



Placement Empowerment Program

Cloud Computing and DevOps Centre

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

Name: Hema S

Department: ECE

Introduction:

In cloud computing, effective monitoring is crucial for ensuring the performance, reliability, and availability of cloud resources. **AWS CloudWatch** provides a comprehensive monitoring solution for AWS resources, enabling users to track various metrics in real-time. This Proof of Concept (PoC) focuses on leveraging **CloudWatch** to monitor the performance of an EC2 instance by enabling basic monitoring for key metrics such as **CPU utilization** and **disk I/O**. This PoC demonstrates how to enable, view, and analyze these metrics, giving insights into the health and performance of cloudbased virtual machines.

Overview:

This PoC will walk through the process of setting up **AWS CloudWatch** to monitor an EC2 instance. The main steps include:

1. Enabling basic cloud monitoring for an EC2 instance.
2. Viewing key metrics such as **CPU utilization** and **disk read/write operations**, to assess the performance of the instance.
3. Exploring how CloudWatch provides real-time insights into the instance's resource usage, allowing administrators to identify performance bottlenecks or issues before they affect the service.

By completing this PoC, users will understand how to integrate CloudWatch monitoring for EC2 instances, enabling effective performance monitoring of virtual machines in the cloud.

Objective:

The primary objective of this PoC is to enable **basic cloud monitoring** using **AWS CloudWatch** and view essential metrics for an EC2 instance. Specific goals include:

Enabling CloudWatch monitoring for an EC2 instance.

Viewing CPU usage and **disk I/O** metrics to analyze the instance's performance.

Understanding how CloudWatch helps in real-time monitoring by providing visibility into cloud resource health.

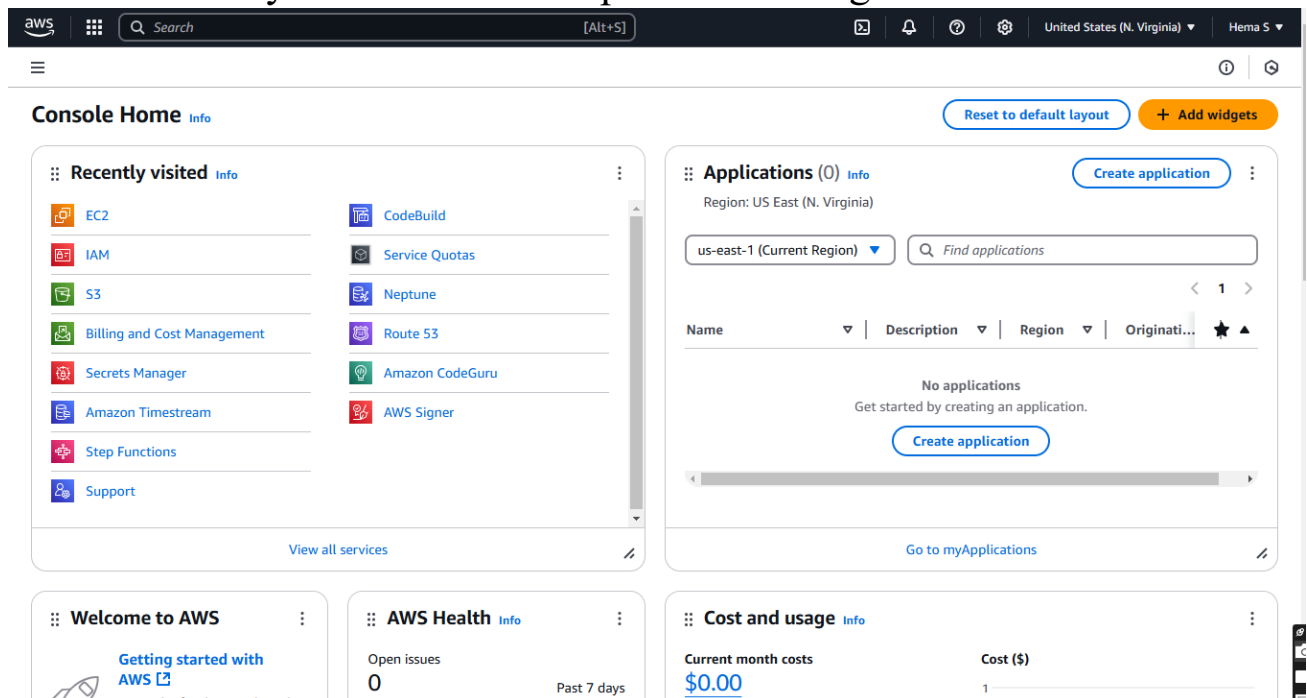
Importance of this PoC:

- 1. Performance Monitoring:** By tracking **CPU usage**, **disk I/O**, and **network traffic**, CloudWatch provides crucial insights into the resource utilization of an EC2 instance, which helps in identifying and troubleshooting performance issues.
- 2. Real-time Visibility:** Enabling CloudWatch monitoring ensures that administrators have access to real-time data about the instance's performance. This allows quick reactions to changes in resource consumption, preventing downtime or service degradation.
- 3. Resource Management:** Understanding the resource consumption of the EC2 instance (such as CPU usage and disk I/O) helps in optimizing the instance's capacity and managing resources efficiently, which can also lead to cost savings.
- 4. Proactive Issue Detection:** CloudWatch allows the user to monitor and understand patterns in the system's resource usage, helping detect performance anomalies or bottlenecks before they impact the system.

Step-by-Step Overview

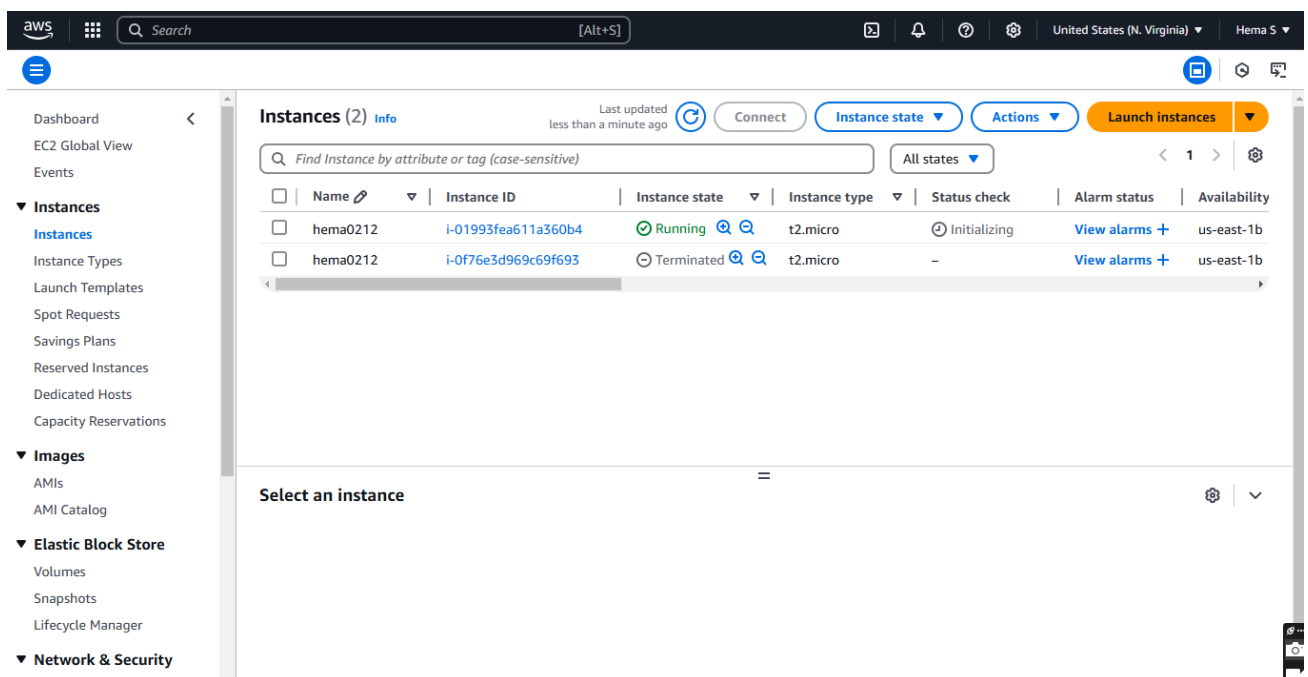
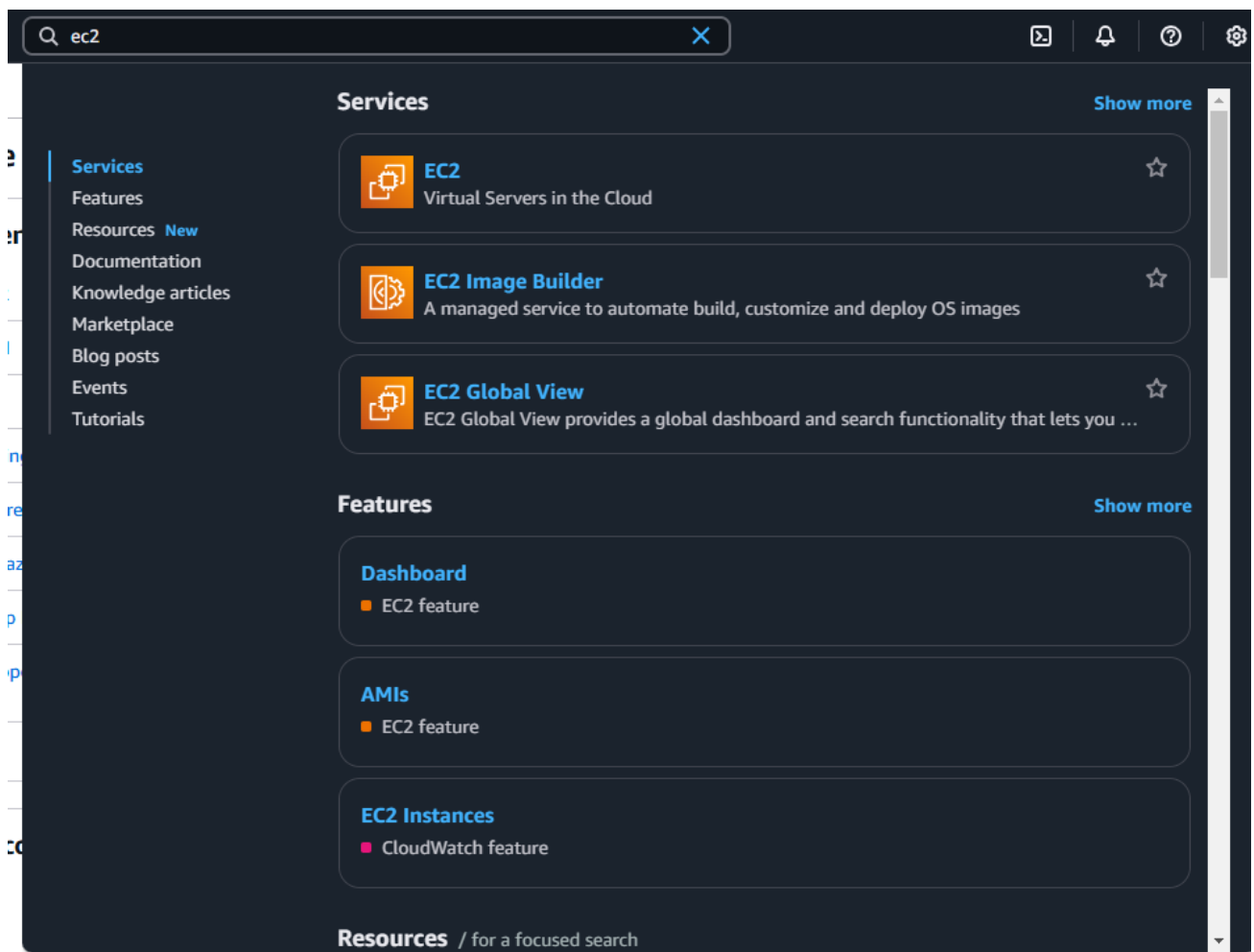
Step 1:

1. Go to [AWS Management Console](#).
2. Enter your username and password to log in.



Step 2:

On the EC2 Dashboard, click on **Launch Instances** and enter a name or your instance (e.g., "My Monitoring Instance"). Leave other settings as default and Click **Launch Instance**.



Step 3:

Go to the EC2 Dashboard in the AWS Console.

In the left menu, click Volumes under Elastic Block Store (EBS).

Click Create Volume.

aws

Search

[Alt+S]

United States (N. Virginia)

Hema S

EC2

Volumes

Create volume

1

Create volume

Info

Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.

Volume settings

Volume type

Info

General Purpose SSD (gp3)

Size (GiB)

Info

100

Min: 1 GiB, Max: 16384 GiB.

IOPS

Info

3000

Min: 3000 IOPS, Max: 16000 IOPS.

Throughput (MiB/s)

Info

125

Min: 125 MiB, Max: 1000 MiB. Baseline: 125 MiB/s.

Availability Zone

Info

us-east-1a

Snapshot ID - optional

Info

Don't create volume from a snapshot

STEP 4 :

Once created, go to your **Volumes** list, select the newly created volume, and click **Actions > Attach Volume**.

The screenshot shows the AWS Management Console interface for a specific EBS volume. The breadcrumb navigation at the top indicates the path: EC2 > Volumes > vol-04465eea0d9175ef6. The left-hand navigation pane lists various AWS services, with 'Elastic Block Store' expanded to show 'Volumes'. The main content area displays the details for volume 'vol-04465eea0d9175ef6'. Key information includes: Volume ID (vol-04465eea0d9175ef6), Size (100 GiB), Type (gp3), Status (Available), and IOPS (3000). It also shows the Availability Zone (us-east-1a), Created time (Wed Feb 05 2025 18:20:38 GMT+0530), and that it is not Multi-Attach enabled. A 'Status check' section shows 'Okay'. Below the main details, there are sections for 'Source' (Snapshot ID) and 'Encryption' (Not encrypted). At the bottom, there are tabs for 'Status checks', 'Monitoring', and 'Tags'.

vol-04465eea0d9175ef6		Actions		Delete	Modify
Volume ID vol-04465eea0d9175ef6	Size 100 GiB	Type gp3	Status check Okay		
AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more	Volume state Available	IOPS 3000	Throughput 126		
Fast snapshot restored No	Availability Zone us-east-1a	Created Wed Feb 05 2025 18:20:38 GMT+0530 (India Standard Time)	Multi-Attach enabled No		
Attached resources -	Outposts ARN -	Managed false	Operator -		
Source					
Snapshot ID -					
Encryption					
Encryption Not encrypted	KMS key ID -	KMS key alias -	KMS key ARN -		
Status checks Monitoring Tags					

The screenshot shows the 'Attach volume' wizard in the AWS Management Console. The breadcrumb navigation is EC2 > Volumes > vol-04a92ca5bf760ed4e > Attach volume. The main content area is titled 'Attach volume' with an 'Info' link. Below the title, it says 'Attach a volume to an instance to use it as you would a regular physical hard disk drive.' The 'Basic details' section contains the following fields: 'Volume ID' (vol-04a92ca5bf760ed4e), 'Availability Zone' (us-east-1b), 'Instance' (i-01993fea611a360b4, hema0212 (running)), and 'Device name' (/dev/sdb). A note at the bottom states: 'Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.' At the bottom right, there are 'Cancel' and 'Attach volume' buttons.

Attach volume [Info](#)

Attach a volume to an instance to use it as you would a regular physical hard disk drive.

Basic details

Volume ID
vol-04a92ca5bf760ed4e

Availability Zone
us-east-1b

Instance [Info](#)
i-01993fea611a360b4 (hema0212) (running)

Only instances in the same Availability Zone as the selected volume are displayed.

Device name [Info](#)
/dev/sdb

Recommended device names for Linux: /dev/xvda for root volume, /dev/sd[f-p] for data volumes.

Info Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.

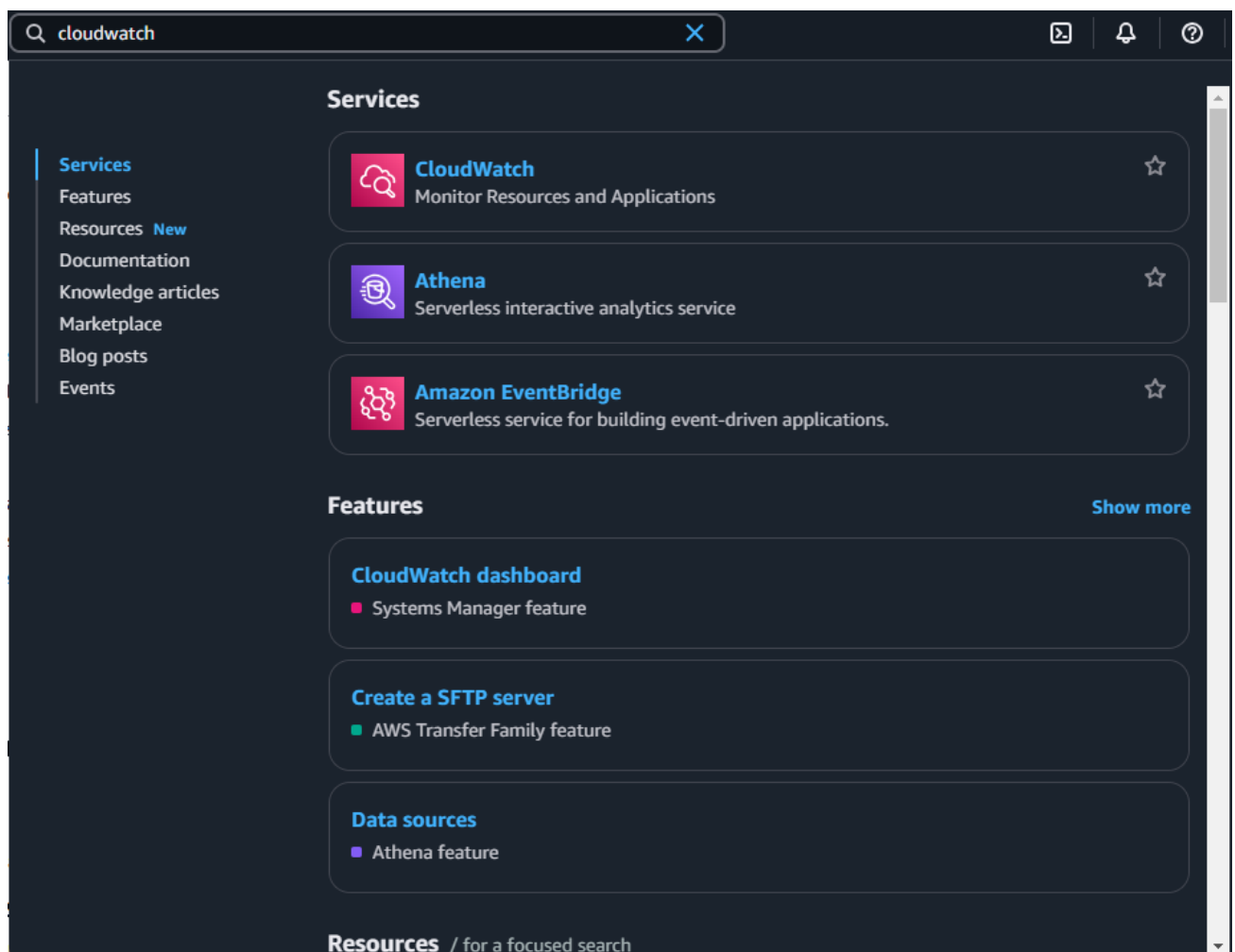
[Cancel](#) [Attach volume](#)

On the AWS Console homepage, look for the search bar at the top.

STEP 5 :

Type **CloudWatch** in the search bar and press **Enter**.

From the search results, click on **CloudWatch**.



In the CloudWatch dashboard, look at the left-hand menu.

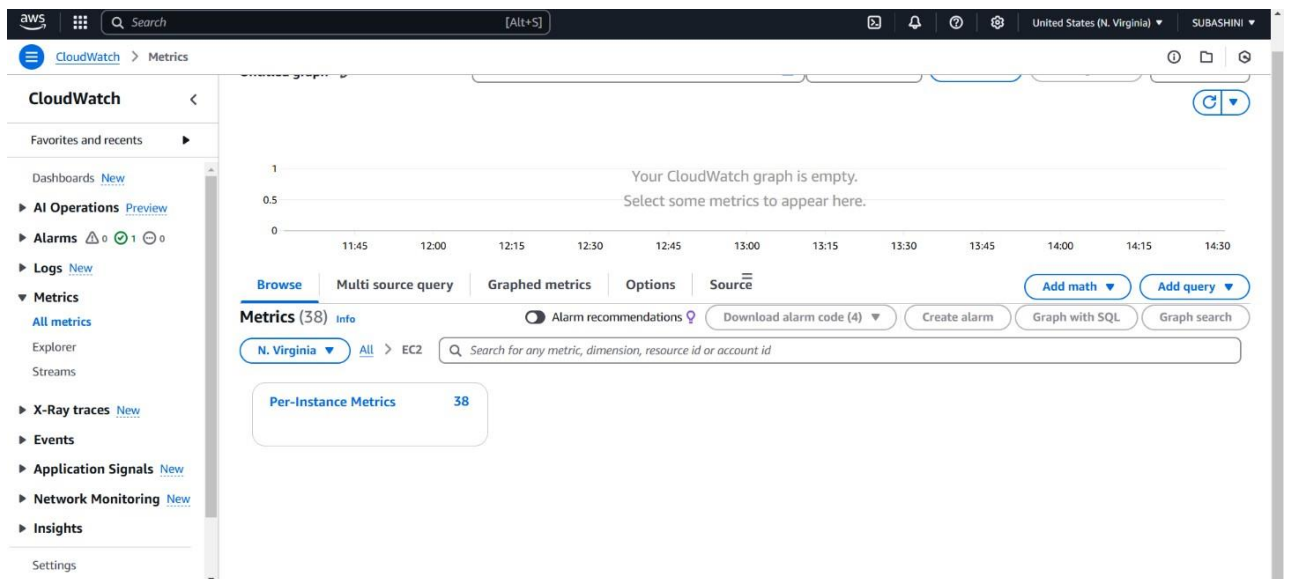
STEP 6 :

Click on **Metrics**.

Under **Browse**, click on **EC2**.

The image displays two screenshots of the AWS CloudWatch Metrics console interface. The top screenshot shows the 'Metrics' page with a search bar and a list of metric categories: EBS (80), EC2 (162), Events (1), Logs (2), AWS/SecretsManager (1), and Usage (112). The bottom screenshot shows the 'EC2' category selected, displaying 'Per-Instance Metrics' with a count of 162. Both screenshots show a graph area with the text 'Your CloudWatch graph is empty. Select some metrics to appear here.' and a time range selector set to '3h'.

Then click on the **Per-Instance Metrics**.



CPUUtilization (CPU usage)

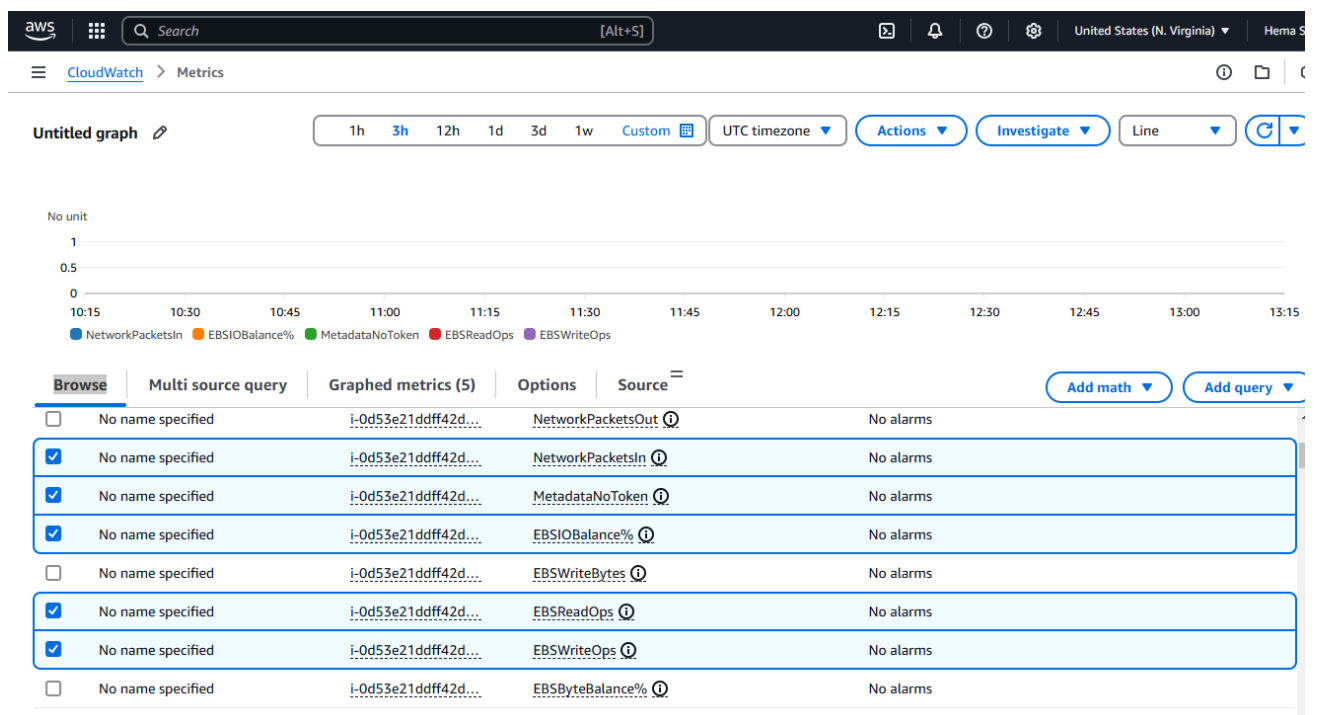
STEP 7 :

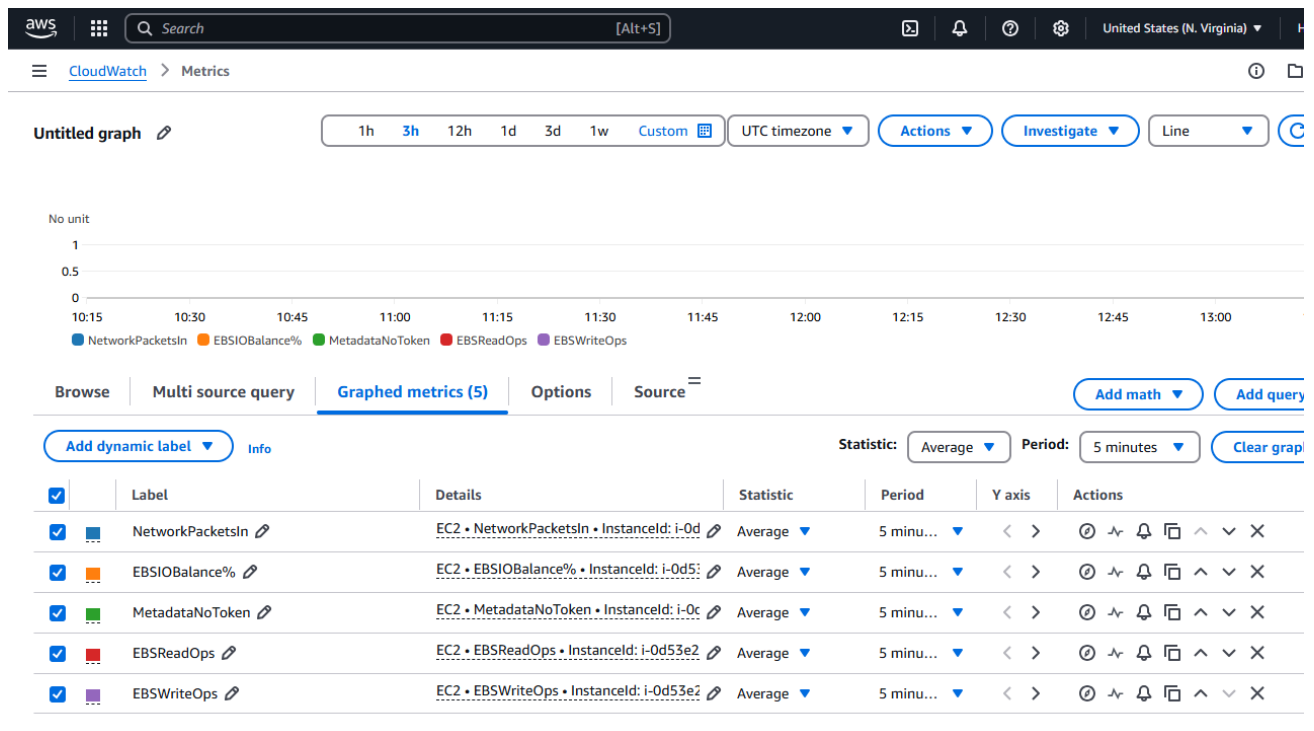
DiskReadOps / DiskWriteOps (Disk I/O)

Identify the specific EC2 instance you want to monitor (it will be listed by its instance ID).

Click on the metrics associated with your instance

To view detail click Graphed metrics





Outcome

This Proof of Concept (PoC) aimed to establish a **cloud-based monitoring service** using **AWS CloudWatch** to track key performance metrics for an EC2 instance, specifically focusing on **CPU utilization** and **Disk I/O** (DiskReadOps and DiskWriteOps).

Here's the outcome of the PoC:

1. **CloudWatch Setup:** Successfully configured AWS CloudWatch to monitor EC2 instance metrics like **CPU utilization** and **Disk I/O**

(DiskReadOps, DiskWriteOps).

2. **Disk I/O Monitoring:** Added an **EBS volume** to the EC2 instance to track **DiskReadOps** and **DiskWriteOps** metrics, which were visualized in CloudWatch.
3. **Cost Efficiency:** The EBS volume was within the **AWS Free Tier** limits (30 GB), and all metrics stayed within **Free Tier** usage, incurring no additional cost.