Amazon Services Setup

Table of Contents

Amazon RDS 2

Step 1: Setting Up the Amazon RDS Instance 2

Step 2: Connecting to the Amazon RDS Database 3

Amazon EC2 5

Step 1: Setting Up the EC2 Instance 5

Step 2: Connecting to the EC2 Instance via SSH 7

Step 3: Update and Install Required Software & Extensions (Run Everything in the Terminal) 8

Step 4: Upload the PHP Project to EC2 8

Step 5: Move Project Files to the Web Directory 9

Step 6: Install Composer and phpdotenv 9

Connecting to EC2 Instance 10

Step 1: Get Your RDS Endpoint & Credentials 10

Step 2: Allow EC2 to Connect to RDS 10

Step 3: Test Connection 11

Step 4: Restart Apache 11

Step 5: Add the .env file in nano /var/www/html/.env 11

Step 6: View the Website 12

Step 7: Disable Root Login for SSH 13

Setting Up Basic AWS CloudWatch for EC2 Instance 13

Step 1: View Metrics in AWS CloudWatch Console 13

Step 2: Setup CloudWatch Alarms 13

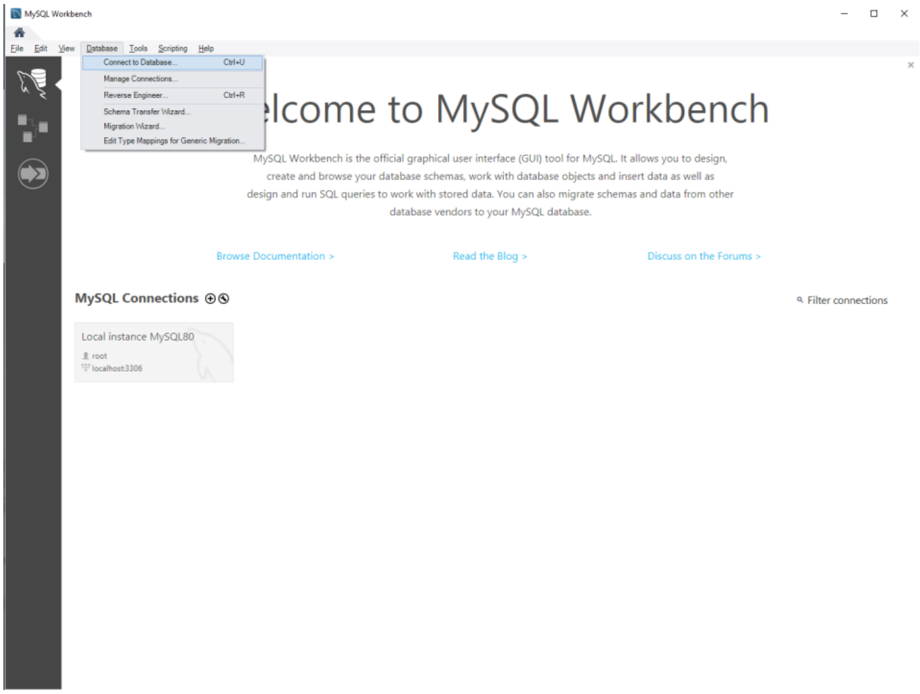
Amazon RDS

Step 1: Setting Up the Amazon RDS Instance

1. Launch the AWS Sandbox Environment. (Start Lab → AWS)
2. In the AWS Management Console, in the search box next to Services, search for and select RDS. In the left navigation pane, choose "Databases", then select "Create database".
3. Choose the following settings:
   - **Database Creation Method:** Standard create
   - **Engine Type:** MySQL
   - **Engine Version:** MySQL 8.0.40
   - **Templates:** Free Tier
   - **DB Instance Identifier:** HRMS-DB
   - **Master Username:** admin
   - **Credentials Management:** Self-Managed (unable to use AWS Secret Manager as it applies additional charges)
   - **Master Password:** k&Z(6F6A|9iy7Gs8
   - **Instance Configuration:** Burstable classes, db.t3.micro
   - **Storage:** General Purpose SSD (gp2) and 20GB, Enabled Storage Autoscaling.
   - **Public Access:** Yes
   - **VPC Security Group (Firewall):**
     - Create New
     - **New VPC Security Group Name:** HRMS-DB-SG
     - **Availability Zone:** No Preference
   - **Database Authentication:** Password Authentication
4. Click on "Create Database" and wait for the Status to turn to "Available". After that, you can proceed to Step 2.

Step 2: Connecting to the Amazon RDS Database

1. Open the MySQL Workbench
2. Click on Database → Connect to Database

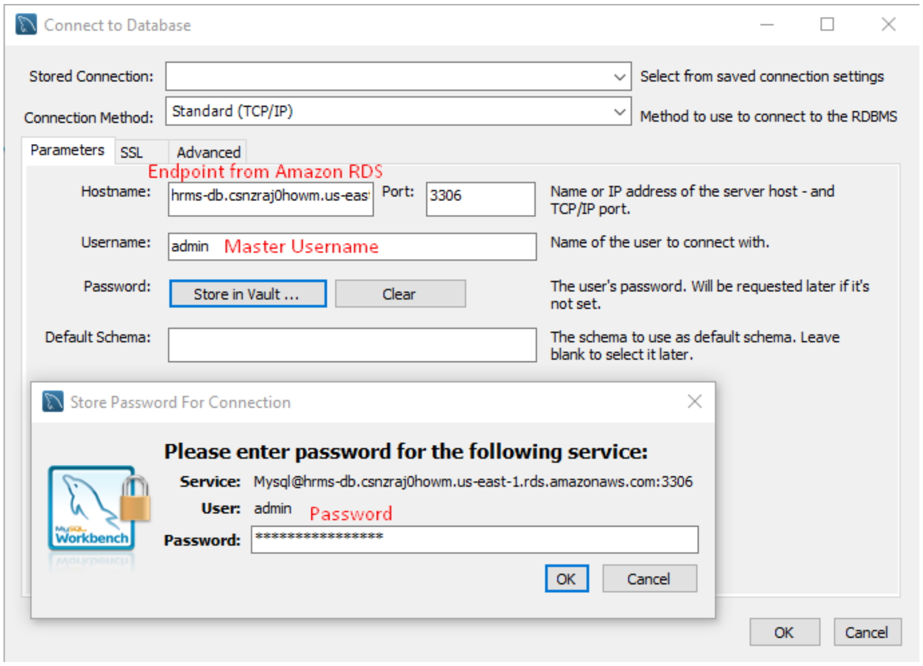


lcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Local instance MySQL80
root
localhost:3306

3. Fill in the details as follows and click on the Store in Vault. Paste the password "k&Z(6F6A|9iy7Gs8" into the field and click **"OK"**.



Stored Connection: Select from saved connection settings

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Endpoint from Amazon RDS

Hostname: hrms-db.csnzraj0howm.us-eas Port: 3306 Name or IP address of the server host - and TCP/IP port.

Username: admin Master Username Name of the user to connect with.

Password: The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

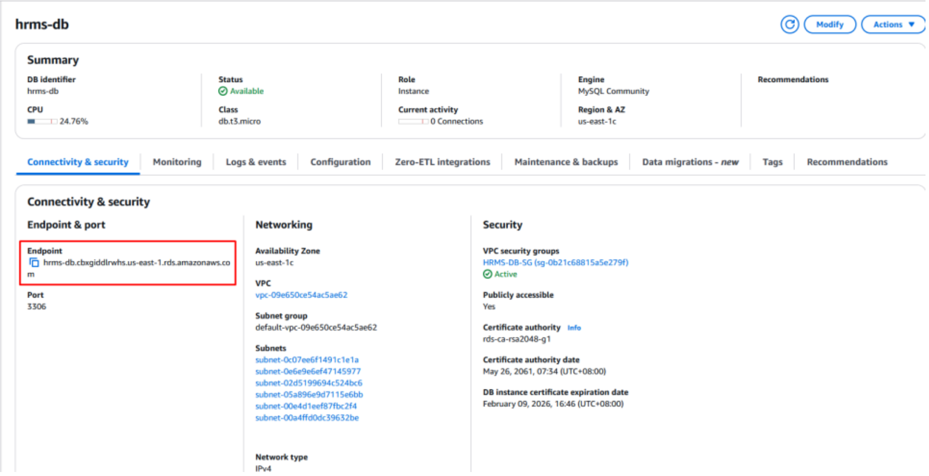
Please enter password for the following service:

Service: Mysql@hrms-db.csnzraj0howm.us-east-1.rds.amazonaws.com:3306

User: admin Password

Password: ****************

- The endpoint can be retrieved here:



hrms-db

Summary

DB identifier
hrms-db

CPU
24.76%

Status
Available

Class
db.t3.micro

Role
Instance

Current activity
0 Connections

Engine
MySQL Community

Region & AZ
us-east-1c

Recommendations

Connectivity & security

Endpoint & port

Endpoint
hrms-db.cbxgiddlrwhs.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1c

VPC
vpc-09e650ce54ac5ae62

Subnet group
default-vpc-09e650ce54ac5ae62

Subnets
subnet-0c07ee6f1491c1e1a
subnet-0e6e9e6ef47145977
subnet-02d5199694c524bc6
subnet-05a896e9d7115e6bb
subnet-00e4d1eef87fbc2f4
subnet-00a4ffd0dc39632be

Network type
IPv4

Security

VPC security groups
HRMS-DB-SG (sg-0b21c68815a5e279f)
Active

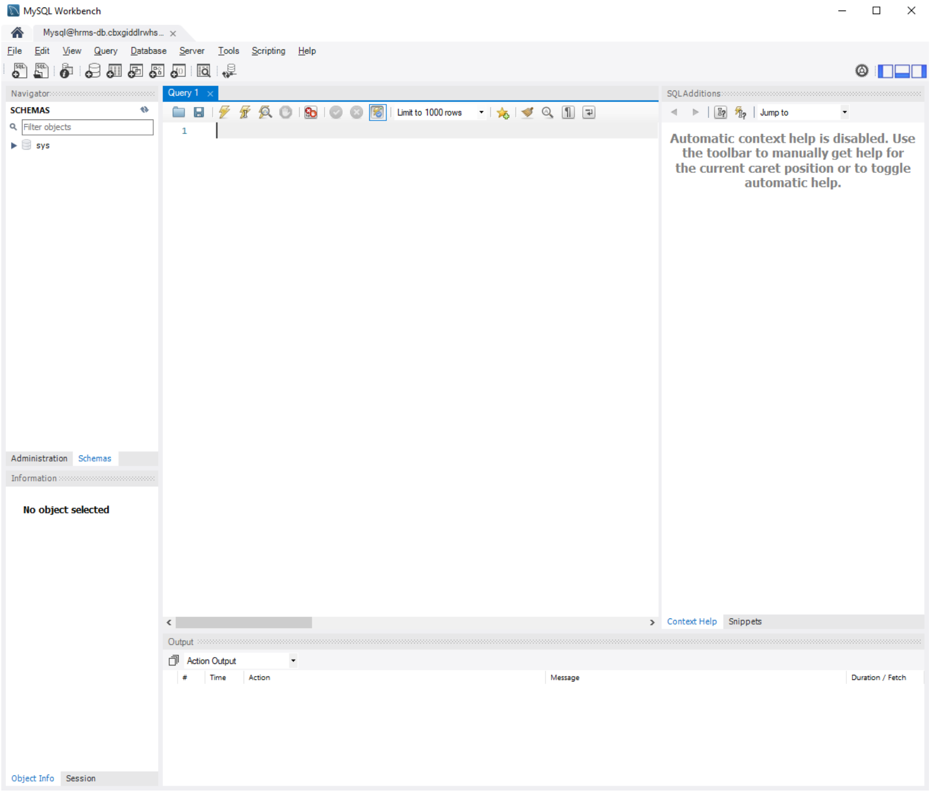
Publicly accessible
Yes

Certificate authority
rds-ca-rsa2048-g1

Certificate authority date
May 26, 2061, 07:34 (UTC+08:00)

DB instance certificate expiration date
February 09, 2026, 16:46 (UTC+08:00)

4. Click on "OK" to connect into the database in Amazon RDS.



Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

No object selected

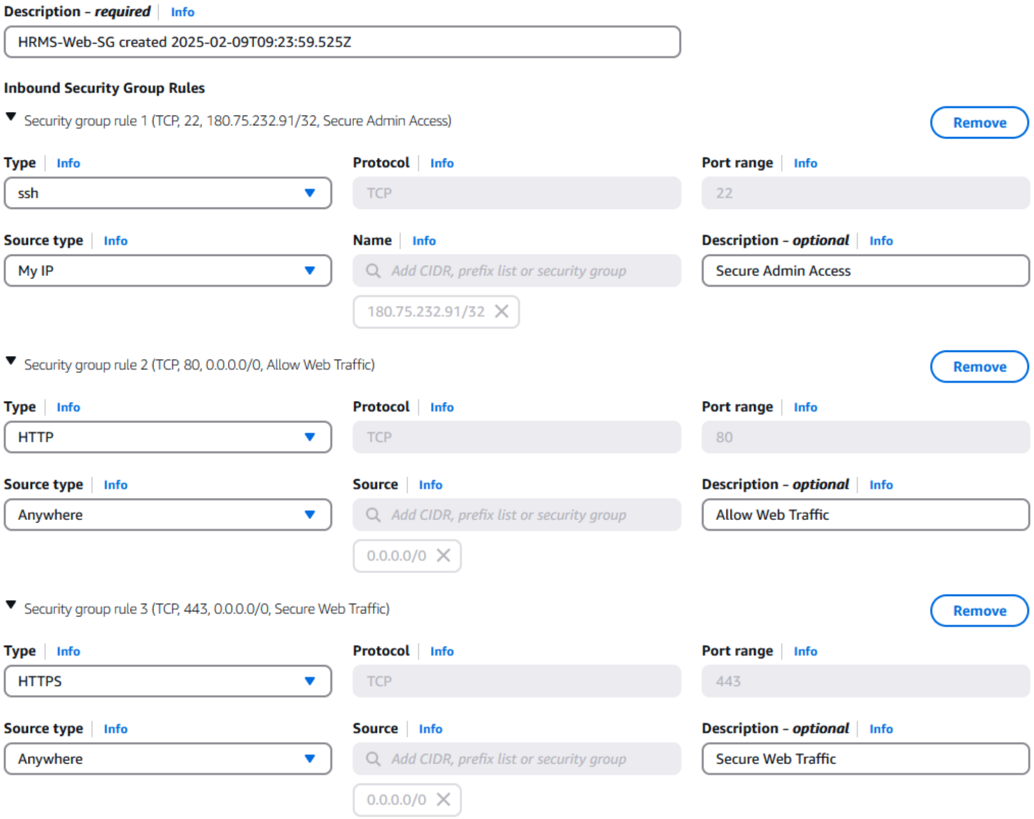
You should see something like this, and you can proceed to running the SQL script as shown in the SQL_Script PDF file. All the queries can be paste inside the Query Tab and Executed by clicking on the Lightning Icon beside the "Save" button. After the steps in the "SQL_Script.pdf" file has been followed, you can then proceed to setting up the **Amazon EC2**.

Amazon EC2

Step 1: Setting Up the EC2 Instance

1. Launch an EC2 Instance
2. Go to EC2 Dashboard → Launch instance.
    - **Name:** HRMSWebServer
    - **AMI:** Amazon Linux 2 AMI (HVM)
    - **Instance Type:** t3.medium
    - **Key pair (login):**
        - Create new key pair
        - **Key pair name:** id_rsa_hrmswebserver
        - **Key pair type:** RSA (recommended for compatibility)
        - **Private key format:** .pem
        - Click on create and the file will be downloaded into your computer. Later this file will be moved to another folder.
    - **Network Settings:**
        - Click on Edit to access more settings. Leave other settings as it is, we will focus on the Firewall (security groups). Click on Create security group.
        - **Security Group Name:** HRMS-Web-Server-SG
        - **Inbound Security Rules**

SSH	TCP	22	My IP (x.x.x.x/32)	Secure admin access
HTTP	TCP	80	0.0.0.0/0 (Anywhere IPv4)	Allow web traffic
HTTPS	TCP	443	0.0.0.0/0 (Anywhere IPv4)	Secure web traffic



Description - *required* | Info
HRMS-Web-SG created 2025-02-09T09:23:59.525Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 180.75.232.91/32, Secure Admin Access) — Remove

Type | Info	**Protocol** | Info	**Port range** | Info
ssh	TCP	22
Source type | Info	**Name** | Info	**Description - *optional*** | Info
My IP	Add CIDR, prefix list or security group 180.75.232.91/32 ✕	Secure Admin Access

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0, Allow Web Traffic) — Remove

Type | Info	**Protocol** | Info	**Port range** | Info
HTTP	TCP	80
Source type | Info	**Source** | Info	**Description - *optional*** | Info
Anywhere	Add CIDR, prefix list or security group 0.0.0.0/0 ✕	Allow Web Traffic

▼ Security group rule 3 (TCP, 443, 0.0.0.0/0, Secure Web Traffic) — Remove

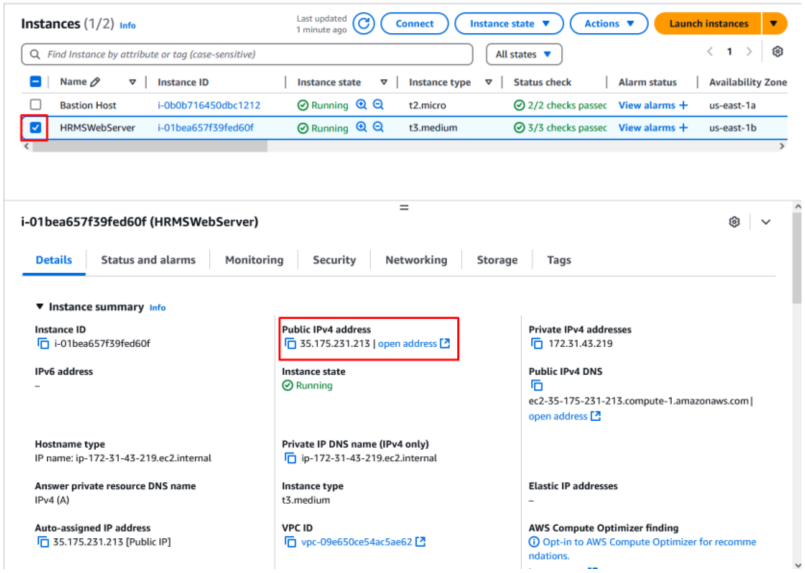
Type | Info	**Protocol** | Info	**Port range** | Info
HTTPS	TCP	443
Source type | Info	**Source** | Info	**Description - *optional*** | Info
Anywhere	Add CIDR, prefix list or security group 0.0.0.0/0 ✕	Secure Web Traffic

- **Storage Settings:**
  - Click on Advanced to access more settings.
  - Click on the arrow beside Volume 1 (AMI Root) (Custom).
  - Change Encrypted settings to “Encrypted”.
  - KMS Key to “Default”.
  - Since we cannot create role, we will just assign it to LabRole, as given by the Sandbox Environment. To do this, click on Advanced Details → IAM Instance Profile → Lab Instance Profile.

3. Launch the instance and wait for the Status Check of that instance to complete. After it shows 3/3 Checks Passed, you may proceed to Step 2.

Step 2: Connecting to the EC2 Instance via SSH

1. Create a folder (if you don’t have) in C:\Users\your-username\.ssh and paste your id_rsa_hrmswebserver.pem file downloaded into this folder.
2. Go to AWS Console → EC2 → Instances → Select the HRMSWebServer EC2 Instance.
3. Copy the Public IPv4 Address of your instance.



☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
☐	Bastion Host	i-0b0b716450dbc1212	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1a
☑	HRMSWebServer	i-01bea657f39fed60f	✓ Running	t3.medium	✓ 3/3 checks passec	View alarms +	us-east-1b

i-01bea657f39fed60f (HRMSWebServer)

▼ Instance summary Info

Instance ID
i-01bea657f39fed60f

IPv6 address
–

Hostname type
IP name: ip-172-31-43-219.ec2.internal

Answer private resource DNS name
IPv4 (A)

Auto-assigned IP address
35.175.231.213 [Public IP]

Public IPv4 address
35.175.231.213 | open address

Instance state
✓ Running

Private IP DNS name (IPv4 only)
ip-172-31-43-219.ec2.internal

Instance type
t3.medium

VPC ID
vpc-09e650ce54ac5ae62

Private IPv4 addresses
172.31.43.219

Public IPv4 DNS
ec2-35-175-231-213.compute-1.amazonaws.com |
open address

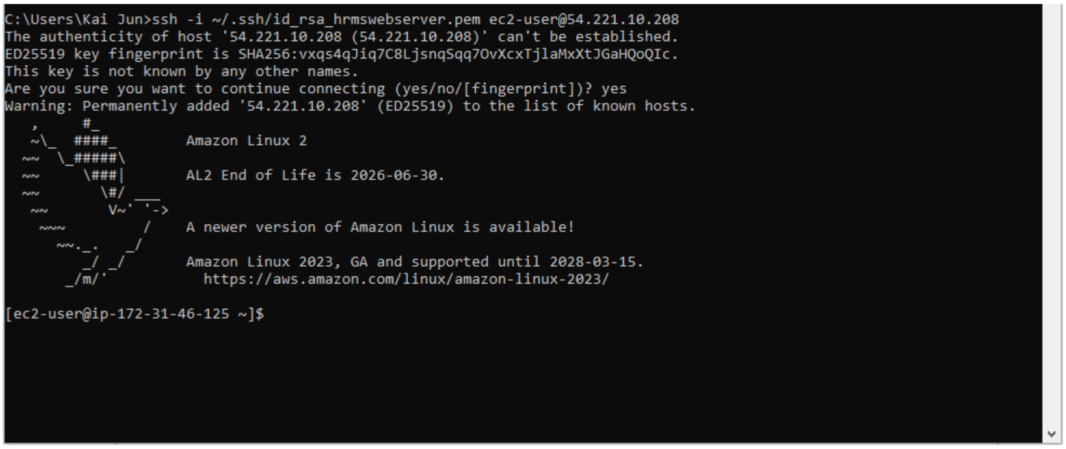
Elastic IP addresses
–

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations.

4. On your local machine, run the Command Prompt as Administrator and paste the following command:

   ssh -i ~/.ssh/id_rsa_hrmswebserver.pem ec2-user@your-ec2-public-ip

5. Type “yes” and you should see the following:



```
C:\Users\Kai Jun>ssh -i ~/.ssh/id_rsa_hrmswebserver.pem ec2-user@54.221.10.208
The authenticity of host '54.221.10.208 (54.221.10.208)' can't be established.
ED25519 key fingerprint is SHA256:vxqs4qJiq7C8LjsnqSqq7OvXcxTjlaMxXtJGaHQoQIc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.221.10.208' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-46-125 ~]$
```

6. Proceed to Step 3.

Step 3: Update and Install Required Software & Extensions (Run Everything in the Terminal)

1. Since we are running a PHP web app, Apache/Nginx, PHP, and MySQL client will be necessary.
2. **Update Command:**

```
sudo yum update -y
```

3. **Required Dependencies and Extensions:**

```
sudo yum install -y httpd
sudo amazon-linux-extras enable php8.2
sudo yum clean metadata
sudo yum install -y php php-bz2 php-curl php-fileinfo php-gettext php-mbstring php-exif php-mysqli php-pdo php-pdo_mysql php-pdo_sqlite php-json php-xml unzip
```

4. **Start and Enable Apache Web Server:**

```
sudo systemctl start httpd
sudo systemctl enable httpd
sudo setsebool -P httpd_can_network_connect on
```

5. Proceed to Step 4.

Step 4: Upload the PHP Project to EC2

1. Using SCP (Secure Copy) from Your Local Machine.
2. Run Windows Powershell as Administrator. Copy-paste the code below into the Powershell, run and wait for the files to be transferred (5 to 10 minutes or longer).
3. scp -i your-key.pem -r /path/to/your/php-project ec2-user@your-ec2-public-ip:/home/ec2-user/

   For Me: scp -i "C:\Users\Kai Jun\.ssh\id_rsa_hrmswebserver.pem" -r "C:\xampp\htdocs\CCS6334-Assignment-2-Group-5" ec2-user@18.234.197.251:/home/ec2-user/
4. After you have done transferring the files, type exit on the Windows Powershell and proceed to Step 5.

Step 5: Move Project Files to the Web Directory

1. Navigate to Apache's default web root directory

```
cd /var/www/html
```

2. Move the PHP project files to this directory

```
sudo mv /home/ec2-user/simple-hrms-main/* /var/www/html/
```

3. Change ownership and permissions:

```
sudo chown -R apache:apache /var/www/html
sudo chmod -R 755 /var/www/html
```

4. Proceed to Step 6.

Step 6: Install Composer and phpdotenv

1. **Paste the following code one by one into the Command Prompt:**

```
sudo php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
sudo php -r "if (hash_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c636f7cd84f6db9189d43a81e5503cda447da73c7e5b6') { echo 'Installer verified'.PHP_EOL; } else { echo 'Installer corrupt'.PHP_EOL; unlink('composer-setup.php'); exit(1); }"
sudo php composer-setup.php
sudo php -r "unlink('composer-setup.php');"
sudo mv composer.phar /usr/local/bin/composer
composer -V
sudo chown -R ec2-user:ec2-user /var/www/html
composer require vlucas/phpdotenv
```

2. Proceed to **Connecting to EC2 Instance**.

Connecting to EC2 Instance

Step 1: Get Your RDS Endpoint & Credentials

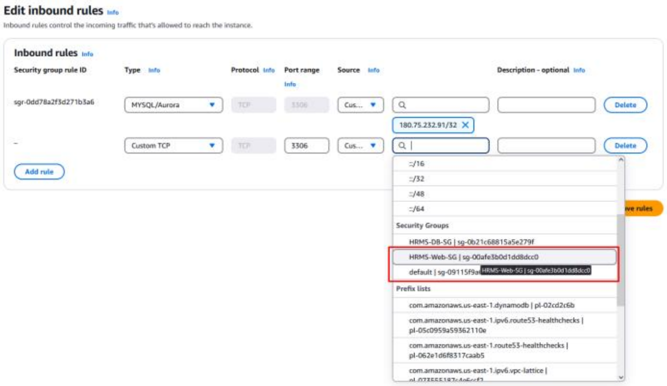
1. Go to AWS Console → RDS → Databases.
2. Click on your database.
3. Copy your Endpoint (e.g., yourdb.xxxxxxx.us-east-1.rds.amazonaws.com).
4. Note Down the:
    - Database Name
    - Master Username
    - Master Password

- For My Case:
    - **Database Name**: hrms-db
    - **Endpoint:** hrms-db.csnzraj0howm.us-east-1.rds.amazonaws.com
    - **Master Username:** admin
    - **Master Password:** k&Z(6F6A|9iy7Gs8

Step 2: Allow EC2 to Connect to RDS

1. Go to AWS Console → EC2 → Security Groups.
2. Find the VPC Security Group attached to your RDS Instance.
3. Edit Inbound Rules, add:

- **Type:** MySQL/Aurora
- **Port:** 3306 (for MySQL)
- **Source:** Your EC2 security group. (HRMS-Web-SG)



Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0dd78a2f3d271b3a6	MYSQL/Aurora	TCP	3306	Cus...	180.75.232.91/32		Delete
-	Custom TCP	TCP	3306	Cus...			Delete

::/16

::/32

::/48

::/64

Security Groups

HRMS-DB-SG | sg-0b21c68815a5e279f

HRMS-Web-SG | sg-00afe3b0d1dd8dcc0

default | sg-09115f9a HRMS-Web-SG | sg-00afe3b0d1dd8dcc0

Prefix lists

com.amazonaws.us-east-1.dynamodb | pl-02cd2c6b

com.amazonaws.us-east-1.ipv6.route53-healthchecks | pl-05c0959a59362110e

com.amazonaws.us-east-1.route53-healthchecks | pl-062e1d6f8317caab5

com.amazonaws.us-east-1.ipv6.vpc-lattice |

Step 3: Test Connection

1. Install MySQL (in EC2 Instance):

```
sudo yum install -y mysql
mysql --version
```

2. Test Connection:

```
mysql -h yourdb.xxxxxxx.us-east-1.rds.amazonaws.com -u yourusername -p
```

For My Case:

```
mysql -h hrms-db.cbxgiddlrwhs.us-east-1.rds.amazonaws.com -u admin -p
```

Password:

k&Z(6F6A|9iy7Gs8 **(Type it)**

3. Type “SHOW Databases;” you should see. Type exit and proceed to Step 4.

Step 4: Restart Apache

```
sudo systemctl restart httpd
```

Step 5: Add the .env file in nano /var/www/html/.env

Copy the .env file from your VSCode .env file and write it into the .env file in EC2 Instance. After done writing, press Ctrl + O, then Enter. Exit with Ctrl + X. The .env file should be something like this:

```
DB_HOST=hrms-db.cbxgiddlrwhs.us-east-1.rds.amazonaws.com
DB_DATABASE=HRMS
DB_USERNAME=admin
DB_PASSWORD=k&Z(6F6A|9iy7Gs8
```

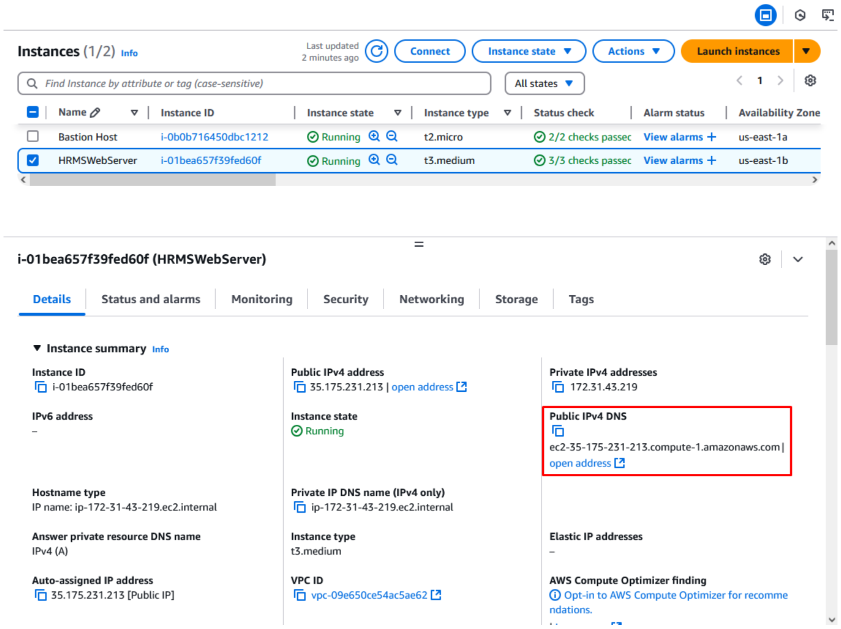
Type out “ls -a” and you should see the .env file.

Reasoning: We decided to choose Amazon Linux 2 as it comes with AWS CLI pre-installed, making it easier to interact with S3, RDS, and IAM. In addition, the Sandbox Environment only support Amazon provided Linux and Microsoft Windows AMIs. Moreover, Amazon Linux 2 is lightweight and optimized for performance in AWS. We decided to select t3.medium since the

HRMS system is a medium-traffic PHP app, better for MySQL. Moreover, it has good balance between cost and performance for PHP & MySQL workloads and burstable performance (saves cost when under low CPU usage). We wanted to use Amazon Route 53 to setup the DNS, however it wasn't allowed in the Sandbox Environment.

Step 6: View the Website

1. Type **http://<public_ipv4_dns>/src/login.php** on your browser (MUST BE HTTP not HTTPS).



Instances (1/2)

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
☐	Bastion Host	i-0b0b716450dbc1212	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1a
☑	HRMSWebServer	i-01bea657f39fed60f	✓ Running	t3.medium	✓ 3/3 checks passec	View alarms +	us-east-1b

i-01bea657f39fed60f (HRMSWebServer)

▼ Instance summary

Instance ID
i-01bea657f39fed60f

Public IPv4 address
35.175.231.213 | open address

Private IPv4 addresses
172.31.43.219

IPv6 address
–

Instance state
✓ Running

Public IPv4 DNS
ec2-35-175-231-213.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-43-219.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-43-219.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t3.medium

Elastic IP addresses
–

Auto-assigned IP address
35.175.231.213 [Public IP]

VPC ID
vpc-09e650ce54ac5ae62

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations.

Due to the restrictions of the AWS Academy Cloud Foundations Sandbox Environment, HTTPS implementation was not feasible.

- The sandbox does not allow Route 53 domain registration, which prevents obtaining an FQDN required for SSL.
- AWS Certificate Manager (ACM) cannot issue certificates in this environment.
- Public IPv4 DNS cannot be used to obtain an SSL certificate from Let's Encrypt.

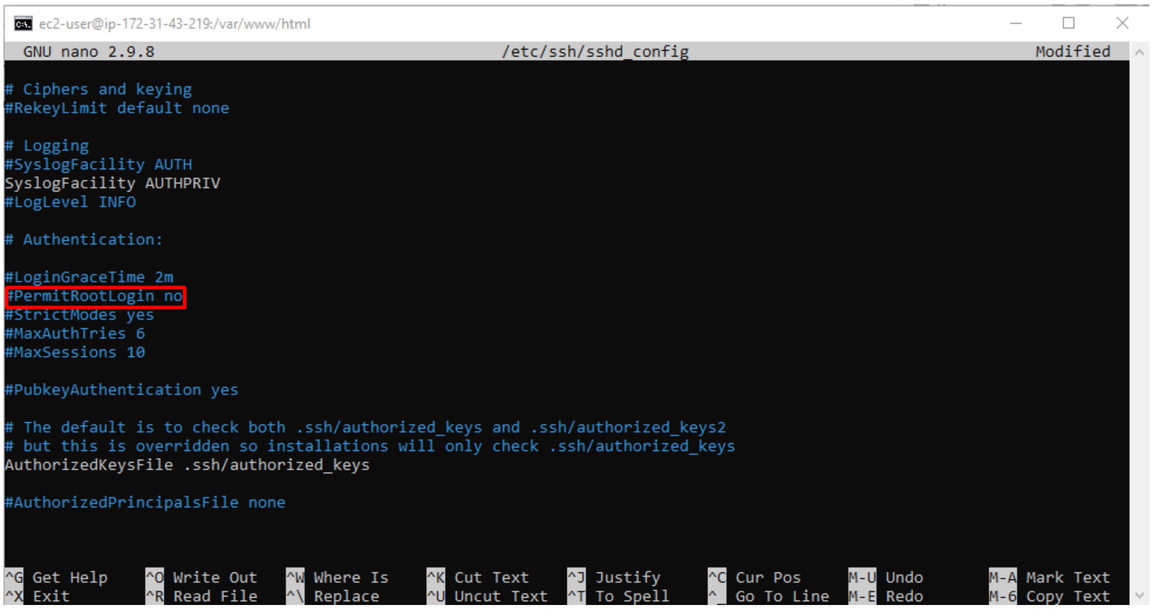
As a result, while security measures such as IAM role-based access control, security groups, and database authentication were implemented, HTTPS could not be configured. If deployed in a production environment, this issue could be resolved by registering a domain, enabling ACM, and configuring SSL/TLS with CloudFront or ELB.

Step 7: Disable Root Login for SSH

1. Type the following commands:

   sudo nano /etc/ssh/sshd_config

   PermitRootLogin no



ec2-user@ip-172-31-43-219:/var/www/html

```
  GNU nano 2.9.8                      /etc/ssh/sshd_config                      Modified

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none


^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo      M-A Mark Text
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line M-E Redo      M-6 Copy Text
```

2. Press on Ctrl + O, then Enter, then Ctrl + X.
3. Type “sudo systemctl restart sshd” to restart Apache.
4. Proceed to **Setting Up Basic AWS CloudWatch for EC2 Instance**.

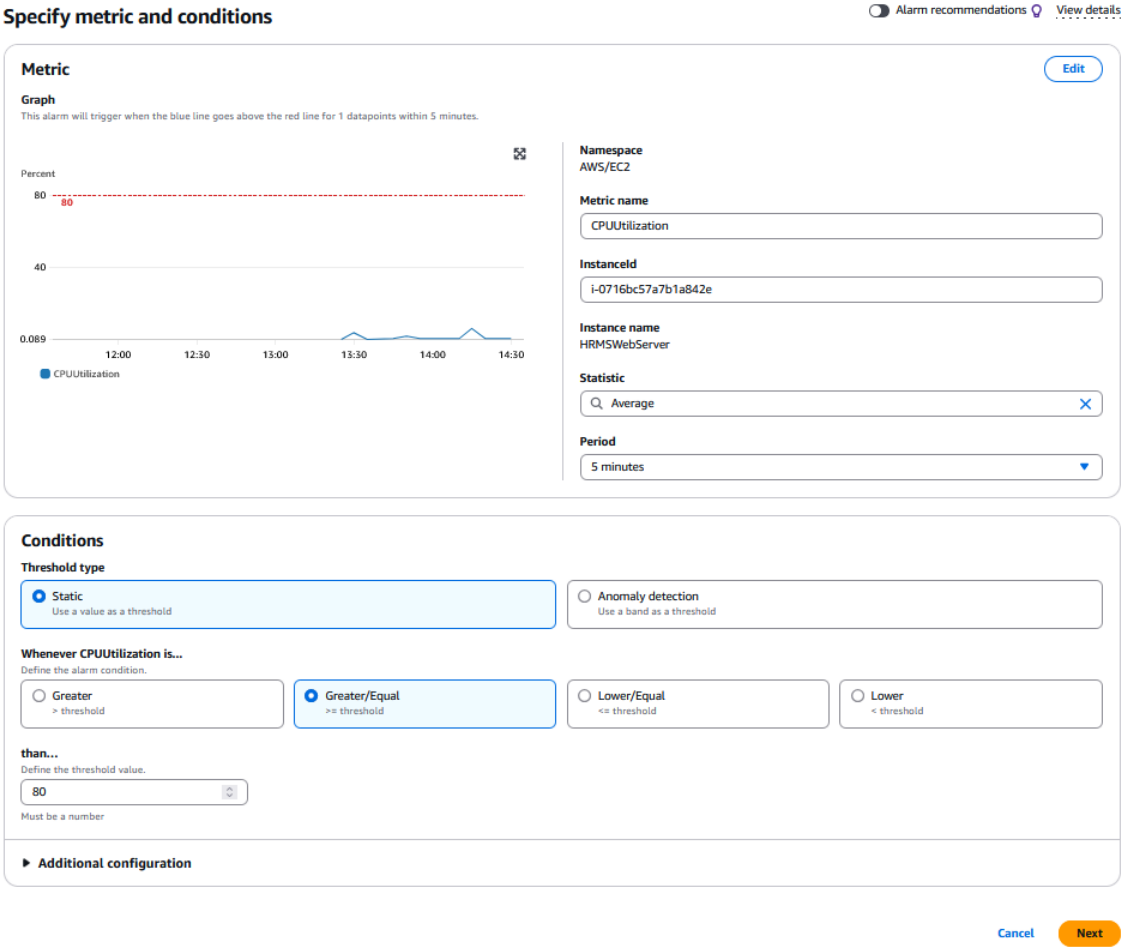
Setting Up Basic AWS CloudWatch for EC2 Instance

Step 1: View Metrics in AWS CloudWatch Console

1. Go to AWS Console → CloudWatch → Metrics.
2. Navigate to CWAgent → EC2 Instance Metrics.
3. Look for CPU, Memory, Disk, and Network Usage.

Step 2: Setup CloudWatch Alarms

1. Go to CloudWatch → Alarms → Create Alarm.
2. Setup for **CPUUtilization to prevent server overload**, ensuring the HRMS remains responsive. Threshold: 80% for 5 minutes.



Specify metric and conditions

Metric

Graph

This alarm will trigger when the blue line goes above the red line for 1 datapoints within 5 minutes.

Namespace
AWS/EC2

Metric name
CPUUtilization

InstanceId
i-0716bc57a7b1a842e

Instance name
HRMSWebServer

Statistic
Average

Period
5 minutes

Conditions

Threshold type

● Static
Use a value as a threshold

○ Anomaly detection
Use a band as a threshold

Whenever CPUUtilization is...
Define the alarm condition.

○ Greater
> threshold

● Greater/Equal
>= threshold

○ Lower/Equal
<= threshold

○ Lower
< threshold

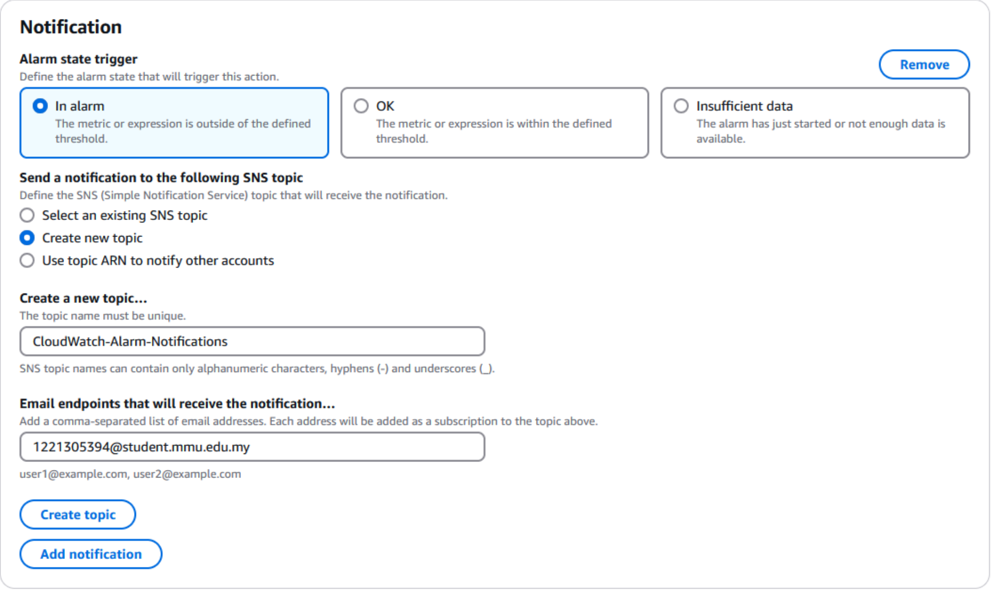
than...
Define the threshold value.

80

Must be a number

▸ **Additional configuration**

3. Create a new topic and set the endpoints email who will receive the notification.



Notification

Alarm state trigger
Define the alarm state that will trigger this action.

● In alarm
The metric or expression is outside of the defined threshold.

○ OK
The metric or expression is within the defined threshold.

○ Insufficient data
The alarm has just started or not enough data is available.

Send a notification to the following SNS topic
Define the SNS (Simple Notification Service) topic that will receive the notification.

○ Select an existing SNS topic
● Create new topic
○ Use topic ARN to notify other accounts

Create a new topic...
The topic name must be unique.

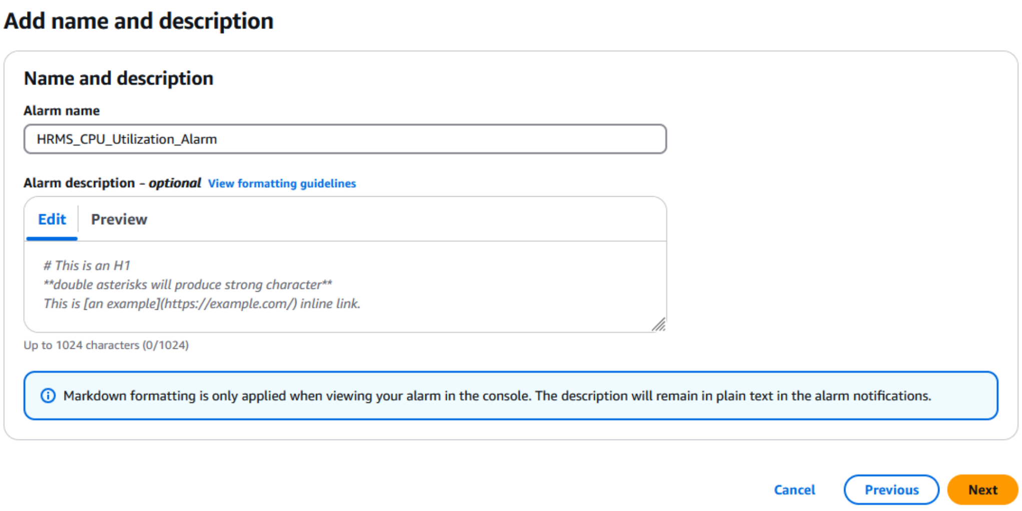
CloudWatch-Alarm-Notifications

SNS topic names can contain only alphanumeric characters, hyphens (-) and underscores (_).

Email endpoints that will receive the notification...
Add a comma-separated list of email addresses. Each address will be added as a subscription to the topic above.

1221305394@student.mmu.edu.my

user1@example.com, user2@example.com



Add name and description

Name and description

Alarm name

HRMS_CPU_Utilization_Alarm

Alarm description - *optional* View formatting guidelines

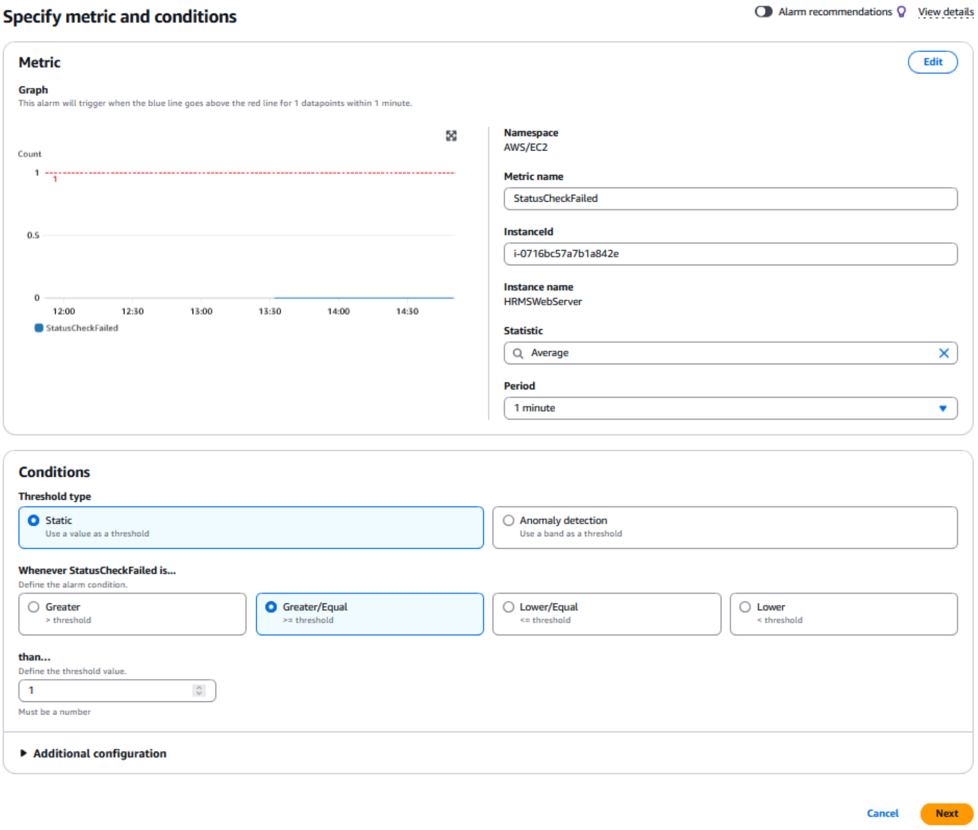
# This is an H1
double asterisks will produce strong character
This is [an example](https://example.com/) inline link.

Up to 1024 characters (0/1024)

ⓘ Markdown formatting is only applied when viewing your alarm in the console. The description will remain in plain text in the alarm notifications.

4. Click “Next” then click on Create Alarm.

5. Setup for **StatusCheckFailed to detects if the instance is unhealthy**.



Specify metric and conditions

Metric

Graph

This alarm will trigger when the blue line goes above the red line for 1 datapoints within 1 minute.

Namespace
AWS/EC2

Metric name
StatusCheckFailed

InstanceId
i-0716bc57a7b1a842e

Instance name
HRMSWebServer

Statistic
Average

Period
1 minute

Conditions

Threshold type

● Static
Use a value as a threshold

○ Anomaly detection
Use a band as a threshold

Whenever StatusCheckFailed is...
Define the alarm condition.

○ Greater
> threshold

● Greater/Equal
>= threshold

○ Lower/Equal
<= threshold

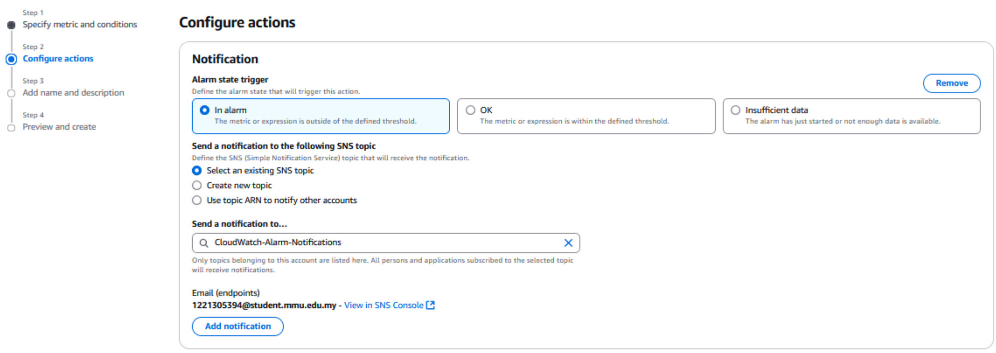
○ Lower
< threshold

than...
Define the threshold value.

1

Must be a number

▸ **Additional configuration**



Configure actions

Notification

Alarm state trigger

Define the alarm state that will trigger this action.

● In alarm — The metric or expression is outside of the defined threshold.

○ OK — The metric or expression is within the defined threshold.

○ Insufficient data — The alarm has just started or not enough data is available.

Send a notification to the following SNS topic

Define the SNS (Simple Notification Service) topic that will receive the notification.

● Select an existing SNS topic

○ Create new topic

○ Use topic ARN to notify other accounts

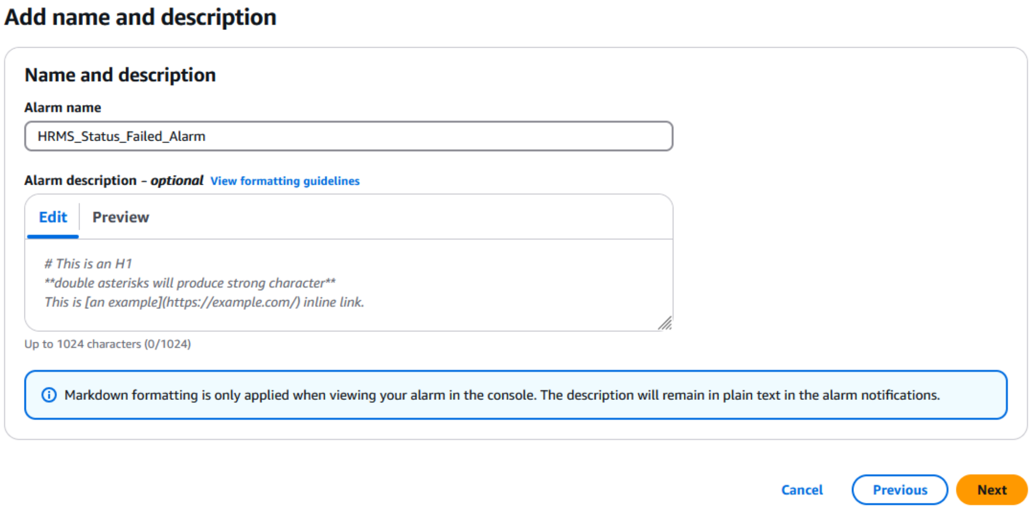
Send a notification to...

CloudWatch-Alarm-Notifications

Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications.

Email (endpoints)

1221305394@student.mmu.edu.my - View in SNS Console



Add name and description

Name and description

Alarm name

HRMS_Status_Failed_Alarm

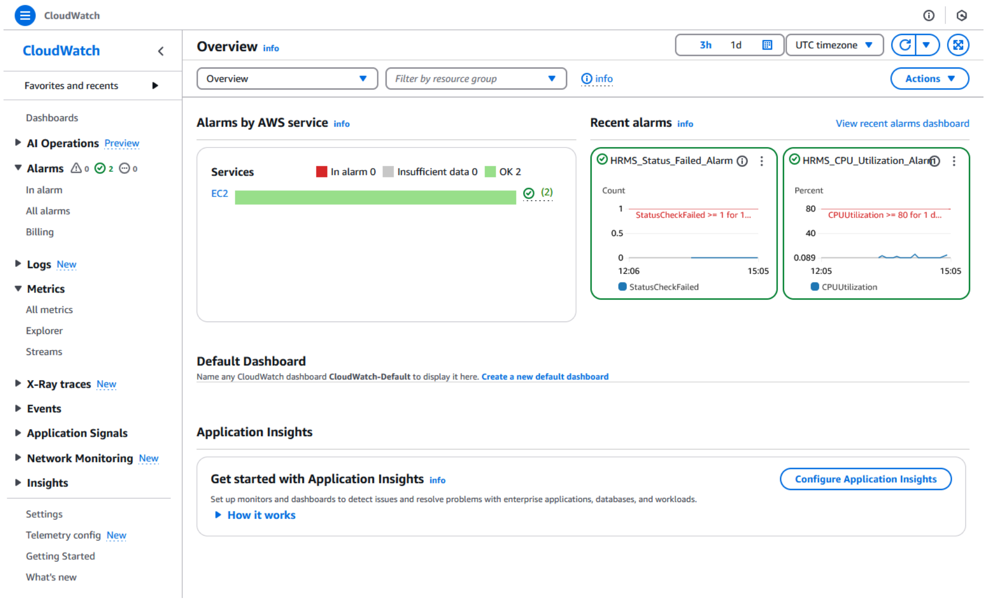
Alarm description - *optional* View formatting guidelines

# This is an H1
double asterisks will produce strong character
This is [an example](https://example.com/) inline link.

Up to 1024 characters (0/1024)

ⓘ Markdown formatting is only applied when viewing your alarm in the console. The description will remain in plain text in the alarm notifications.

6. Click “Next” then click on Create Alarm.
7. From the CloudWatch Overview, you should see the following after setting up the two alarms.



Overview info

Alarms by AWS service info

Services	🟥 In alarm 0	▨ Insufficient data 0	🟩 OK 2
EC2			✓ (2)

Recent alarms info — View recent alarms dashboard

✓ HRMS_Status_Failed_Alarm

✓ HRMS_CPU_Utilization_Alarm

Default Dashboard

Name any CloudWatch dashboard **CloudWatch-Default** to display it here. **Create a new default dashboard**

Application Insights

Get started with Application Insights info

Set up monitors and dashboards to detect issues and resolve problems with enterprise applications, databases, and workloads.

▸ How it works