Databases

RDS - Relational Database Service

AWS RDS
Managed Database Service

• RDS (Relational Database Server) is Amazon's SQL offering
• Of course, you can always run your own database server, on an instance anywhere...
  • https://www.mysql.com/
• But why go to all that trouble?

AWS RDS
Managed Database Service

• Search for RDS in the top search bar.
• Click on RDS in the Services results.
From the dashboard, you may see a "Create Database" button in an announcement.
If not, click on "Databases" in the left sidebar.
Click "Create Database".

Select "Standard Create." We have to turn off some features which aren't allowed in AWS Academy.
Select MySQL
Leave the Edition and version as default.
MySQL Community
Version 8.0.x

Make sure to select "Free tier"
AWS RDS Managed Database Service

- Set a name for your DB Instance. This only shows up in the AWS console, it's not used for connecting to the database.
- Choose a good password, and keep it somewhere safe and memorable. If you forget it, you can reset this later.
- Change the instance class to db.t3.micro
- Change the Allocated storage to the smallest allowed: 20 GiB
- Un-check “Enable storage autoscaling”
- We won’t use anywhere near that much space.

AWS RDS Managed Database Service

- Don’t connect to an EC2 resource (we’ll configure that ourselves)
- Make sure you have “No” selected for Public access.
- Create a new security group
- Name your security group “database-sg”
- If you delete this RDS instance and create a new one later, you can re-use this VPC security group
AWS RDS Managed Database Service

- Leave "Password authentication" selected
- Expand "Additional configuration"
- Disable automated backups
  - Automated backups are usually the correct default for things, but we really want to minimize costs for the class, and daily backups really add up!
- Disable encryption
  - Usually a good idea, keep it simple for class.
- Click "Create database"
AWS RDS Managed Database Service

- Your database may take several minutes to be ready for use. The cloud is not instant

AWS RDS Managed Database Service

- Dark Patterns:
  - AWS now tries to up-sell you when creating things
  - Can only “hide” for 30 days!
  - You can avoid this by deploying resources through automation

AWS RDS Managed Database Service

- Eventually your RDS instance will complete. This may take 5 minutes or more.
- Click on your database name to get details on it.
You will need to copy down the Endpoint domain name. This is how you will connect to your database from a server.

Our new RDS instance does NOT have a public IP address
- Because RDS is a fully managed service, you cannot ssh into it
- With no Public IP you cannot connect to it directly from your laptop
- Our new RDS instance has a private IP address, and is listening on port 3306
- We need to give our EC2 instance access

With no public access, we must allow our EC2 instance access to the RDS instance
- This is done by allowing the security group attached to the RDS instance
- This is a very common pattern for cloud applications

We'll do this by adding the security group ID attached to our EC2 instance to the ingress rules of the RDS security group.

We want to add a rule that says, “Connections to the DB [port 3306] are allowed from the EC2 Security Group.”

In the EC2 console, select “Security Groups” from the left sidebar.

• the launch-wizard-1 SG is the one attached to our EC2 instance

• The database-sg SG is attached to our database
**AWS RDS Security Groups**

- We need to update the database security group, so select that one
- Then click “Edit inbound rules”

**AWS RDS Security Groups**

- Add a new rule
- Select MySQL/Aurora for the rule type
- For the source, click in the input field, and scroll down until you find the “launch-wizard-1” security group
- Click “Save rules”

**ECS ➝ RDS Connecting at last**

- Connect to your EC2 instance using your method of choice
- We need the mysql client software
- Docker!
- Use the hostname for your RDS instance, and the password you wrote down for the admin user (you did write down the password, right?)

```
sudo docker run -it --rm mysql:latest mysql -h class-db...rds.amazonaws.com -u admin -p
```
• Docker lets us run programs without installing them permanently

• Now we can use the MySQLdb module within python on our EC2 instance.
ECS ➡ RDS
Connecting from python

AWS RDS
Cleaning Up
- RDS instances are NOT automatically stopped when your AWS Academy lab session ends
- You will keep getting charged as long as it is active
- You can temporarily stop an RDS instance though

AWS RDS
Cleaning Up
- Can stop an RDS instance for up to 7 days
- After that it will automatically restart so AWS can keep it patched
- Still have to pay for the storage
- If you are done with an RDS instance, terminate it instead