



NextWork.org

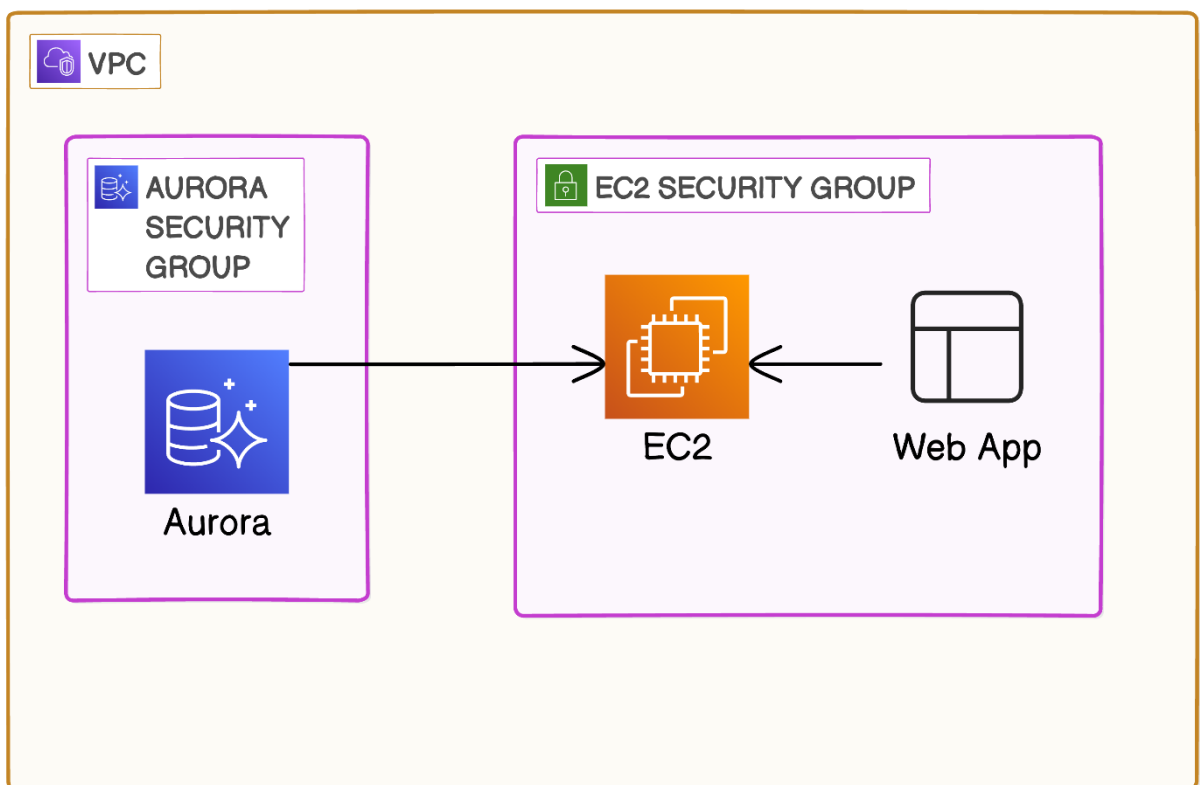
Connect A Web App with Aurora



Prajit Venkatachalam

<https://www.linkedin.com/in/prajit-venkatachalam/>

Aurora to Web App





Introducing Today's Project!

What is Amazon Aurora?

Amazon Aurora is an AWS relational database service, meaning it stores data in rows and columns. Aurora is particularly well-suited for handling high-performance and high-volume data use cases.

How I used Amazon Aurora in this project

In this project, I used Amazon Aurora to connect to a web app. This allowed us to input data through the browser (by loading the web app) and see the updates reflected in the backend Aurora database.

One thing I didn't expect in this project was...

One thing I didn't expect in this project was that manually updating the database in the backend would still update the table displayed on the frontend by the web app.

This project took me...

This project took me 3.5 hours to complete include report writing.

Launch an EC2 instance for your web app

In this step, I will launch an EC2 instance, which will later be used to host the web app in this project.

Create an Aurora MySQL Database

In this step, I am going to create a relational database from scratch using Amazon Aurora.

Creating a Web App

In this step, I will set up an EC2 instance and then install a web app to run on the instance.

First, I connected to my EC2 instance from my local computer via SSH using command terminal.

To connect to my EC2 instance, I used a key pair that was downloaded to my local computer and utilized its details to SSH into the EC2 instance from my local computer's terminal.

[illegible]



To help me create my web app, I first installed PHP, MariaDB, Apache web server, and php-mysql. These tools enabled me to set up a web server and provided the functionality for my web app and EC2 instance to interact with my database.

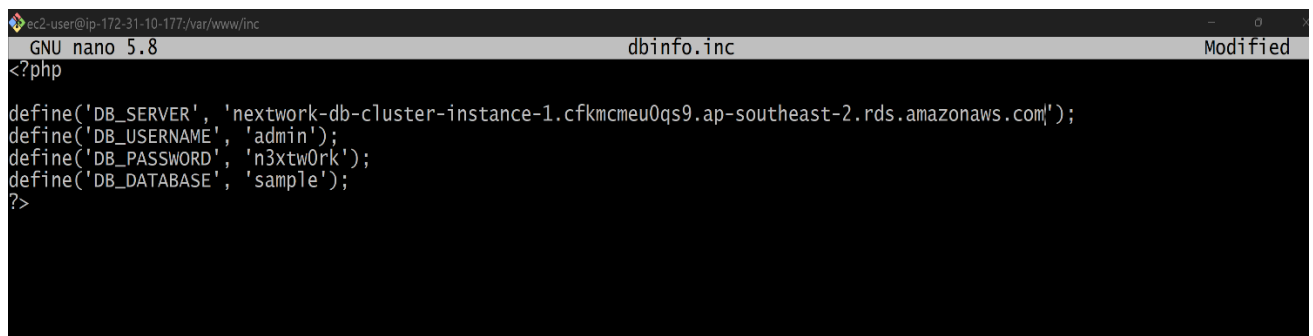
I used this command to install the above mentioned softwares.

```
sudo dnf install -y httpd php php-mysqli mariadb105
```

I started the most basic version of our web app with the following command: `sudo systemctl start httpd`

Connecting my Web App to Aurora

In this step, I upgraded my web app to capture user information, which I can store in my Aurora database.



```
ec2-user@ip-172-31-10-177:/var/www/inc
GNU nano 5.8 dbinfo.inc Modified
<?php
define('DB_SERVER', 'nextwork-db-cluster-instance-1.cfkmcmeu0qs9.ap-southeast-2.rds.amazonaws.com');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'n3xtw0rk');
define('DB_DATABASE', 'sample');
?>
```

I set up my EC2 instance's connection to the database by creating a file: dbinfo.inc and storing details like username, password, endpoint, and database name. The EC2 instance can now use this file to access the necessary connection information. Code used in dbinfo.inc

```
<?php

define('DB_SERVER', 'YOUR_ENDPOINT');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'n3xtw0rk');
define('DB_DATABASE', 'sample');
?>
```



My Web App Upgrade

Next, I upgraded my web app by creating a new PHP file that connects to the database and added some light frontend work, including a submission form and a table.

In this step, I create a new file in the html directory named SamplePage.php, and then edit the file by calling nano

The code for PHP file is included in a separate file.

Sample page

NAME		ADDRESS
<input type="text"/>		<input type="text"/>
<input type="button" value="Add Data"/>		

ID	NAME	ADDRESS
1	Mr Eddy Example	100 Example Street, New York
2	Miss Sarah	02 Silver Lane, San Francisco



Testing my Web App

In this step, we are testing the web app in the browser to ensure it's functioning correctly and validating that the database is being updated by using the My SQL CLI

To ensure my web app was working correctly, I downloaded software (MySQL CLI) that allows running SQL queries from the command line. I executed queries to select all data from a table in my database and verified that the table contained all the input.

Download the MySQL repository into your EC2 instance:

```
sudo yum install https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm -y
```

Install MySQL:

```
sudo yum install mysql-community-client -y
```

Connect to your Aurora MySQL Database:

```
mysql -h YOUR_ENDPOINT -P YOUR_PORT -u  
YOUR_AURORA_USERNAME -p
```

```
MySQL [sample]> SELECT * FROM EMPLOYEES;
```

ID	NAME	ADDRESS
1	Mr Eddy Example	100 Example Street, New York
2	Miss Sarah	02 Silver Lane, San Francisco

```
2 rows in set (0.001 sec)
```



NextWork.org

**Everyone
should be in a
job they love.**

Check out nextwork.org for
more projects