

Steps for AWS CloudWatch

1. Log in to the AWS Management Console

- Go to the [AWS Management Console](#).
- Log in with your credentials.

2. Launch an EC2 Instance (if you don't have one)

1. Go to the **EC2** service.
2. Click **Launch Instance**.
3. Choose an Amazon Machine Image (AMI) (e.g., Amazon Linux 2).
4. Select an instance type (e.g., t2.micro).
5. Configure instance details (leave defaults if unsure).
6. Add storage (default is usually sufficient).
7. Add tags (optional).
8. Configure security groups to allow SSH (port 22) and HTTP (port 80) traffic.
9. Launch the instance and download the key pair.

3. Enable CloudWatch Monitoring for the EC2 Instance

1. Go to the **EC2 Dashboard**.
2. Select your instance.
3. Click **Actions > Monitor and troubleshoot > Manage CloudWatch monitoring**.
4. Enable **Detailed Monitoring** (optional but recommended for more granular metrics).
5. Click **Save**.

4. Create a CloudWatch Dashboard

1. Go to the **CloudWatch** service.
2. In the left sidebar, click **Dashboards**.
3. Click **Create dashboard**.
4. Enter a dashboard name (e.g., Codetech-Monitoring-Dashboard).
5. Click **Create dashboard**.
6. Add widgets to the dashboard:

- Click **Add widget**.
 - Choose a widget type (e.g., Line, Stacked Area, Number).
 - Select the metric you want to monitor (e.g., CPUUtilization, NetworkIn, NetworkOut).
 - Configure the widget and click **Create widget**.
7. Repeat to add more widgets for other metrics (e.g., Memory Usage, Disk I/O).

5. Create CloudWatch Alarms

1. In the CloudWatch console, go to **Alarms** in the left sidebar.
2. Click **Create alarm**.
3. Click **Select metric**.
4. Choose a metric (e.g., EC2 > Per-Instance Metrics > CPUUtilization).
5. Set the conditions for the alarm:
 - Threshold type: Static.
 - Define the threshold (e.g., CPUUtilization > 80%).
 - Set the alarm to trigger when the threshold is breached for 1 consecutive period.
6. Configure actions:
 - Send a notification to an SNS topic (create an SNS topic if you don't have one).
 - Add an email address to the SNS topic to receive notifications.
7. Enter an alarm name (e.g., High-CPU-Utilization-Alarm).
8. Click **Create alarm**.

6. Verify Alarms and Dashboard

1. Go to the **CloudWatch Dashboard** to view the metrics in real-time.
2. Simulate high CPU usage on your EC2 instance (e.g., run a stress test).
3. Check if the alarm triggers and you receive an email notification.

Deliverables

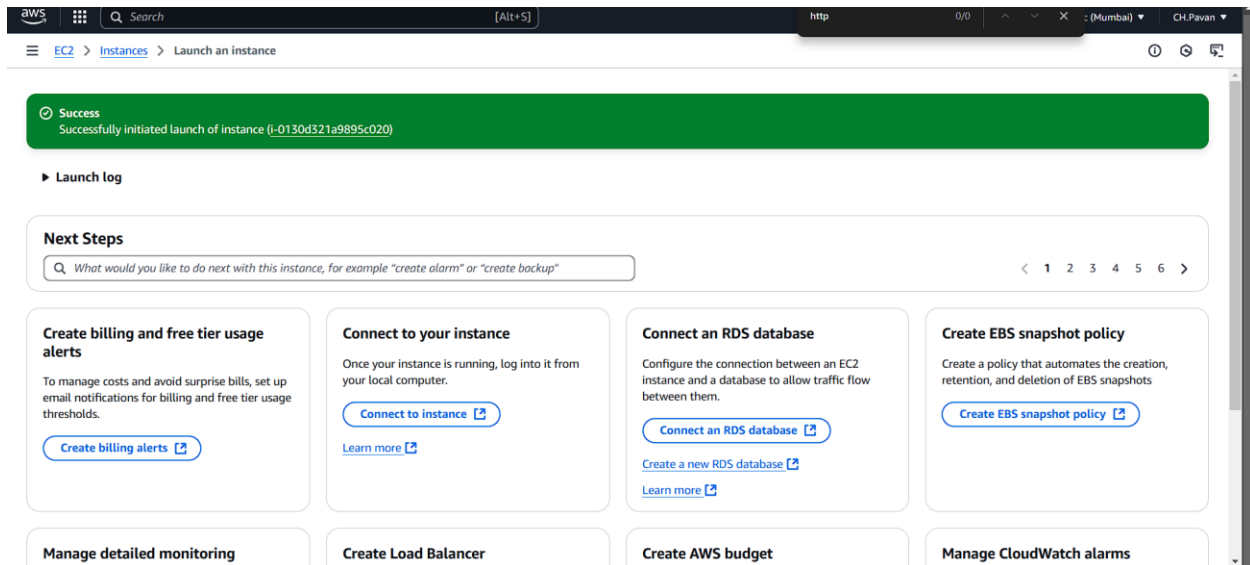
1. A CloudWatch dashboard showcasing metrics (e.g., CPU, Memory, Network).

Intern ID: CT06DG1190

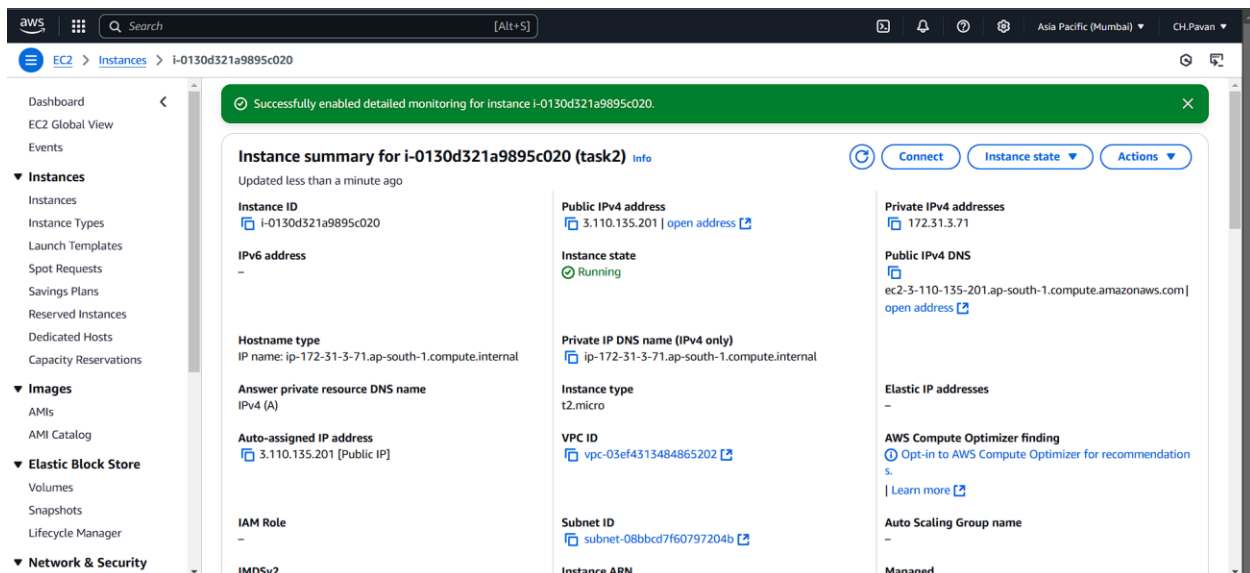
2. Configured CloudWatch alarms (e.g., for high CPU usage).
3. Screenshots or documentation of the dashboard and alarms.

Screen shots

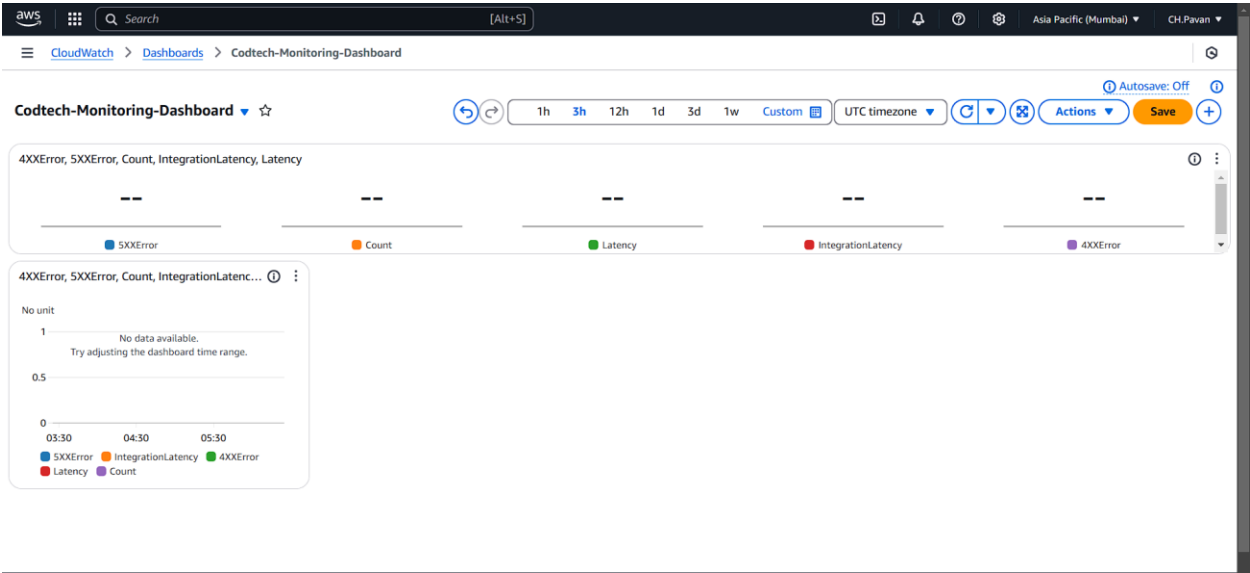
Launch instances



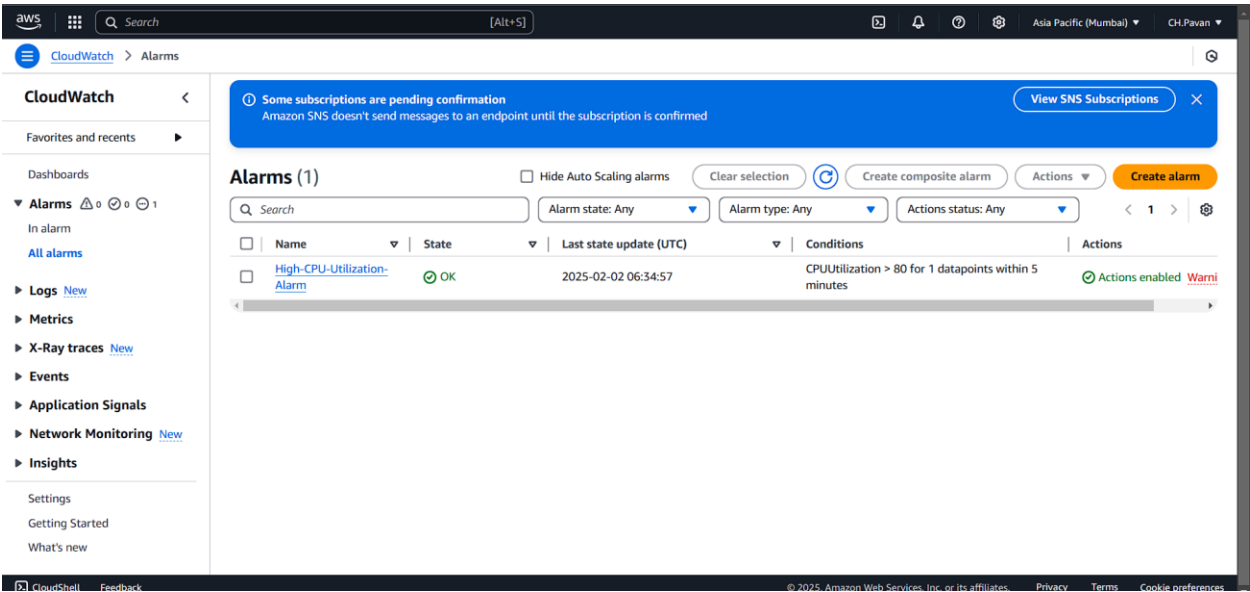
Enabled Detailed Monitoring

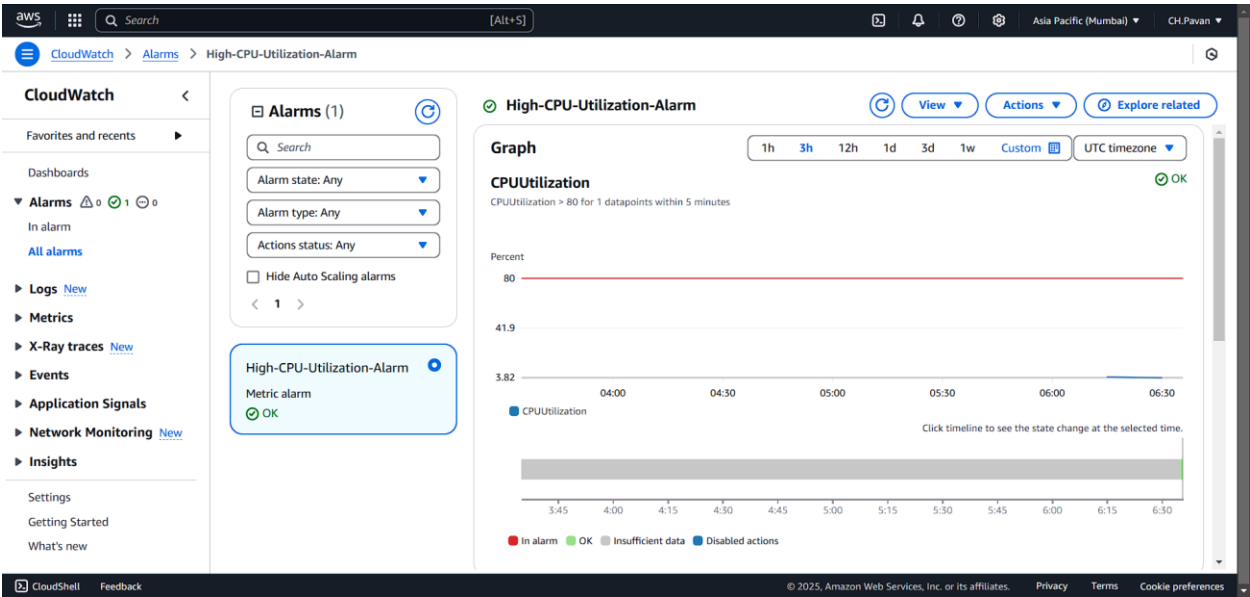


Created Cloud watch Dashboard

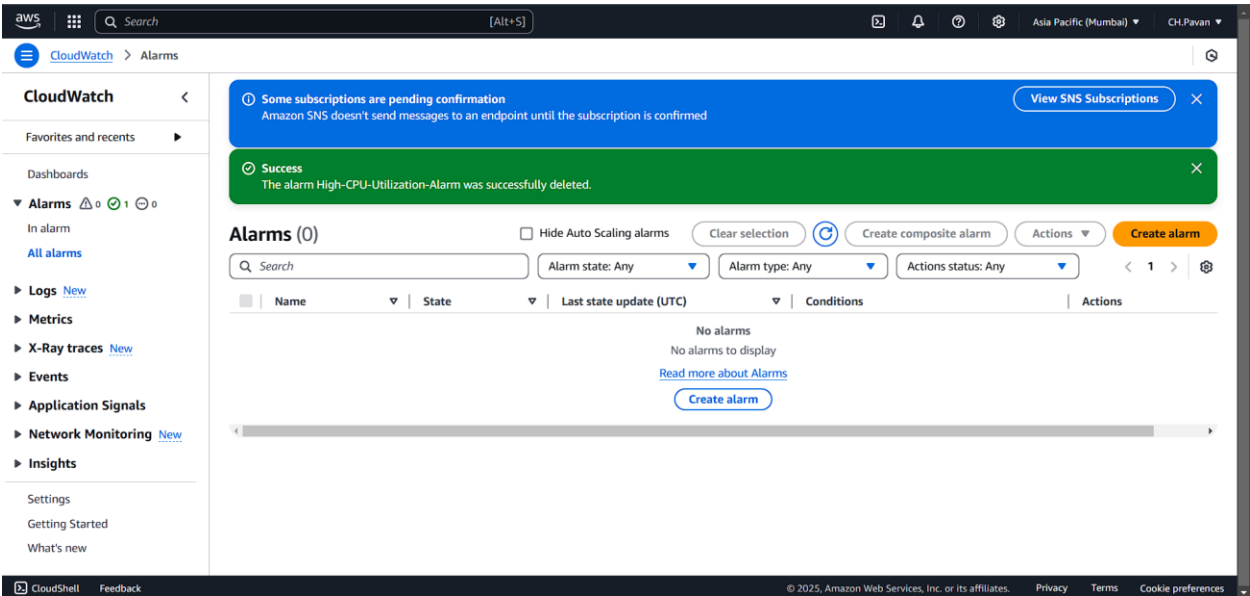


Created an alarm





Deleted alarm



Deleted instance

Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

Successfully initiated termination (deletion) of i-0130d321a9895c020

Instance summary for i-0130d321a9895c020 (task2)

Updated less than a minute ago

Instance ID

i-0130d321a9895c020

IPv6 address

-

Hostname type

-

Answer private resource DNS name

-

Auto-assigned IP address

-

IAM Role

-

IMDSv2

Required

Public IPv4 address

-

Instance state

Terminated

Instance type

t2.micro

VPC ID

-

Subnet ID

-

Instance ARN

arn:aws:ec2:ap-south-1:952216971402:instance/i-0130d321a9895c020

Private IPv4 addresses

-

Public IPv4 DNS

-

Elastic IP addresses

-

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendation s.

Learn more

Auto Scaling Group name

-

Managed

false

Connect

Instance state

Actions

© 2025, Amazon Web Services, Inc. or its affiliates

Privacy

Terms

Cookie preferences