Steps to Enable IAM DB Authentication on RDS Instance

1. Activate IAM DB authentication on the RDS DB instance.

* Select RDS instance and click on Modify
* Enable IAM DB authentication on the RDS DB instance

1. **Create an IAM Policy:** Create an IAM policy that allows your IAM user to access the RDS instance. The policy should include permissions for the **rds-db:connect** action for your RDS instance.

{

{

   "Version": "2012-10-17",

   "Statement": [

      {

         "Effect": "Allow",

         "Action": [

             "rds-db:connect"

         ],

         "Resource": [

             "arn:aws:rds-db:us-east-2:{Account\_No}:dbuser:{RDS\_Resource ID} /{DB User Name} "

         ]

      }

   ]

}

1. **Attach the IAM Policy to the IAM User:** Attach the IAM policy to your IAM user. This allows the IAM user to authenticate to the RDS instance using IAM database authentication.
2. **Generate an Authentication Token:** When connecting to the RDS instance, you will need to generate an authentication token using the AWS CLI or SDK. The token serves as the password for IAM database authentication.

Here's how you can generate an authentication token using the AWS CLI:

* **AWS CLI Command:** Use the generate-db-auth-token command to generate the authentication token. You'll need to provide the hostname of your RDS instance, the port number (usually 3306 for MySQL or 5432 for PostgreSQL), the AWS region, and the username of the IAM user or role you want to authenticate as.
* **Command:** If you're using MySQL and your RDS instance endpoint is mydbinstance.abcdefg.us-west-2.rds.amazonaws.com, and your IAM user or role ARN is arn:aws:iam::123456789012:user/myuser, the command would look like this:

aws rds generate-db-auth-token --hostname {endpoint}--port {db-port}--region {region}--username arn:aws:iam::{account\_id}:user/{db-user}

Example:

aws rds generate-db-auth-token --hostname disb-prod.cyt0cytg4zsl.ap-southeast-1.rds.amazonaws.com --port 3306 --region ap-southeast-1a -–username raqim-disbursement

* **Usage in Connection String:** Once you have the authentication token, you'll use it as the password in your connection string when connecting to the database. The IAM user or role ARN serves as the username.

mysql -h {end-point}-P 3306 -u arn:aws:iam::{account}:user/{raqim-disbursement} -p<authentication\_token>

Replace <authentication\_token> with the actual token generated by the generate-db-auth-token command.

This way, the authentication token acts as a temporary password that can be used to authenticate the IAM user or role to the RDS database with IAM database authentication enabled.

1. Download the AWS RDS Certificate pem file,

mkdir -p /var/mysql-certs/

cd /var/mysql-certs/

curl -O https://s3.amazonaws.com/rds-downloads/rds-combined-ca-bundle.pem

1. Connect to the RDS Instance: Use the authentication token along with the database username to connect to the RDS instance. The connection string should include the IAM user ARN as the username and the authentication token as the password.

Here's an example of how you might connect using the AWS CLI:

export AWS\_REGION=your\_region

export RDS\_INSTANCE\_ENDPOINT=your\_rds\_instance\_endpoint

export IAM\_USER\_ARN=your\_iam\_user\_arn

TOKEN=$(aws rds generate-db-auth-token --hostname $RDS\_INSTANCE\_ENDPOINT --port 3306 --region $AWS\_REGION --username your\_db\_username)

mysql -h $RDS\_INSTANCE\_ENDPOINT -P 3306 -u $IAM\_USER\_ARN -p$TOKEN

Replace **your\_region**, **your\_access\_key**, **your\_secret\_key**, **your\_rds\_instance\_endpoint**, **your\_iam\_user\_arn**, and **your\_db\_username** with your actual values.

Please note that IAM database authentication is only available for MySQL and PostgreSQL RDS instances.