

## Building an ETL Pipeline in AWS

### 0.0 Description and Objective

We connected to an external data source for this project, loading it into a MySQL instance and then visualizing the data through the Tableau desktop version. The data set we used was the 2019 IRS submissions from the IRS 990 database.

Here are the steps to follow:

- Set-up the infrastructure and create the database
- Load the data in S3 and test the Glue Crawler
- Create the connections and AWS ETL job using Glue
- Create the visualization

The background of this dataset can be found in the following documentation:

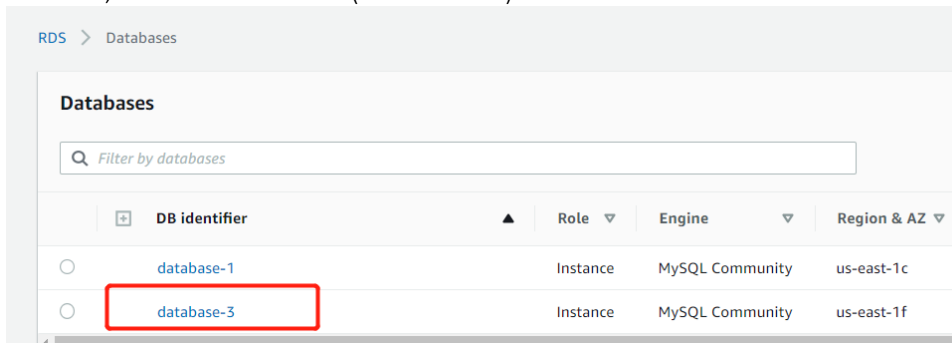
- <https://docs.opendata.aws/irs-990/readme.html>
- <https://aws.amazon.com/opendata/public-datasets/>

Objective:

Generated a CSV file from public data source (IRS 990) data and load it into a database that we created (S3 bucket), which was propagated to a MySQL database for connection to Tableau.

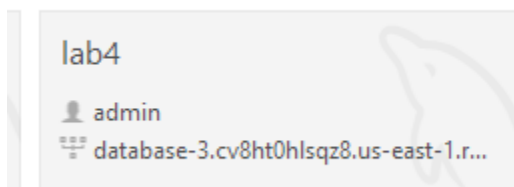
### 1.0 Set up the Infrastructure and Create the Database

1.1 First, created a database ('database-3') in Amazon RDS



1.2

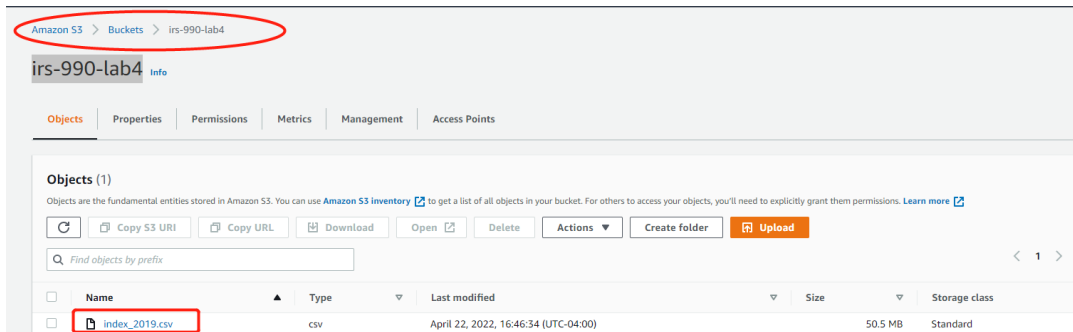
1.3 Once created, test the connection using our database credentials and connection string from MySQL workbench<sup>1</sup>



<sup>1</sup> For the RDS MySQL configuration, only MySQL v5 is compatible with [AWS Glue](#).

## 2.0 Load the data in S3 bucket

Now that we have set up the database connection, we created an S3 bucket called 'irs-990-lab4'. Then, Configure the bucket so that it is publicly accessible.



## 2.0 Create the connections and AWS ETL job using Glue

2.1 Now that we have created our S3 bucket, we set up an AWS Glue Crawler and connected it to the AWS database we created in MySQL. This Crawler runs a check on the S3 bucket we made and extracts and loads the file from IRS into your MySQL instance.

2.1.1 Note: we created a Glue Role with the following privileges:

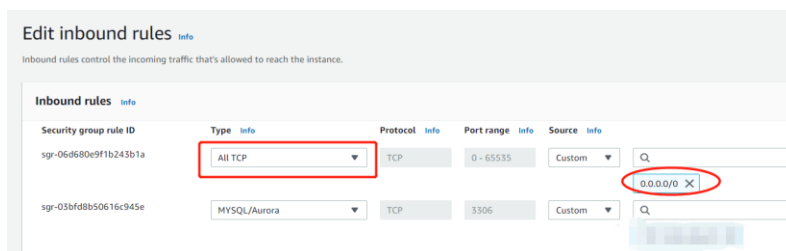
AmazonRDSFullAccess  
AmazonRDSDirectoryServiceAccess  
AWSGlueServiceRole  
AWSGlueServiceNotebookRole  
AdministratorAccess  
AmazonRDSDataFullAccess  
AWSGlueConsoleFullAccess  
AWSGlueConsoleSageMakerNotebookFullAccess  
AmazonS3FullAccess

2.1.2 Before connecting to mysql instance, the inbound rules of security should be updated. If we do not add all tcp type and choose 'anywhere, 0.0.0.0/0' it will cause below error.



For avoiding this error, we need follow below instructions.

RDS → database-3 → VPC security group → inbound rules tab → Edit inbound rules →





Also , we need to add endpoint to avoid the other error .

**irs-990 failed.** VPC S3 endpoint validation failed for SubnetId: subnet-0684af38d4ed28b87. VPC: vpc-020ac7a06286f2ae2. Reason: Could not find S3 endpoint or NAT gateway for subnetId: subnet-0684af38d4ed28b87 in Vpc vpc-020ac7a06286f2ae2 .

[Add connection](#) [Test connection](#) [Action](#) Showing: 1 - 1

<input checked="" type="checkbox"/> Name	Type	Date created	Last updated	Updated by
<input checked="" type="checkbox"/> irs-990	JDBC	21 April 2022 8:02 AM UTC-4	21 April 2022 8:02 AM UTC-4	root

Add ENDPOINTS instructions:

VPC → endpoints → create endpoints →

### Create endpoint Info

There are three types of VPC endpoints – Interface endpoints, Gateway Load Balancer endpoints, and Gateway endpoints. Interface endpoints and Gateway Load Balancer endpoints are powered by AWS PrivateLink, and use an Elastic Network Interface (ENI) as an entry point for traffic destined to the service. Interface endpoints are typically accessed using the public or private DNS name associated with the service, while Gateway endpoints and Gateway Load Balancer endpoints serve as a target for a route in your route table for traffic destined for the service.

#### Endpoint settings

**Name tag - optional**  
Creates a tag with a key of 'Name' and a value that you specify.

123

**Service category**  
Select the service category.

☒ AWS services  
Services provided by Amazon

☐ PrivateLink-ready partner services  
Services with an AWS Service Ready designation

☐ AWS Marketplace services  
Services that you've purchased through AWS Marketplace

☐ Other endpoint services  
Find services shared with you by service name

#### Services (1/2)

Filter services

Service Name: com.amazonaws.us-east-1.s3

Service Name	Owner	Type
<input checked="" type="radio"/> com.amazonaws.us-east-1.s3	amazon	Gateway
<input type="radio"/> com.amazonaws.us-east-1.s3	amazon	Interface

#### Services (1/2)

Filter services

Service Name: com.amazonaws.us-east-1.s3

Service Name	Owner	Type
<input checked="" type="radio"/> com.amazonaws.us-east-1.s3	amazon	Gateway
<input type="radio"/> com.amazonaws.us-east-1.s3	amazon	Interface

**VPC**  
Select the VPC in which to create the endpoint

VPC  
The VPC in which to create your endpoint.

vpc-0d504ad3e8591fca0

**Choose VPC relevant to database-3**

**Route tables (1/1)** [Info](#)

Filter route tables

<input checked="" type="checkbox"/>	Name	Route Table ID	Main
<input checked="" type="checkbox"/>	-	rtb-080e605c13bdce450	Yes

**Endpoints (1)** [Info](#)

Filter endpoints

<input type="checkbox"/>	Name	VPC endpoint ID	VPC ID	Service name	Endpoint type	Status	Creation time
<input type="checkbox"/>	myendpoint-1	vpce-0a67091ac48e60fbb	vpc-0d504ad3e8591fca0	com.amazonaws.us-east-1.s3	Gateway	Available	Friday, April 22, 2022, 16:35:02 EDT

After we added endpoint and updated the security type, we successfully connected to database.

[Connections](#) > database-connect-to-irs-990

[Edit](#)

Type JDBC  
JDBC URL jdbc:mysql://database-3.cv8ht0hlsqz8.us-east-1.rds.amazonaws.com:3306/irs-990-2  
VPC Id vpc-0d504ad3e8591fca0  
Subnet subnet-0bd8e306a89e3cec1  
Security groups sg-0b5bcdbbdf7d533f3  
Require SSL connection false  
Description -  
Username admin  
Created 20 April 2022 11:06 PM UTC-4  
Last modified 20 April 2022 11:06 PM UTC-4

Then, Create a Data Store called 'irs-990' and link it to our database using the jdbc connection string

**Location** jdbc:mysql://database-3.cv8ht0hlsqz8.us-east-1.rds.amazonaws.com:3306/irs-990-2

**Databases** A database is a set of associated tabl

[Add database](#)

[View tables](#)

[Action](#)

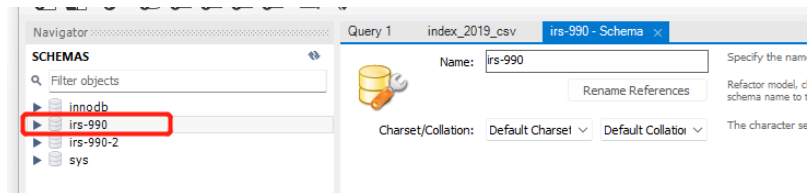
☐ **Name**

☐ irs-990

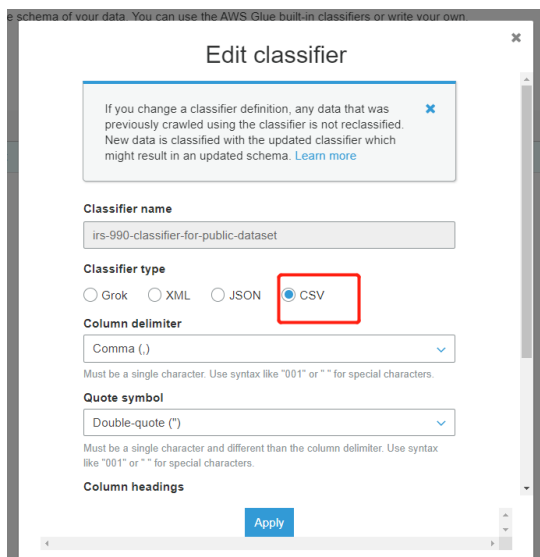


2.1 Create a Classifier to specify the structure of the file set being created and use the database name to create a schema in our own AWS RDS MySQL instance

Click right and create schema → update name that same as AWS database name



2.1.1 Name: *irs-990-classifier-for-public-dataset*



2.2 Create the Crawler under AWS Glue left-hand menu items. The Crawler you will create will crawl the public IRS 990 s3 bucket with the following path:

2.2.1 s3://irs-990-lab4/index\_2019.csv

2.2.2 Name: *irs-990-crawl-public-dataset*



Crawlers &gt; irs-990-crawl-public-dataset

Run crawler

Edit

Name	irs-990-crawl-public-dataset
Description	
Create a single schema for each S3 path	false
Security configuration	
Tags	-
State	Ready
Schedule	
Last updated	Fri Apr 22 16:30:04 GMT-400 2022
Date created	Fri Apr 22 16:30:04 GMT-400 2022
Database	irs-990
Table level	
Service role	glue-admin
Selected classifiers	irs-990-classifier-for-public-dataset
Data store	S3
Include path	s3://irs-990-lab4/index_2019.csv
Connection	
Exclude patterns	

## Configuration options

Schema updates in the data store	Update the table definition in the data catalog.
Object deletion in the data store	Mark the table as deprecated in the data catalog.

2.3 Once we have created the Crawler, run the Crawler to the specified location in our S3 bucket ('irs-990'). The Crawler pulls the file into S3 location. Note: this may take several minutes to execute.

2.4 Once we have crawled the irs-form-990 database, check to see that the database and table was successfully created by navigating to 'Tables' under the Glue service screen.

**Tables** A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

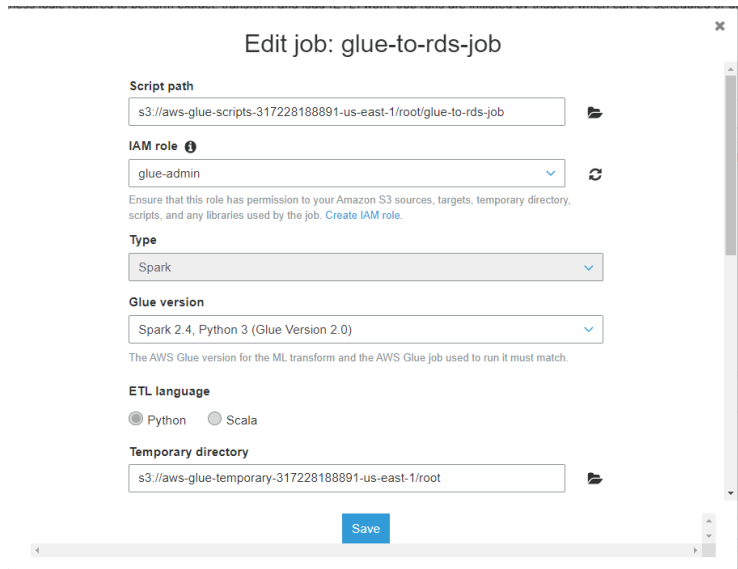
<a href="#">Add tables</a>	<a href="#">Action</a>	<input type="text" value="Filter by attributes or search by keyword"/>	<a href="#">Save view</a>	Showing: 1
<input type="checkbox"/> Name	Database	Location	Classification	Last updated
<input type="checkbox"/> index_2019_csv	irs-990	s3://irs-990-lab4/index_2019.csv	csv	22 April 2022 5:20 AM UTC-4

2.5 Once confirmed, now we can create a 'Job' to move the data from this location (your S3 bucket), to the MySQL database. we can use the Spark/Python Shell to autogenerate the script using the Glue interface.

The screenshot shows the AWS Glue console interface. On the left, the 'Data catalog' section is expanded, and 'Jobs (legacy)' is selected. The main area displays a table of jobs. A red circle highlights the job named 'glue-to-rds-job'. The table has columns for Name, Type, ETL language, Script location, Last modified, and Job bookmark. The job 'glue-to-rds-job' is of type 'Spark', uses 'python' as the ETL language, and was last modified on 23 April 2022 at 10:41 AM UTC-4.

2.6 Name: glue-to-rds-job

2.7

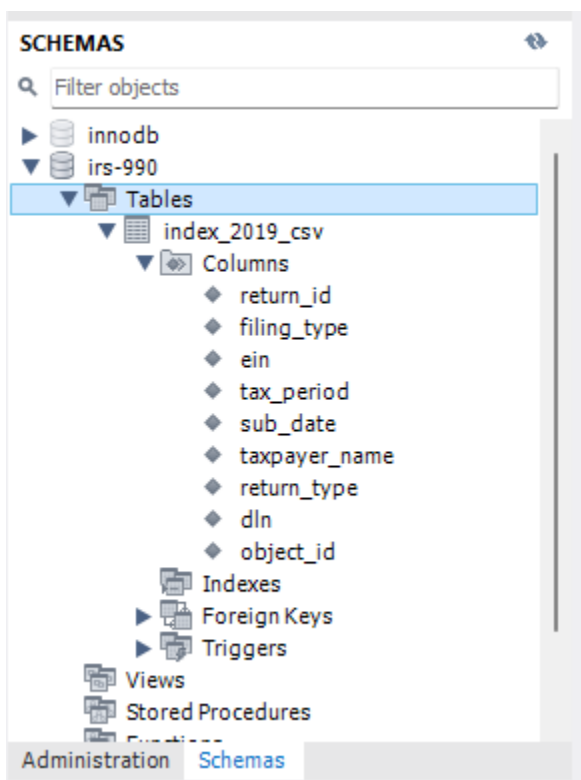


The screenshot shows the 'Edit job' configuration for 'glue-to-rds-job'. The fields are as follows:

- Script path:** s3://aws-glue-scripts-317228188891-us-east-1/root/glue-to-rds-job
- IAM role:** glue-admin
- Type:** Spark
- Glue version:** Spark 2.4, Python 3 (Glue Version 2.0)
- ETL language:** Python (selected), Scala
- Temporary directory:** s3://aws-glue-temporary-317228188891-us-east-1/root

A 'Save' button is located at the bottom right of the form.

Once we have created the Job, run it and check that the database was properly updated with the table. Our result looks like this:



Query 1 index\_2019\_csv x

Limit to 50000 rows

1 SELECT \* FROM 'irs-990'.index\_2019\_csv;

Result Grid

return_id	filing_type	ein	tax_period	sub_date	taxpayer_name	return_type	dn
16285381	EFILE	133065892	201809	5/10/2019 6:06:12 AM	LOGOS ENCOUNTER INC	990	9349309
16279505	EFILE	640411847	201805	5/8/2019 9:46:22 PM	MISSISSIPPI CHRISTIAN FOUNDATION	990	9349310
16279502	EFILE	870211329	201805	5/8/2019 9:46:20 PM	INTL SOC DAUGHTERS OF UT PIONEERS	990	9349310
16279501	EFILE	204223437	201806	5/8/2019 9:46:19 PM	SCHOOLHOUSE SUPPLIES INC	990	9349310
16279248	EFILE	475066819	201806	5/8/2019 9:13:09 PM	MINDFUL LIFE PROJECT	990	9349309
16279241	EFILE	541030357	201806	5/8/2019 9:13:04 PM	INTERNATIONAL TRUMPET GUILD	990	9349309
16279221	EFILE	942967138	201806	5/8/2019 9:12:49 PM	PTA GRATTAN ELEMENTARY	990	9349309
16283144	EFILE	20785117	201804	5/9/2019 2:06:47 PM	HOUSTON MARATHON FOUNDATION	990	9349305
16283142	EFILE	237410605	201806	5/9/2019 2:06:45 PM	VILLAGE SOUTH INSTITUTE OF HUMAN RESOU...	990	9349305
16283135	EFILE	943362724	201806	5/9/2019 2:06:38 PM	KIPP FOUNDATION	990	9349305
16279900	EFILE	221487121	201806	5/8/2019 9:46:18 PM	CATHOLIC FAMILY AND COMMUNITY SERVICES...	990	9349310

bx\_2019\_csv 1 x

Read Only

### 3.0 Create Connection and Visualization

Connect to our Tableau Desktop instance. We will connect to the IRS 990 database we created in our AWS Database Instances.

#### Connect Tableau Desktop to AWS RDS

To create a connection to our AWS RDS MySQL instance, we will follow a very similar process to connecting the MySQL

MySQL

General Initial SQL

Server  
database-3.cv8ht0hlsqz8.us-east-1.rds.amazonaws.com

Port  
3306

Database  
irs-990

Username  
[Redacted]

Password  
[Redacted]

☐ Require SSL

Sign In

Workbench.

#### Creating visualizations

