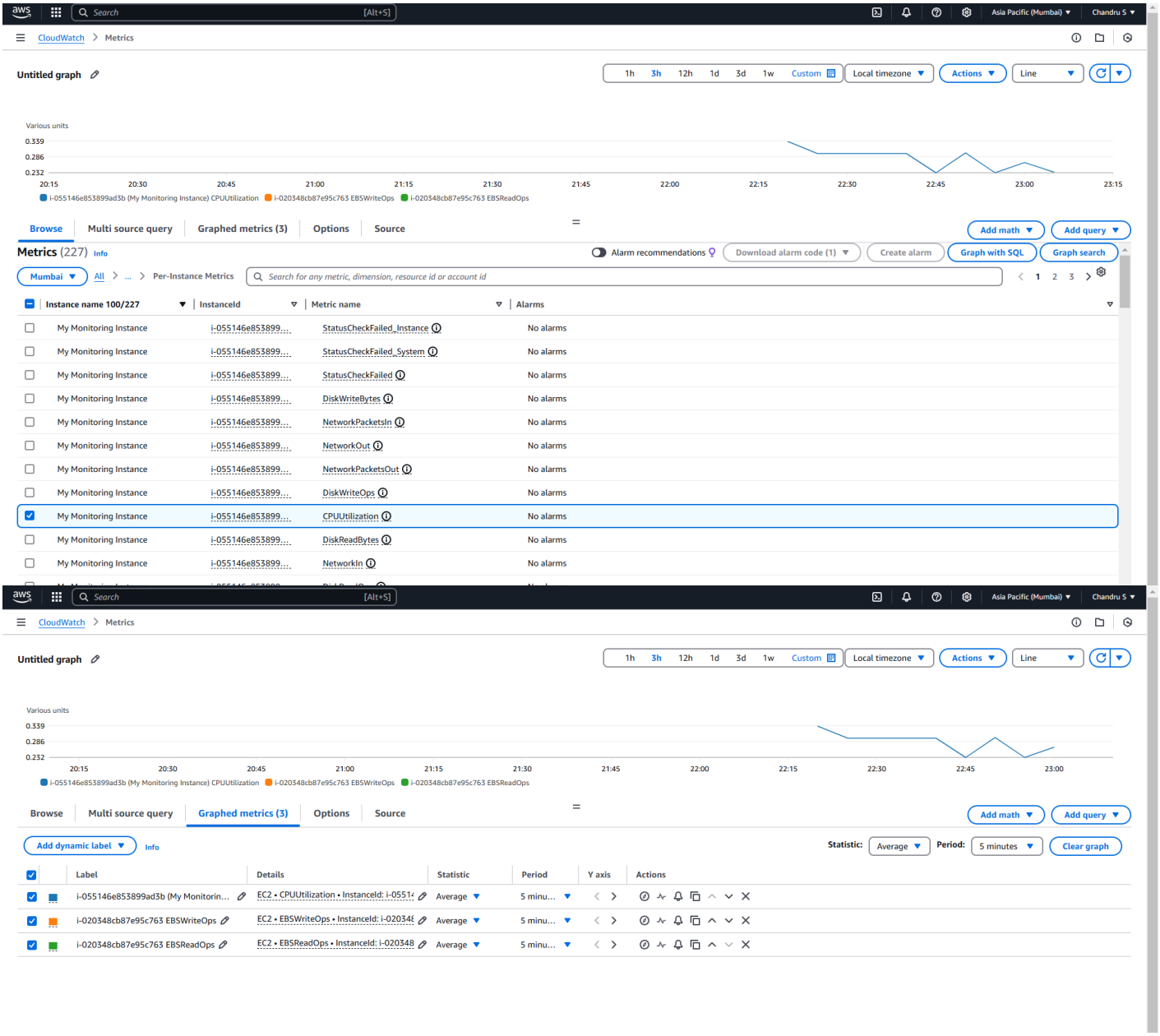


- Click on **Per-Instance Metrics** to view instance-specific data.



Step 7:

- A list of metrics for all EC2 instances will appear, including:
  - **CPUUtilization** (CPU usage)
  - **DiskReadOps** / **DiskWriteOps** (Disk I/O operations)
- Locate the specific EC2 instance by Instance ID.
- Click on the relevant metrics for your instance.
- To analyze the data, go to the Graphed metrics section for detailed visualization.



By following these steps, you can successfully enable AWS CloudWatch monitoring for your EC2 instance, track key performance metrics, and gain real-time insights into system health and resource usage.

Outcome

This Proof of Concept (PoC) successfully implemented AWS CloudWatch to monitor key performance metrics for an EC2 instance, focusing on **CPU utilization** and **Disk I/O** (DiskReadOps and DiskWriteOps).

Key Outcomes:

1. **CloudWatch Setup** – Successfully configured AWS CloudWatch to track EC2 instance metrics, including CPU utilization and Disk I/O.
2. **Disk I/O Monitoring** – Integrated an EBS volume with the EC2 instance to monitor DiskReadOps and DiskWriteOps, visualizing these metrics in CloudWatch.

3. **Cost Efficiency** – Ensured that the EBS volume usage remained within AWS Free Tier limits (30 GB), with all monitored metrics incurring no additional costs.

This PoC provided hands-on experience in configuring CloudWatch monitoring, optimizing cloud resource tracking, and ensuring cost-effective performance monitoring.