# RDS Cloud Monitoring & Alerts using CloudWatch + SNS

## Project Overview:

This project demonstrates how to monitor an Amazon RDS instance using Amazon CloudWatch and set up alerting through Amazon SNS. When CPU utilization crosses a defined threshold, an alert is triggered and a notification email is sent.

### Key Features:

* AWS RDS (MySQL) setup using Free Tier
* CloudWatch metrics monitoring
* CPUUtilization alarm configuration
* SNS topic creation with email alert
* Simulated CPU spike using BENCHMARK () query from EC2

Note: Already launch an EC2 instance of type ubuntu (I have used that one) so that we can perform RDS-EC2 connection while creating database.

Instead we need to edit inbound rules of security groups and add port 3306 manually from EC2 to RDS.

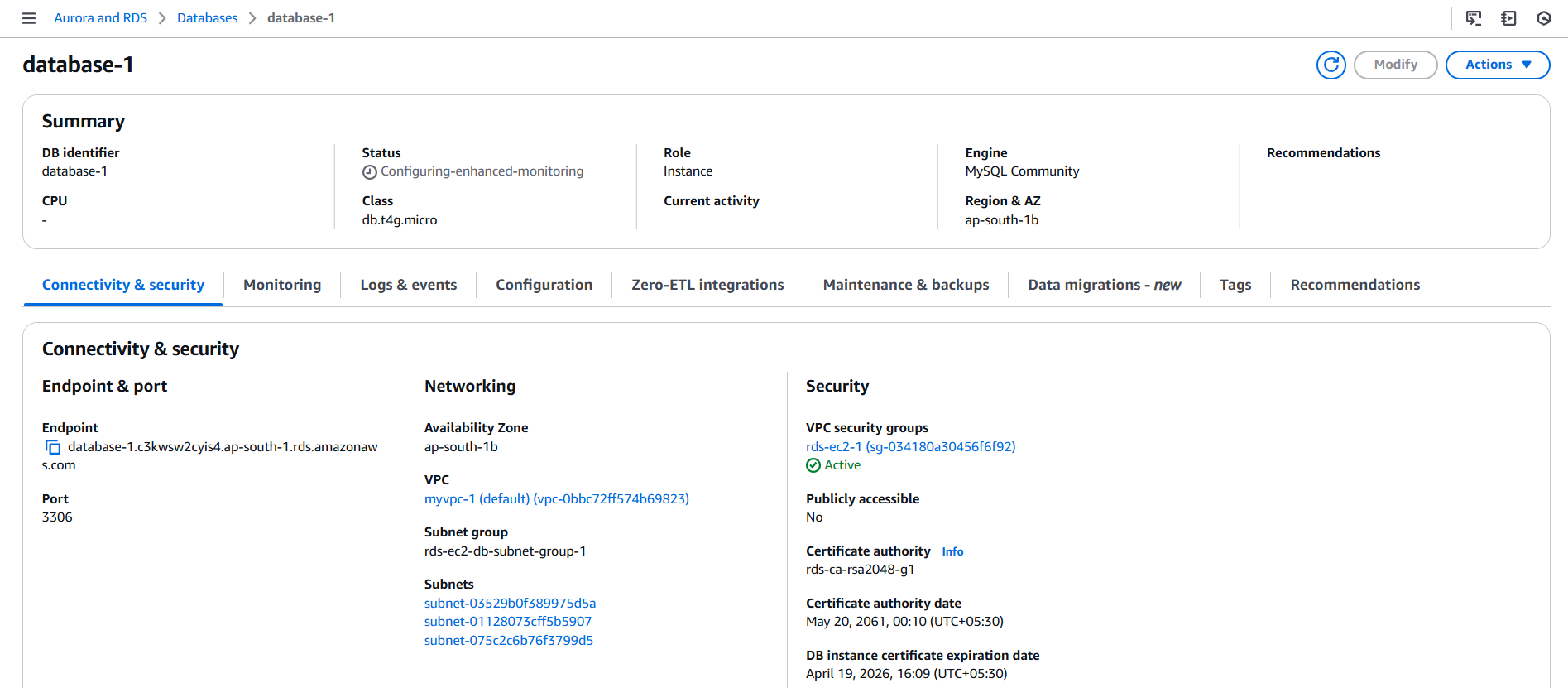
### Step 1: RDS

* Go to AWS Aurora & RDS dashboard.
* Open databases interface from sidebar and start creating database.

#### Configurations:

1. Database creation method: Easy Create.
2. Configuration: MySQL.
3. DB instance size: Free tier.
4. Master username: admin.
5. Credentials manager: Self-managed and enter password.
6. Set up EC2 connection: Choose already created instance.
7. Create Database.

* Wait for some time as database creation and availability takes some time.



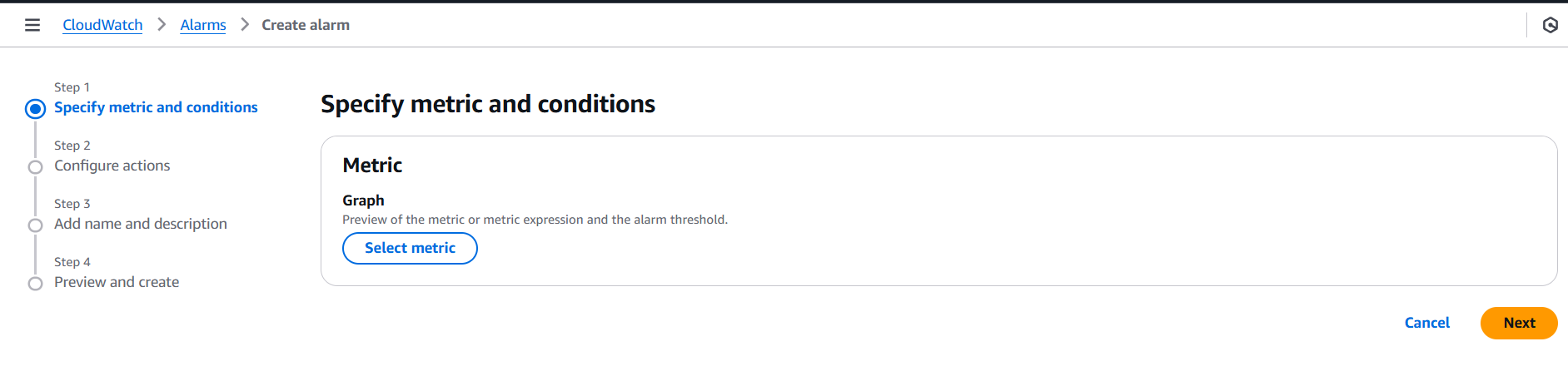
* Once the endpoint appears, copy it and go to EC2 steps.

### Step 2: EC2

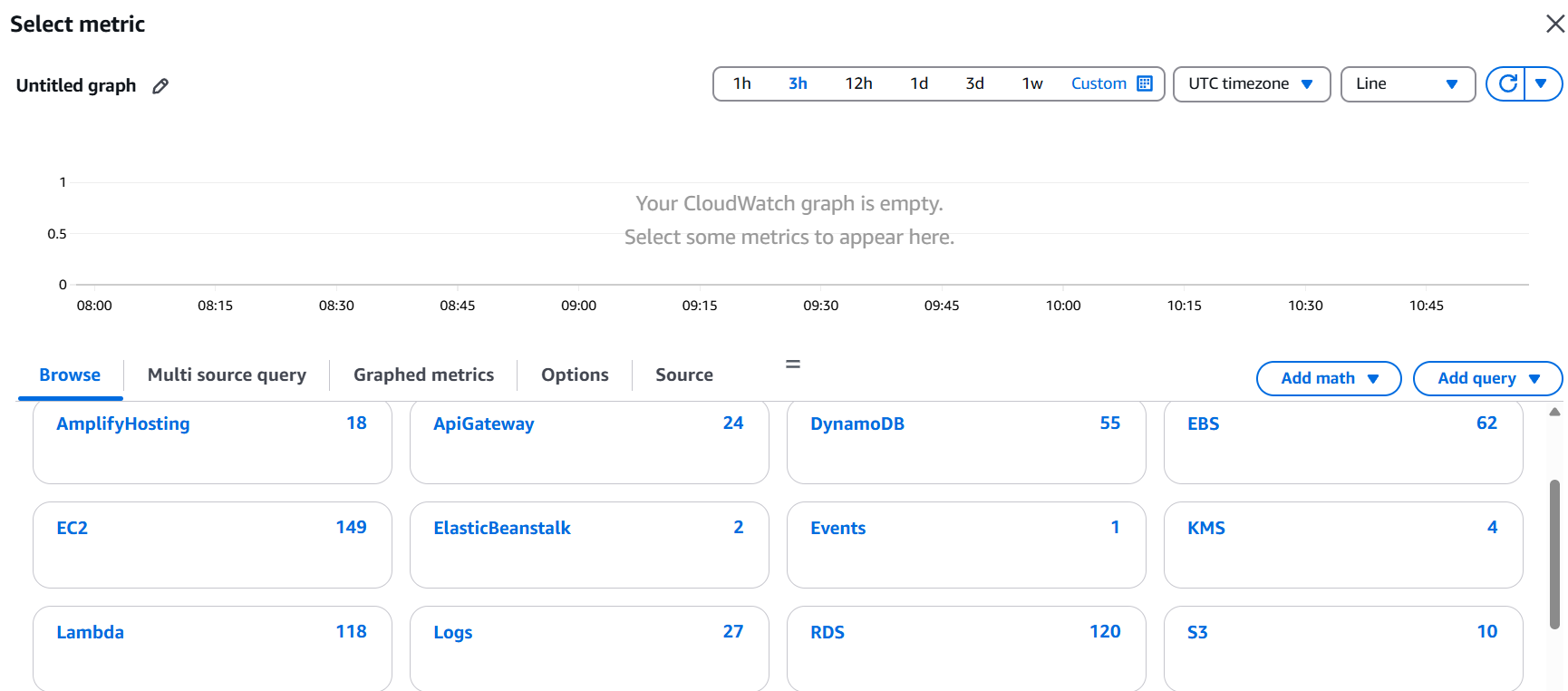
* Connect to the EC2 instance from powershell or your machine’s terminal via ssh (ssh -i your-key.pem ubuntu@your-ec2-public-ip).
* Update package manager and install maraidb105-server to have mysql features.
* Run below commands to do so:
  + sudo apt-get update
  + sudo apt-get install mariadb-server -y
  + sudo systemctl start mariadb

### Step 3: CloudWatch

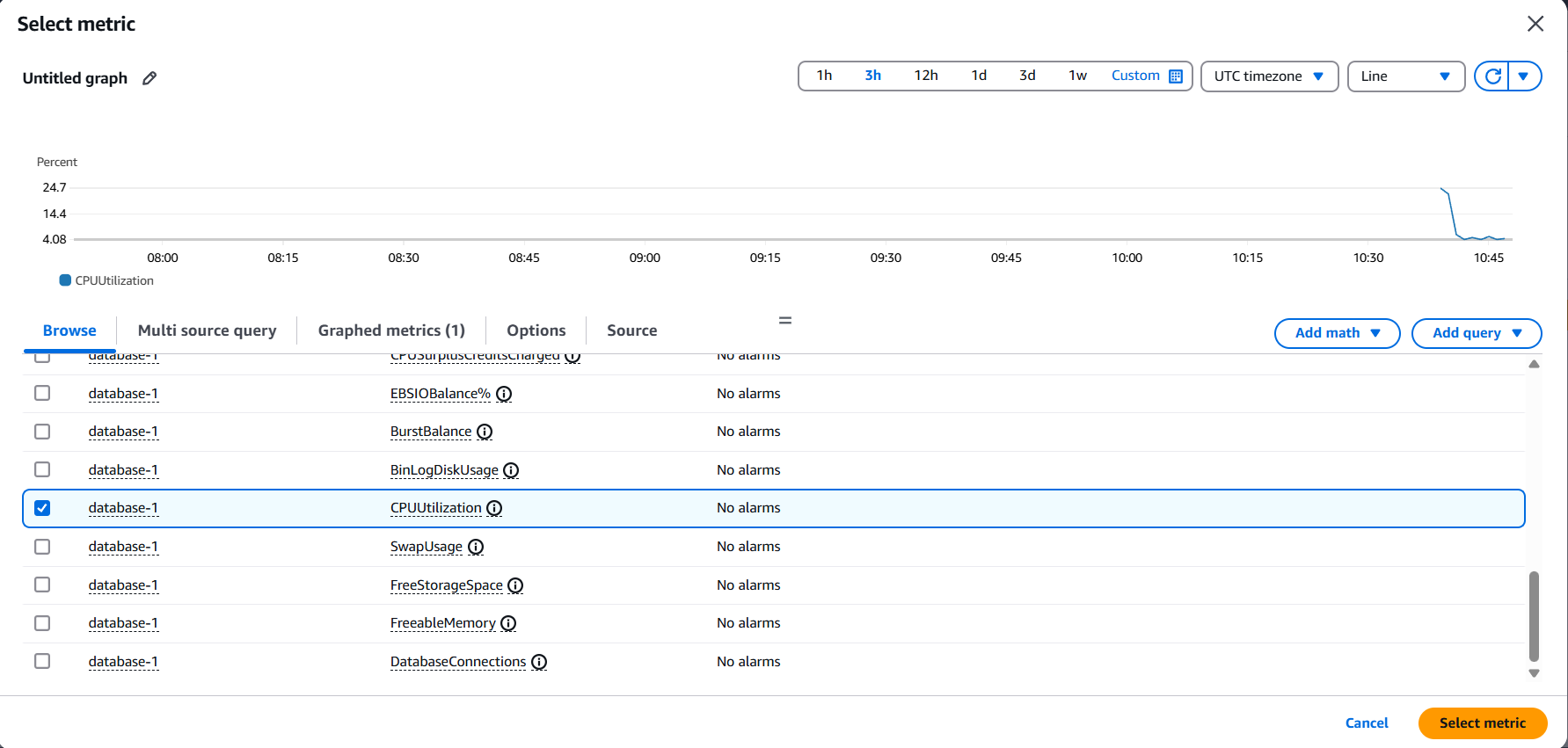
* Go to CloudWatch dashboard, choose “All alarms” from side pane and click on create alarm.



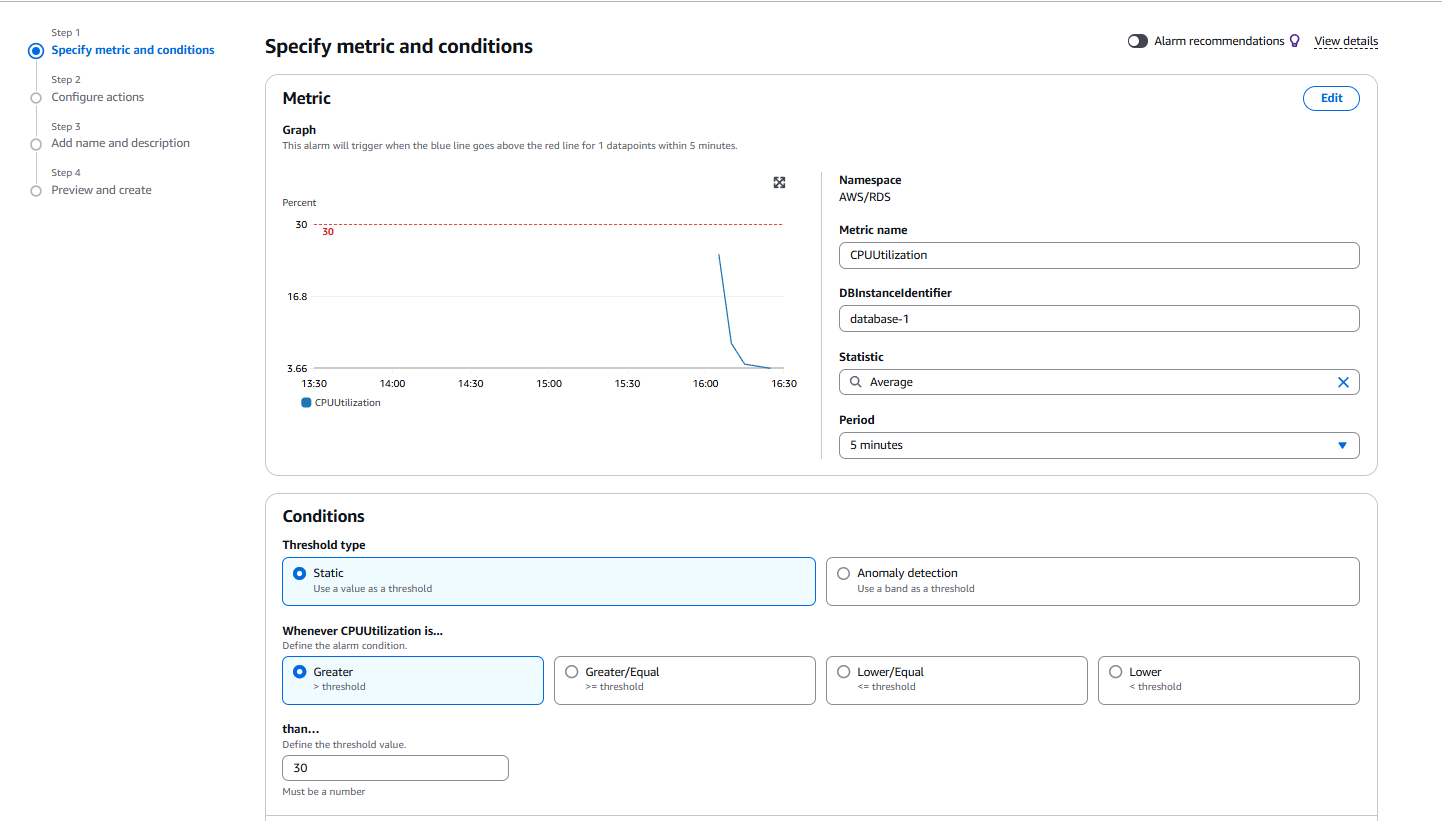
* Select metric: RDS



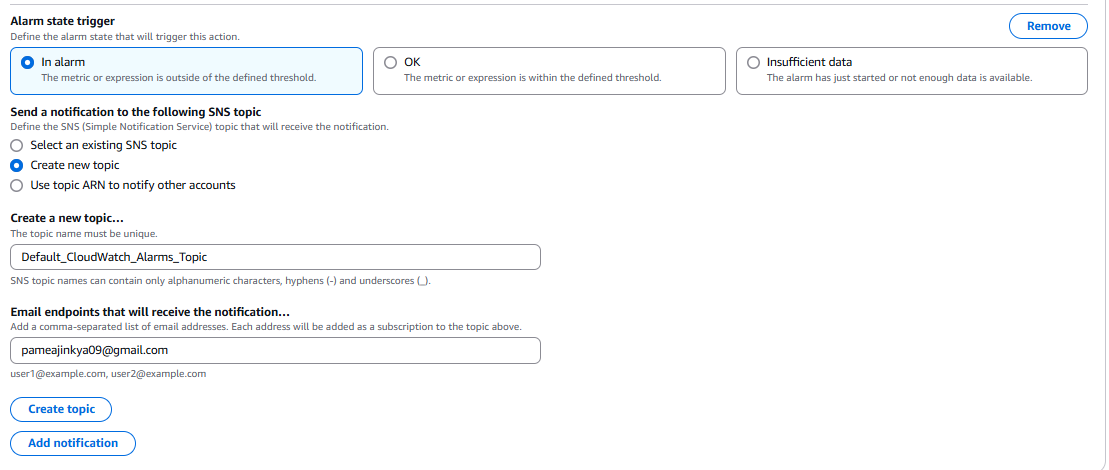
* Select DBInstanceIdentifier and choose CPUUtilization.

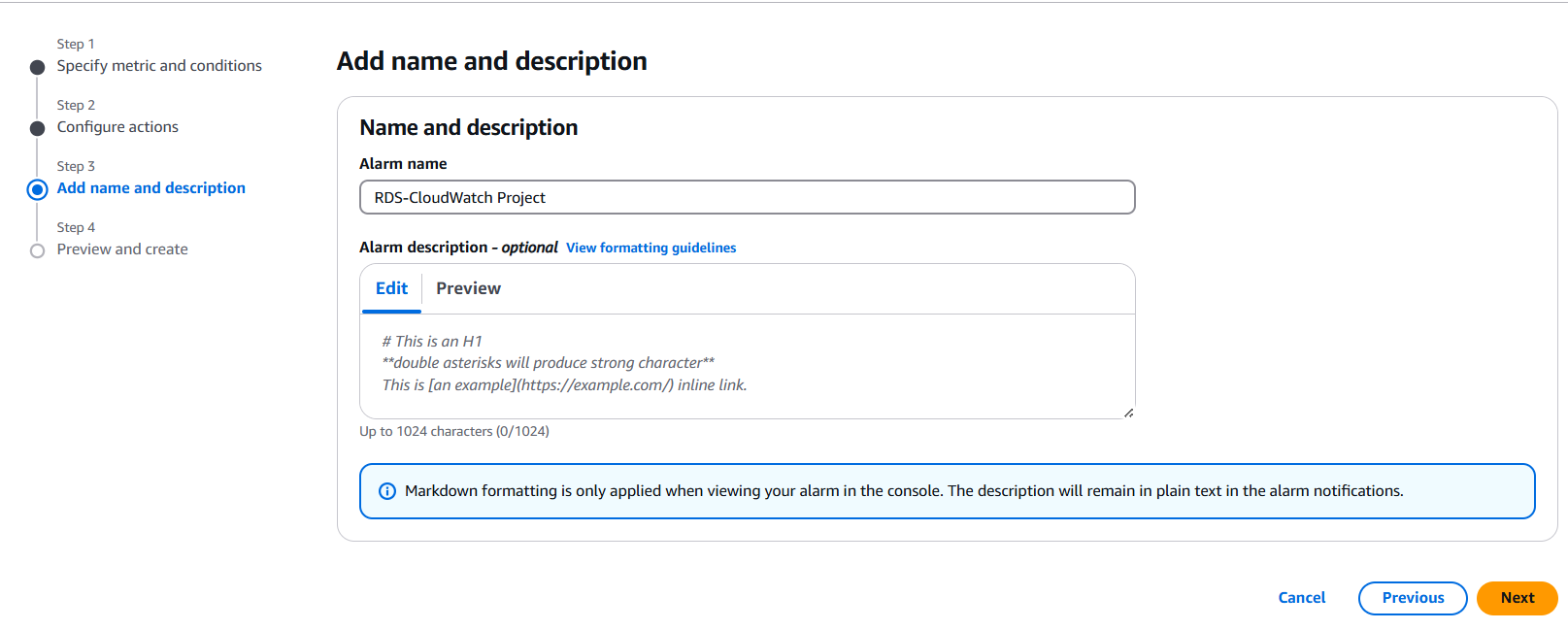


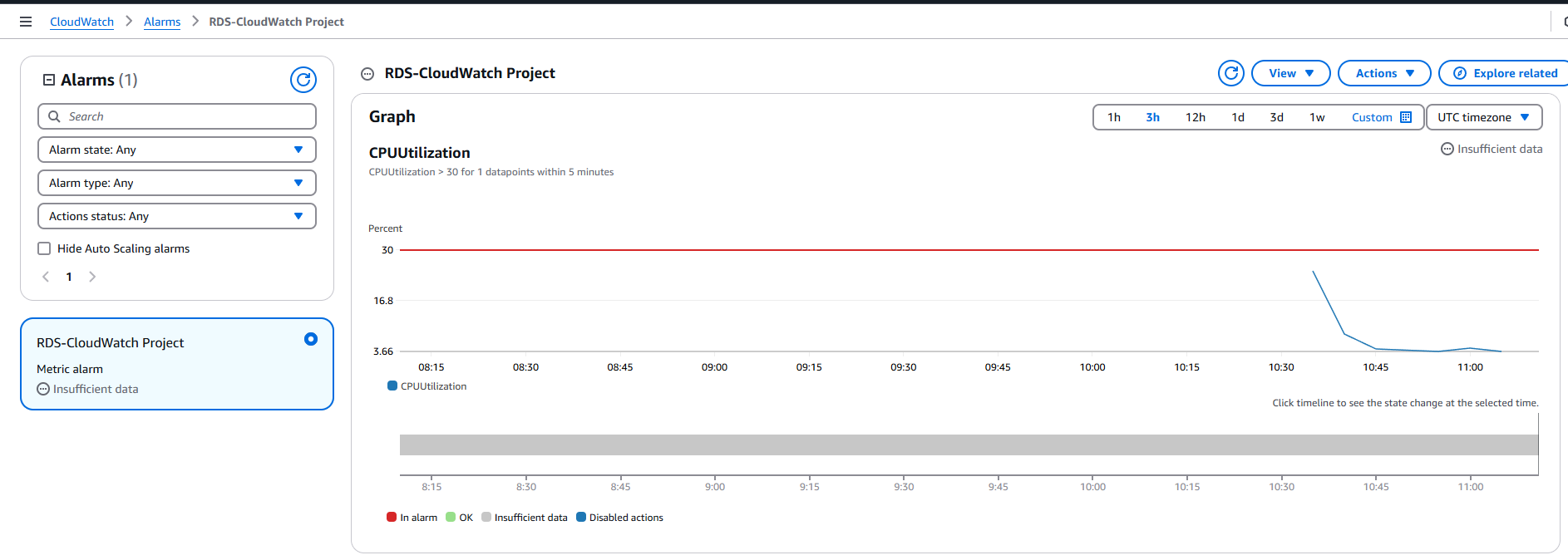
* Select details as below: (instead of 30 keep 10 or 15 for faster response)



* In configure actions:
  + Click on create topic.
  + Enter your mail.
  + Create the topic with endpoint as your mail.
  + Choose that topic as notification.
  + Click next to give name to the alarm.
  + Enter name of your own and go to last step.
  + Preview and create alarm.



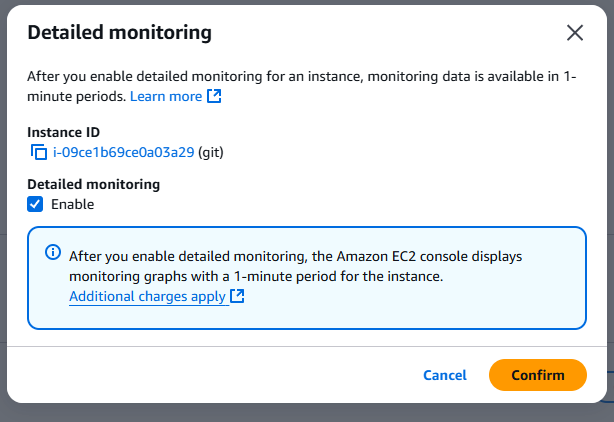




* As of now there is no load on CPU so it is alright.
* Let’s increase stress on it form instance.
* Go to SNS topic and confirm mail subscription so that a mail will come as soon as threshold hits.

### Step 4: RDS in EC2

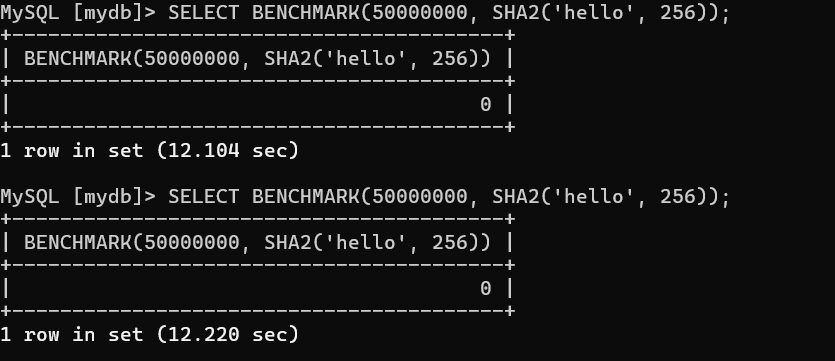
* Before doing anything, enable detailed monitoring to get faster response by going to monitoring section of instance.
* Click on manage detailed monitoring and enable it.



* Connect to the RDS from EC2 instance.

*sudo mysql -u admin -p -h your\_endpoint\_of\_RDS*

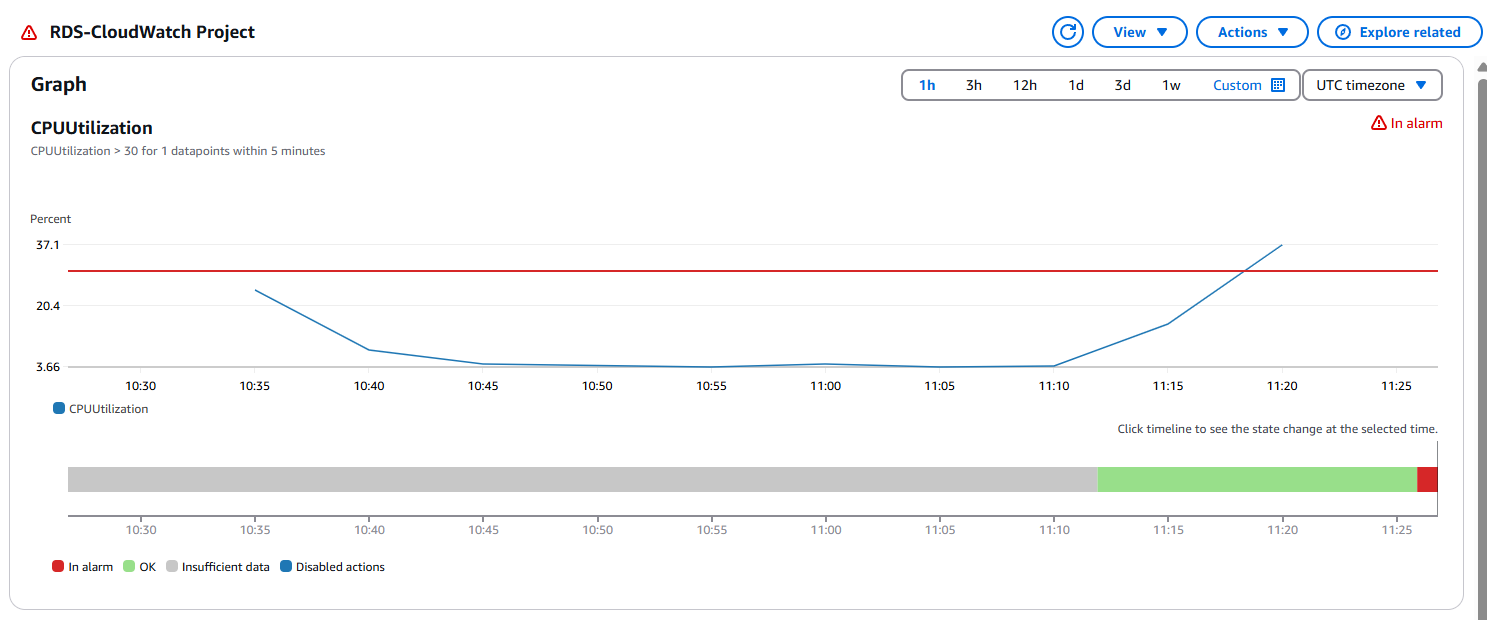
* Enter your password.
* Create a database: create database mydb;
* Use the database: use mydb:
* Run below command for CPU stress: SELECT BENCHMARK (50000000, SHA2('hello', 256));



* Keep running this command continuously and you can try to connect the instance from multiple powershell windows and put stress simultaneously.

### Step 5: Check CloudWatch

* Keep checking CloudWatch alarm and the worm will go above threshold value.



* If we see clearly the worm is above threshold value of 30 and alarm state is “in alarm” with danger sign and you might receive a mail too if you have confirmed subscription.

*Note:*

*Instead of creating topic while creating alarm, create a SNS topic manually, add email subscription and confirm it. After that you can select it from dropdown menu if you choose “select existing topic” option.*

*Turn off detailed monitoring of instance and delete all the resources after usage.*

In this way, we have completed our RDS monitoring for CPUUtilization metric using CloudWatch and added SNS topic and mail subscription to get notified.

♣♣♣