Lab 2: Build Your DB Server and Interact With Your DB Using an App

This lab builds on the previous lab. This lab is designed to reinforce the concept of leveraging an AWS-managed database instance for solving relational database needs.

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, which allows you to focus on your applications and business. Amazon RDS provides you with six familiar database engines to choose from: Amazon Aurora, Oracle, Microsoft SQL Server, PostgreSQL, MySQL and MariaDB.

Amazon RDS ***Multi-AZ*** deployments provide enhanced availability and durability for Database (DB) instances, making them a natural fit for production database workloads. When you provision a Multi-AZ DB instance, Amazon RDS automatically creates a primary DB instance and synchronously replicates the data to a standby instance in a different Availability Zone (AZ).

Objectives

After completing this lab, you can:

- Launch an Amazon RDS DB instance with high availability.
- Configure the DB instance to permit connections from your web server.
- Open a web application and interact with your database.

Duration

This lab takes approximately **45 minutes**.

Access the AWS Management Console

Task 1: Create a VPC Security Group for the RDS DB Instance

In this task, you create a VPC security group to allow your web server to access your RDS DB instance.

1. [1] In the **AWS Management Console,** on the **Services** menu, click **VPC**.
2. In the navigation pane, click **Security Groups**.
3. Click **Create Security Group**.
4. In the **Create Security Group** dialog box, configure the following settings (and ignore any settings that aren’t listed):
    - **Name tag**: type `DBSecurityGroup`
      Ignore errors about description being required.
      That will be fixed below.

    - **Group name**: type `DBSecurityGroup`
      It will be filled automatically.
    - **Description**: type `DB Instance Security Group`
    - **VPC**: Click **My Lab VPC**.
5. Click **Yes, Create**.
6. Select the security group you just created (**DBSecurityGroup**) and ensure that all other security groups are not selected.
7. Click the **Inbound Rules** tab, and then click **Edit**.
8. Configure the following settings (and ignore any settings that aren't listed):
    - **Type**: Click `MySQL/Aurora (3306)`.
    - **Protocol**: Click **TCP(6)**.
    - **Source**: Click **WebSecurityGroup**.
9. Click **Save**.

Task 2: Create a DB Subnet Group

In this task, you create a DB subnet group that is used to tell RDS which subnets can be used for the database. Each DB subnet group should have subnets in at least two Availability Zones in a given region.

1. [10] On the **Services** menu, click **RDS**.
2. In the navigation pane, click **Subnet Groups**.
3. Click **Create DB Subnet Group**.
4. On the **Create DB Subnet Group** page, configure the following settings (and ignore any settings that aren't listed):
    - **Name**: type `dbgroup`
    - **Description**: type `DB Instance Subnet Group`
    - **VPC ID**: Click **My Lab VPC**.
5. For **Availability Zone**, select the first Availability Zone.
6. For **Subnet ID**, select **10.0.3.0/24**.
7. Click **Add**.
8. For **Availability Zone**, click the second Availability Zone. This adds another subnet to the DB subnet group.
9. For **Subnet ID**, click **10.0.4.0/24**.
10. Click **Add**.
11. Click **Create**.
    If you do not see your new subnet group, click the refresh icon in the upper-right corner of the console.

Task 3: Create an RDS DB Instance

In this task, you configure and launch your MySQL-backed Amazon RDS DB instance.

1. [26] In the navigation pane, click **Instances**.

2. Click **Launch DB Instance**.
3. Click **MySQL** > **Select**.
4. Under **Production**, click **MySQL**.
5. Click **Next Step**.

6. On the **Specify DB Details** page, configure the following settings (and ignore any settings that aren't listed):

   - **DB Instance Class**: Click the first option in the list.
     (for example: **db.t2.micro**)
   - **Multi-AZ Deployment**: Click **Yes**.
   - **DB Instance Identifier**: type `Lab5DB`
   - **Master Username**: type `labuser`
   - **Master Password**: type `labpassword`
   - **Confirm Password**: type `labpassword`

7. Click **Next Step**.

8. On the **Configure Advanced Settings** page, configure the following settings (and ignore any settings that aren't listed):
   - **VPC**: Click **My Lab VPC**.
   - **Subnet Group**: Click **dbgroup**.
   - **Publically Accessible**: Click **No**.
   - **VPC Security Group(s)**: Click **DBSecurityGroup (VPC)**.
   - **Database Name**: type `sampleDB`
   - **Enable Enhanced Monitoring**: Click **No**.
9. Click **Launch DB Instance**.
10. Click **View Your DB Instances**.
11. Select **DB1** and wait until the endpoint is *available* or *modifying*, or has transitioned from *Not available yet* to a string ending with *3306*; this may take up to 10 minutes. Click the refresh icon in the upper-right corner to check for updates.
12. Copy and save the value of the endpoint in a text file, ommiting **:3306**.
    Your endpoint should look similar to the following example:
    lab5db.cnrczgvaxtw8.us-west-2.rds.amazonaws.com

Task 4: Interact with Your Database

In this task, you interact with your database through a PHP web application that was deployed to the web server in Lab 1. You will open a web application running on your web server.

1. [38] On the **Services** menu, click **EC2**.
2. In the navigation pane, click **Instances**.
3. Select **Web Server 1**, ensure that all other instances are cleared, and view the **Description** tab in the lower pane.

4. Copy the **IPv4 Public IP** address of **Web Server 1** that appears in the lower pane.
5. Paste the IP address in a new browser tab or window.
   A web application is displayed with the web server’s instance metadata.
6. Click the **RDS** link.
7. Configure the following settings (and ignore any settings that aren’t listed):
   - **Endpoint**: Paste the endpoint you copied previously.
     Omit the **:3306** at the end.
   - **Database**: type `sampleDB`
   - **Username**: type `labuser`
   - **Password**: type `labpassword`
     Ignore any message saying the connection
     is not secure. This is expected.
8. Click **Submit**.
   The connection string is displayed and then the page is redirected.
   (Alternately click on the **here** link to force redirection)
   Two new records are added to the address table and displayed.
9. Test whether the PHP web application can communicate with the RDS DB database, by adding another contact.
   Click **Add Contact** and enter a **Name**, **Phone**, and **Email**, and then click **Submit Query**.
10. To edit a contact, click **Edit**, modify one of the fields, and then click**Submit Query**.
11. To remove a record, click **Remove**. You can now close this browser tab or window.

Lab Complete

Congratulations! You have successfully configured a relational data store for your website. Clean up your lab environment.