# ETL Pipeline between Amazon S3 and Amazon Redshift using AWS Lambda Function

## 

Github repository : <https://github.com/ravi0dubey/ETL_Pipeline_from_S3_to-_Redshift_using_AWS_Lambda/tree/main>

## Step 1: Activities to be done with S3 bucket

1. Create a bucket "etl-s3-lambda-redshift" .

## Step 2: Create IAM User

1. Create IAM user and download its access and secret key which will be used in Lambda configuration.

## Step 3: Setup IAM Role

* + 1. Create IAM role which has following permission.
       1. Administrator Access
       2. Amazon EC2 Full Access

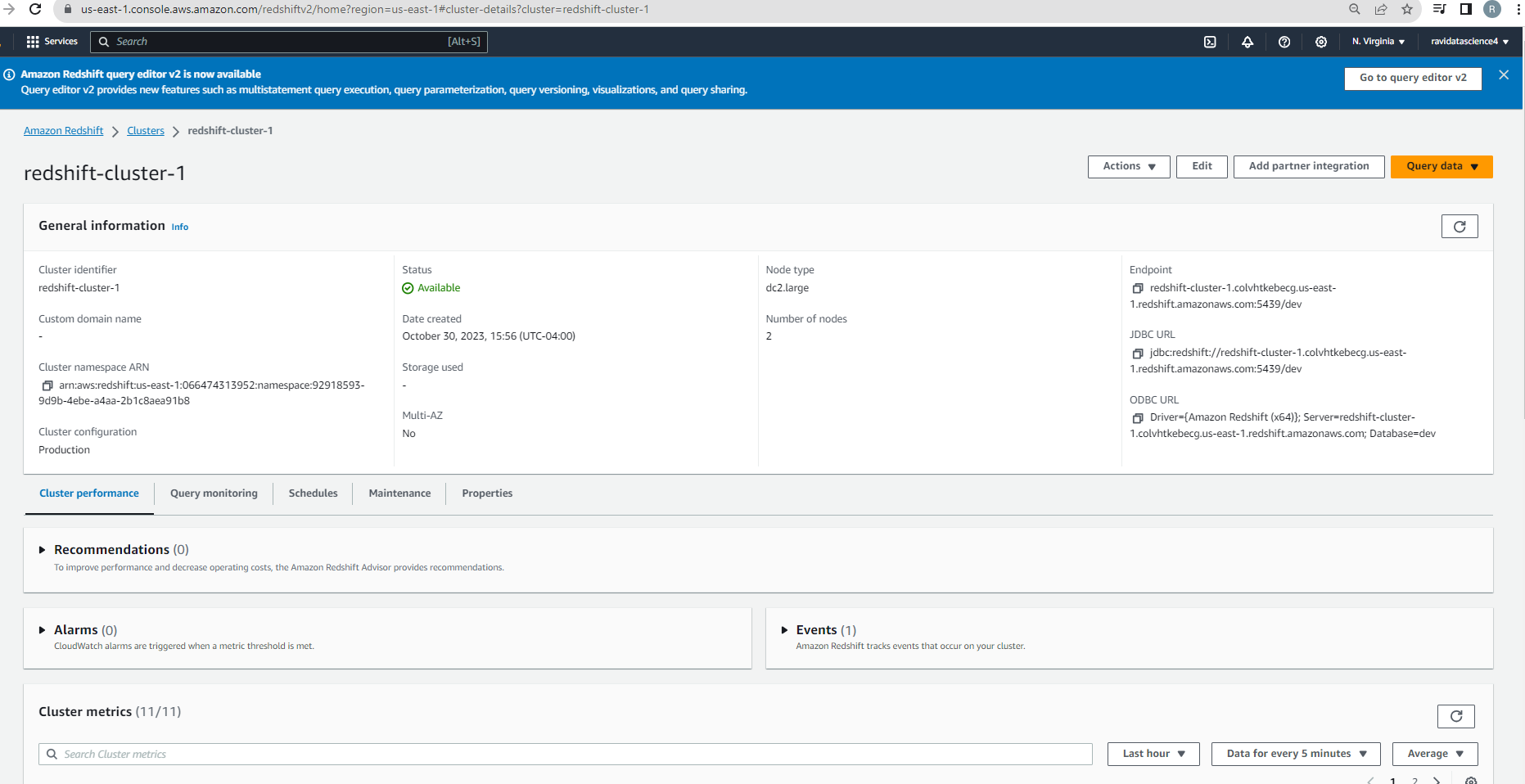
A screenshot of a computer

Description automatically generated

## Step 4 : Configuration to be done in Redshift

1. Create cluster in Amazon Redshift “redshift-cluster-1”
   1. Admin Username : “awsuser” (default)
   2. Manually add Admin user password : “ ”(give password for the redshift cluster)
   3. We can generate a new role or associate an existing role : “[AWSServiceRoleForRedshift](https://console.aws.amazon.com/iam/home?region=us-east-2" \l "/roles/AWSServiceRoleForRedshift" \t "_blank)”

It will take around 5 minutes to get started



* 1. Verify the VPC created by default in the cluster. Go to Properties -> Network and Security Settings

A screenshot of a computer

Description automatically generated

* + 1. Copy VPC and VPC security . It will be used in AWS Lambda settings.
    2. Click on the VPC security group -> Edit Inbound rules to allow all ports

A screenshot of a computer

Description automatically generated

* + 1. Verify the Subnet groups. It will be used in Lambda function

A screenshot of a computer

Description automatically generated

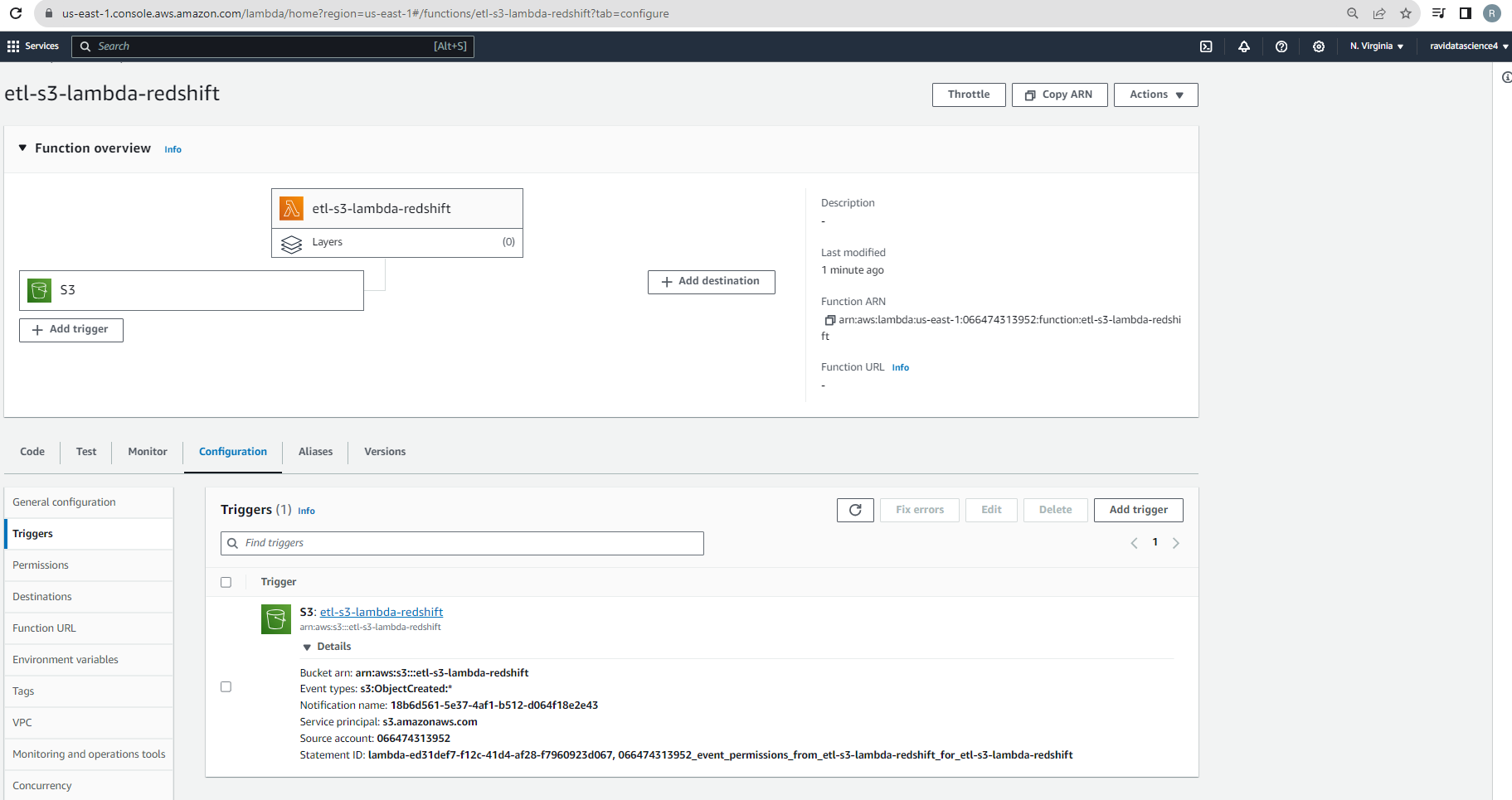
* 1. Go to query editor of redshift cluster and create table “Advertising” which we want to populate by placing file into S3 bucket. Create Table code can be found in GitHub repo “”

A screenshot of a computer

Description automatically generated

## Step 5: Activities to be done in Lambda Function

1. Create a lambda function "etl-s3-redshift",
   1. Runtime = python 3.7
   2. Execution role = new role
2. Add a Trigger in lambda function and select S3 bucket created in step 1.



1. Copy the python code in lambda function and deploy the changes. Code can be found in github repository.
2. Verify Lambda Function is setup correctly by checking following
   1. Go to the Configuration and check each element of it.
      1. **General Configuration** : Change the Timeout
         1. Timeout : 5 min
         2. Memory : 128 MB

A screenshot of a computer

Description automatically generated

* + 1. **Triggers** : Trigger created in Step 4.2 will get displayed.
    2. Edit **VPC**: Go to VPC -> Edit -> select the VPC which was assigned by default in Redshift cluster.

A screenshot of a computer

Description automatically generated

* + 1. Create **Layer** to be attached to Lambda function
       1. Go to Code -> Layers -> Create Layers -> enter details as mentioned below -> upload python.zip file kept at GitHub repo and select runtime as Python 3.7 and then create the layer.

A screenshot of a computer

Description automatically generated

* 1. **Attach** the layer to the lambda function
     1. Go to Code -> Layers -> Add Layers
     2. Choose custom layers “psycopg2” created above and click on add layer.

A screenshot of a computer

Description automatically generated

* 1. Set Environment variables. Go to Configuration -> Environment variables.

A screenshot of a computer

Description automatically generated

## Step 6: Test the ETL pipeline

1. Upload Advertising1.csv in S3 bucketA screenshot of a computer

   Description automatically generated
2. Monitor the logs in Cloud watch

A screenshot of a computer

Description automatically generated

1. Query the table in Redshift. A screenshot of a computer

   Description automatically generated