Linux Academy
Hands-on Lab

Create and Configure an RDS Database with Backup Solutions

Contents

Related Courses

AWS Certified Solution Architect - Associate

Related Videos

RDS Essentials

Need Help?

Linux Academy Community

... and you can always send in a support ticket on our website to talk to an instructor!

Lab Connection Information

- Labs may take up to five minutes to build

- Access to an AWS Console is provided on the Hands-on Lab page, along with your login credentials

- Ensure you are using the N. Virginia region

- Labs will automatically end once the alloted amount of time finishes

In this lab, we'll learn how to create an RDS MySQL database, configure a subnet group and security group, and set up a read replica to serve as a backup solution.

To get started, log in to the AWS console with the credentials provided on the Hands-On Lab page. Once we're logged in, we'll navigate to the **VPC Dashboard**.

Check VPC Configuration

From the menu on the left, select **Your VPCs**. There may be a few leftover VPCs from others who have used this lab environment, but we'll be looking for the one named *LinuxAcademy*.

Next, we'll click the **Subnets** section from the menu on the left. There should be two subnets associated with the *LinuxAcademy* VPC. In the bottom pane on this screen, we'll select the **Route Table** tab for each one and ensure that it contains a route to an internet gateway (the *Target* should begin with *igw-*).

Once we've checked our VPC's configuration, we're ready to start working with the RDS.

Configure RDS Subnets

Return to the **AWS Services Console** and select *RDS* under the **Database** section. Before we set up the database itself, we'll need to configure the subnets it can use, so we'll select **Subnet Groups** from the menu on the left.

We may see some leftover subnet groups in the list, but we'll create our own by clicking the **Create DB Subnet Group** button at the top of the page.

In the **Name** field, enter *MyLabSnGroup*. **We can use this for the** Description as well.

From the **VPC** dropdown menu, be sure to select the *LinuxAcademy* VPC that was created for this lab environment. Next, we need to add both subnets to the group. If we're not sure of their availability zones, we can select each availability zone from the dropdown until we find the subnet, then click **Add**.

Once both subnets have been added to the group, click **Create**.

Launch RDS Instance

Next, we'll navigate to **Instances** from the sidebar menu and click the **Launch DB Instance**. We'll see a list of database engines to choose from; for this lab, we'll be using MySQL.

On the next screen, we'll have the choice of a production or dev/test environment. We'll select the *Dev/Test* option for this lab, and click **Next Step**.

The next screen will allow us to set database details. To make things simple, we'll first check the box to *Only show options that are eligible for RDS free tier*. Select the *t2.micro* option from the **DB Instance Class** menu.

Under the **Settings** section below, we'll enter *MyLabDBInstance* as the **DB Instance Identifier**. For the **Master Username** and **Master Password**, enter values that are secure and memorable. We will need these later, so it might be a good idea to write them down. Once we're done here, we'll click **Next Step**.

On the **Configure Advanced Settings** screen, we'll start with the Network & Security **section. From the** VPC menu, select the *LinuxAcademy* VPC we looked at earlier. We'll also select our subnet group *MyLabSnGroup* from the **Subnet Group** menu. For this lab, we'll set **Publicly Accessible** to *Yes*, and choose either of the availability zones. In the **VPC Security Group** setting, make sure that *Create new Security Group* is highlighted.

In the **Database Options** section, we'll first set our **Database Name**. This will be the name of the database itself, not the instance. For this lab, we'll call it *MyLabDatabase*. Make sure that the port is set to *3306*, and leave the rest of the settings at their default values.

We won't be configuring **Backup**, **Monitoring**, or **Maintenance** for this lab, but take a moment to look over the options available in these sections. Finally, click **Launch DB Instance**.

Check Network Rules

After our instance is created, we should see it on the **Instances** screen with a status of *available*.

When we select our database, we'll see an expanded panel with three tabs on the left side. If we click on the top tab, we're able to review its statistics and get its endpoint URL. Next, we can click the middle tab to see configuration details and security and network information. Under the **Security and Network** heading, click the link next to **Security Groups**.

A new tab will open, showing the RDS instance's security group.

Next, we'll go back to the **VPC Dashboard** and navigate to the **Network ACLs** screen. From this list, we should see an access control list with our two subnets associated. Select this ACL, and choose the **Inbound Rules** tab in the pane at the bottom of the page.

What we're looking for here is port *3306* to be open to inbound traffic. This will usually fall under a *Custom TCP Rule* with a range that includes 3306 and is marked *ALLOW*. When we see this, it means our database is accepting connections on the correct port.

We can also check the **Outbound Rules** to make sure the instance allows outbound traffic on port 3306. Once we confirm this, we can close the browser tab.

Create Snapshots

Back in the **RDS Dashboard**, we'll select **Snapshots** from the sidebar. We'll see that a snapshot has already been taken automatically, and we can also create manual snapshots. Let's explore this by clicking **Create DB Snapshot** at the top of the page.

Enter *myfirstsnap* as the **Snapshot Name**, then click **Create**. Next, we'll see another snapshot in our list, with a status of *creating*. If we select it, we'll see that the **Snapshot Type** is *manual*.

Create a Read Replica

To create a read replica, let's head back to the **Instances** screen and select our RDS instance. At the top, we'll see a dropdown titled **Instance Actions**. From that menu, select *Create Read Replica*.

Next, we'll configure the replica instance. Make sure that the **Read Replica Source** is set to the RDS instance we created, and set the **DB Instance Identifier** to *MyReadReplicaInstance*.

Set **Publicly Accessible** to *Yes* and choose an availability zone. For this lab, it doesn't matter which one you choose since both options will match those that we set when creating our subnet group.

The **Database Port** should be *3306* by default, but it's a good idea for us to stop for a moment and check to make sure it's correct. The rest of the settings on this page will be left in their default states.

At this point, if we click **Create Read Replica**, we may encounter an error: *DB instance is not in the available state...* This will happen if we finish configuring the read replica instance before our manual snapshot has completed. Many backup operations are not available to be run concurrently, and it's good to keep this in mind when setting automatic snapshots or creating new read replicas in the future.

If this error appears, wait for the backup to complete. Once the RDS instance's **Status** shows as *available*, we can select *Create Read Replica* from the **Instance Actions** menu and set it up again. When we're done, we'll click **Create Read Replica**.

Return to the **Instances** screen, and we'll see a new read replica instance being created.

Promote a Replica

Once both the original and replica RDS instances have a **Status** of *available*, we can proceed.

If we scroll right on the **Instances** page, we'll see the **Replication Role** field. The original, or primary, instance will have a role of *master* and the replica will be *replica*.

Select the replica instance, click the **Instance Actions** dropdown menu, and choose *Promote Read Replica*.

On the next screen, we'll see some configuration options. We'll select *Yes* for **Enable Automatic Backups**, since the replica will become a primary and it's a good idea to keep backups of our primary data sources. We'll also increase the **Backup Retention Period** to *7* days so it matches that of our existing primary. When we're done, we'll click **Continue**.

After reading the information on the next screen, click **Promote Read Replica**. The promotion will take a few minutes.

Delete the Primary Instance

Now that our replica has been promoted, we may no longer need the original primary instance. To delete it, we can select it on the **Instances** page, and choose *Delete* from the **Instance Actions** menu.

A warning will appear, telling us that _automated_ snapshots will be deleted along with the instance itself. However, _manual_ snapshots will not. This difference can be extremely important when deleting production databases.

Once we've read the warning and made sure we understand the consequences, we can click the **Delete** button.

Create an Instance from a Snapshot

Next, we'll choose **Snapshots** from the sidebar to see that our automated snapshots have been deleted, but our manual snapshot remains. If we want to use it to create a new instance, we can select our manual snapshot, click the **Snapshot Actions** menu at the top of the page, and select *Restore Snapshot*.

This will take us to another configuration screen like the ones we've seen before. From here, we can fill out the fields and launch the instance, but we don't need to do so to complete this lab.

Review

In this lab, we created a subnet group, created and configured an RDS instance, launched a read replica, promoted the replica to a primary instance, deleted the original primary, and retained snapshots--just in case! With this information, you now have a solid basis for understanding RDS database configuration and management.

Congratulations! You've completed the lab on creating and configuring an RDS database with backup solutions.