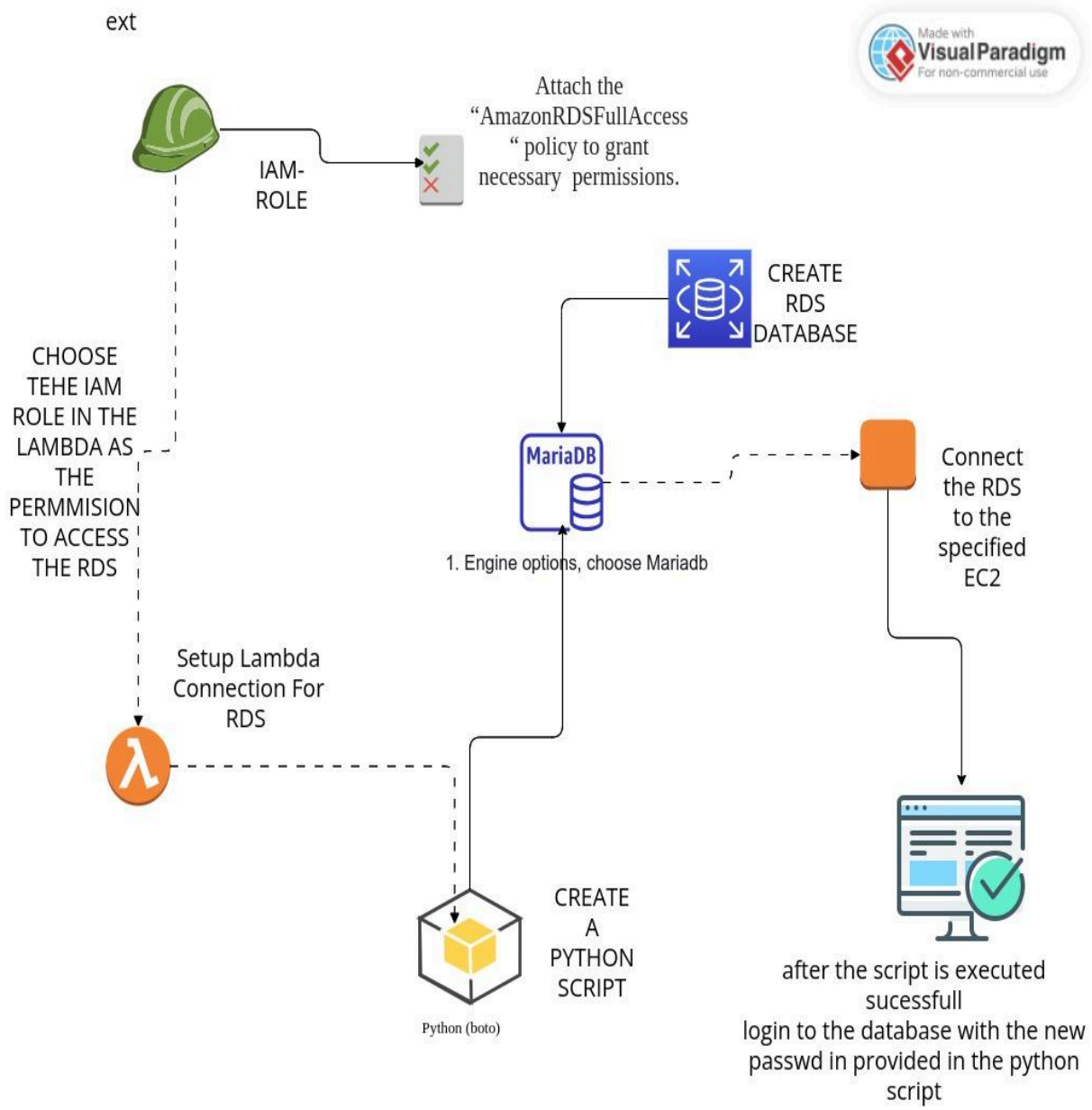
Using a Lambda function to change an Amazon RDS passwd



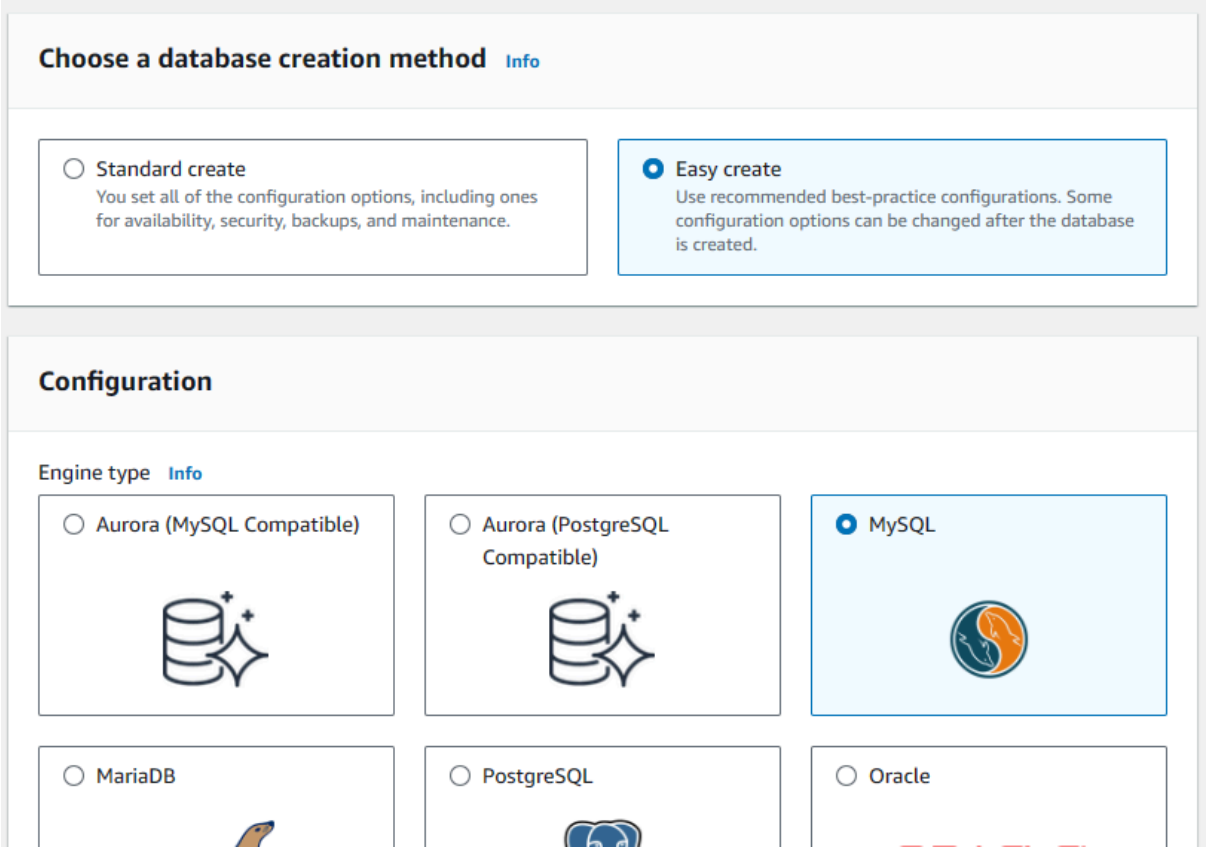
1] CREATE IAM ROLE

Create an IAM role Attach the "AmazonRDSFullAccess " policy to grant necessary permissions.

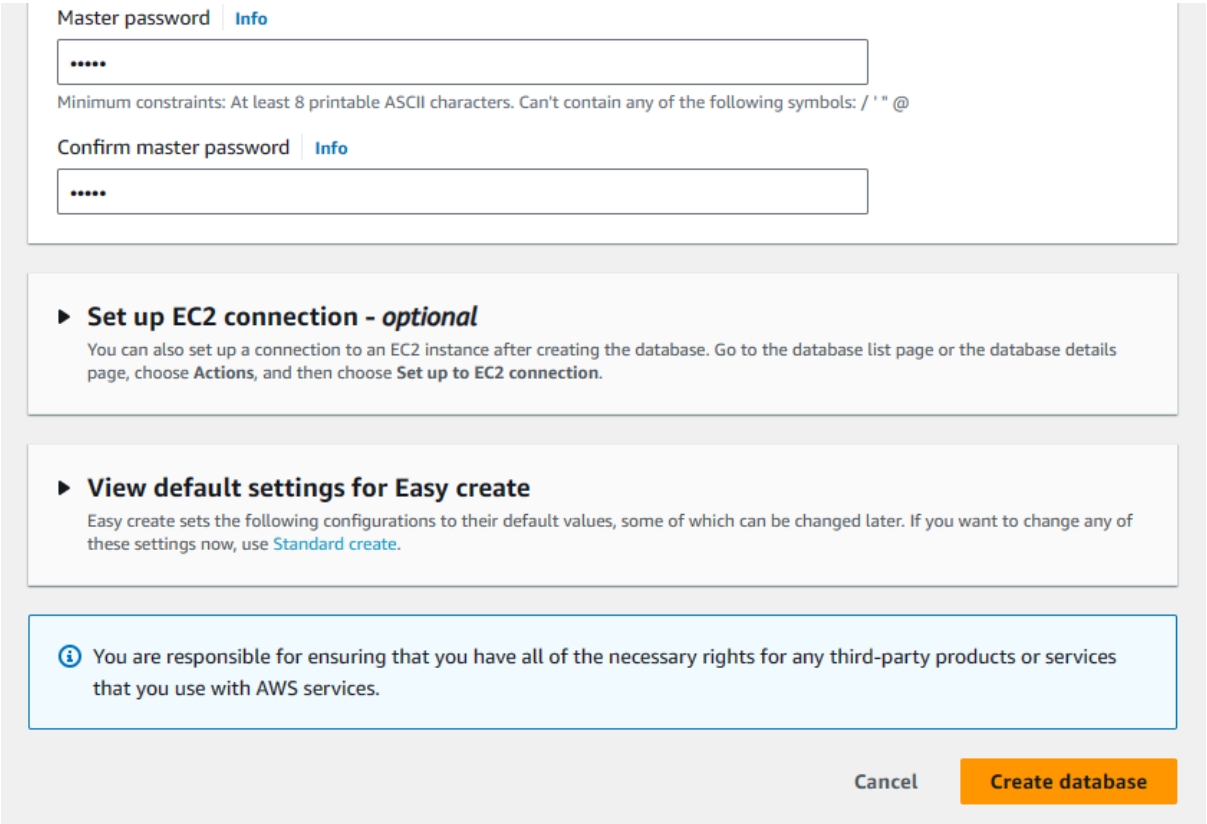
2]CREATE RDS DATABASE

To create a database instance

1. Open the Amazon RDS console and choose Create database.



2. Leave the Standard create option selected, then in Engine options, choose Mariadb
3. In Templates, choose Free tier.
4. In Settings, for DB instance identifier, enter “your identifier name”.
5. Set your username and password by doing the following:



- In Credentials settings, leave Master username set to `admin`.
- For Master password, enter and confirm a password to access your database.

6. Connect the RDS to the specified EC2
7. Leave all the remaining default options selected and choose Create

---

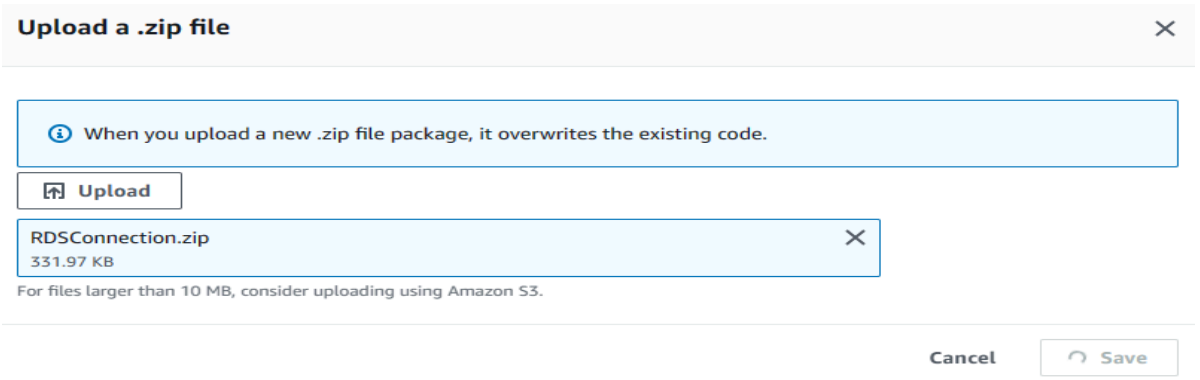
Setup Lambda Connection For RDS

Create the function

1] On this page select the execution environment as **Python**

- Install **botos3 and required** package for the code to accessing DB.

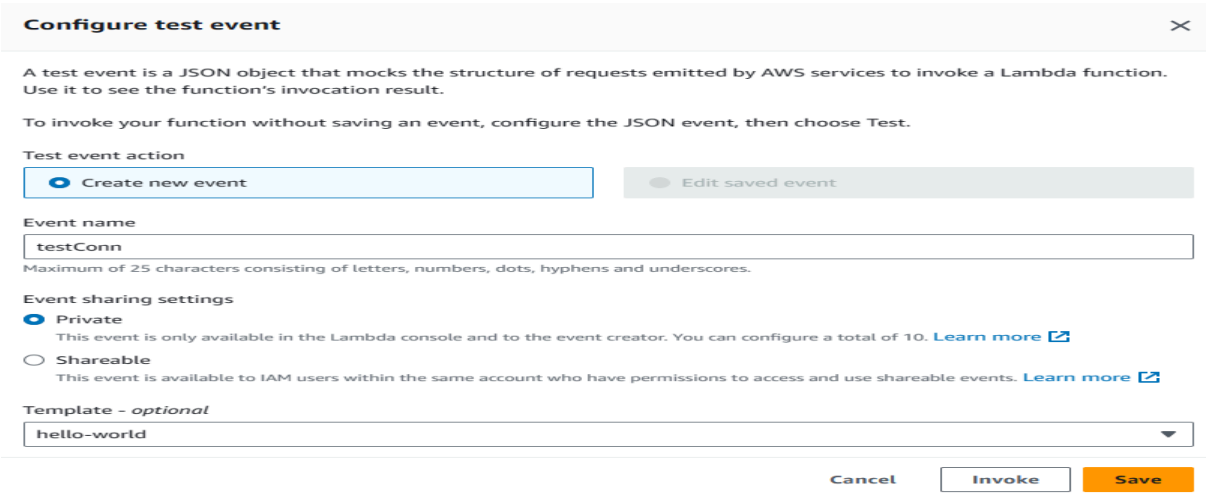
- Make it as a zip file and upload the zip to the lambda function



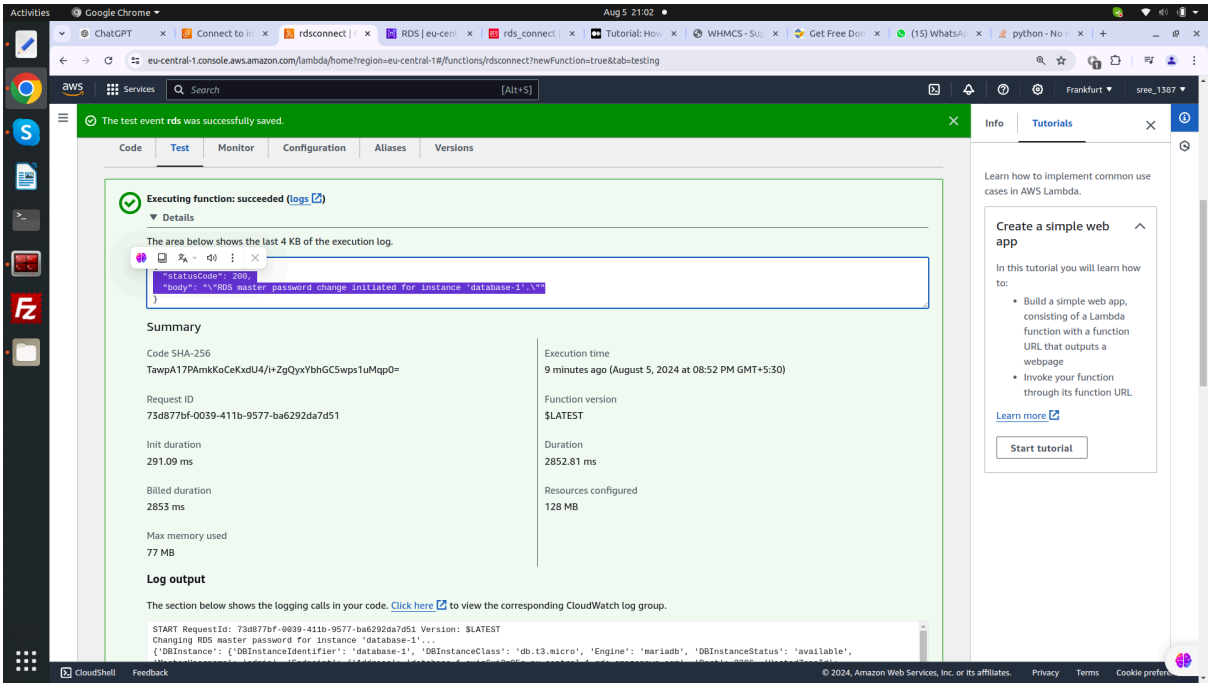
-

- **Test The Function For Access**
- Click on Test from code editor of lambda function. It will open event creation page.

Give any name to the event and leave everything as default and proceed.



- Once the event is created click on Test.



Python script

———————————

```
import boto3
import json

def change_rds_password(db_instance_identifier, new_password):
    # Create an RDS client using the IAM role credentials
    rds_client = boto3.client('rds')

    try:
        # Modify the RDS instance to update the master user password
        response = rds_client.modify_db_instance(
            DBInstanceIdentifier=db_instance_identifier,
            MasterUserPassword=new_password,
            ApplyImmediately=True  # Apply the changes immediately
```

```
        )

        print(f"Changing RDS master password for instance
'{db_instance_identifier}'...")
        print(response)
        return {
            'statusCode': 200,
            'body': json.dumps(f"RDS master password change initiated for
instance '{db_instance_identifier}'.")
        }
    except Exception as e:
        print(f"Error changing RDS master password: {e}")
        return {
            'statusCode': 500,
            'body': json.dumps(f"Error changing RDS master password:
{str(e)}")
        }

def lambda_handler(event, context):
    # Define your RDS instance details
    db_instance_identifier = 'database-1'
    new_password = 'admin1234'

    return change_rds_password(db_instance_identifier, new_password)

if __name__ == "__main__":
    # For local testing, call the lambda_handler function with dummy
event and context
    lambda_handler({}, {})
```