

# **Streaming Data Analytics with Amazon Kinesis Firehose and Redshift**

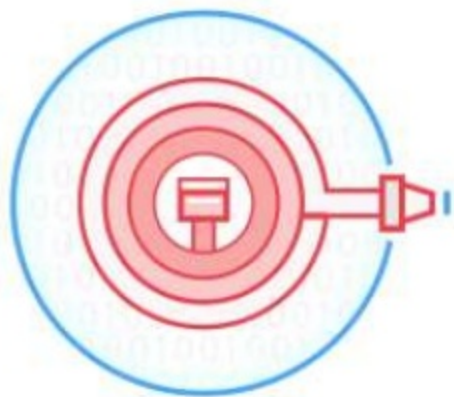
Ray Zhu, Sr. Product Manager, Amazon Kinesis

9/1/2016

# Agenda

- Kinesis Firehose and Redshift
- Stream Data to Redshift
  - Step 1 Set Up Redshift DB and Table
  - Step 2 Create Firehose Delivery Stream
  - Step 3 Send Data to Firehose Delivery Stream
  - Step 4 Query and Analyze the Data from Redshift
  - Step 5 Monitor Streaming Data Pipeline

# Kinesis Firehose



Load streaming data into Amazon S3,  
Amazon Redshift, and Amazon  
Elasticsearch Service

# Amazon Redshift



Petabyte-scale data warehouse

# **Stream Data to Redshift**

# Data Flow Overview



*Capture and submit streaming data to Firehose*

*Firehose loads streaming data continuously into Amazon S3, Redshift, or Elasticsearch Service*

*Analyze streaming data using your favorite analytical tools*

**Zero administration:** Capture and deliver streaming data into Amazon S3, Redshift, and Elasticsearch Service without writing an application or managing infrastructure.

**Direct-to-data store integration:** Batch, compress, and encrypt streaming data for delivery into data destinations in as little as 60 secs using simple configurations.

**Seamless elasticity:** Seamlessly scale to match data throughput without intervention.

# **Step 1 Set Up Redshift DB and Table**

# Cluster Details

AWS

Services

Edit

Ray Zhu · N. Virginia · Support

Redshift Dashboard

Clusters

Snapshots

Security

Parameter Groups

Reserved Nodes

Events

Connect Client

CLUSTER DETAILS

NODE CONFIGURATION

ADDITIONAL CONFIGURATION

VIEW

Provide the details of your cluster. Fields marked with \* are required.

Cluster Identifier\*

raystreaming

This is the unique key that identifies a cluster. This parameter is stored as a lowercase string (e.g. my-db-instance).

Database Name

streamingdb

Optional. A default database named dev is created for the cluster. Optionally, specify a custom database name (e.g. mydb) to create an additional database.

Database Port\*

5439

Port number on which the database accepts connections.

Master User Name\*

rayzhu

Name of master user for your cluster. (e.g. awsuser)

Master User Password\*

\*\*\*\*\*

Password must contain 8 to 64 printable ASCII characters excluding /, ", ', \, and @. It must contain 1 uppercase letter, 1 lowercase letter, and 1 number.

Confirm Password\*

\*\*\*\*\*

Confirm Master User Password.

Cancel

Continue



# Node Configuration

The screenshot shows the AWS Redshift console's 'Node Configuration' page. The top navigation bar includes the AWS logo, 'Services' dropdown, a set of service icons, an 'Edit' dropdown, and user information 'Ray Zhu', 'N. Virginia', and a 'Support' link. A left-hand sidebar contains links to 'Redshift Dashboard', 'Clusters', 'Snapshots', 'Security', 'Parameter Groups', 'Reserved Nodes', 'Events', and 'Connect Client'. The main content area features a progress bar with four steps: 'CLUSTER DETAILS', 'NODE CONFIGURATION' (the active step), 'ADDITIONAL CONFIGURATION', and 'REVIEW'. Below the progress bar, a note states: 'Choose a number of nodes and Node Type below. Number of Compute Nodes is required for multi-node clusters.' The configuration options are as follows: 'Node Type' is a dropdown menu set to 'dc1.large', with a description: 'Specifies the compute, memory, storage, and I/O capacity of the cluster's nodes.' Below this, the specifications for 'dc1.large' are listed: 'CPU: 7 EC2 Compute Units (2 virtual cores) per node', 'Memory: 15 GiB per node', 'Storage: 160GB SSD storage per node', and 'I/O Performance: Moderate'. 'Cluster Type' is a dropdown menu set to 'Single Node'. 'Number of Compute Nodes\*' is a text input field containing the value '1', with a description: 'Single Node clusters consist of a single node which performs both leader and compute functions.' Below the input field, the 'Maximum' and 'Minimum' values are both set to '1'. At the bottom of the page, there are three buttons: 'Cancel', 'Previous', and 'Continue'.

**CLUSTER DETAILS** **NODE CONFIGURATION** ADDITIONAL CONFIGURATION REVIEW

Choose a number of nodes and Node Type below. Number of Compute Nodes is required for multi-node clusters.

**Node Type**  Specifies the compute, memory, storage, and I/O capacity of the cluster's nodes.

**CPU** 7 EC2 Compute Units (2 virtual cores) per node

**Memory** 15 GiB per node

**Storage** 160GB SSD storage per node

**I/O Performance** Moderate

**Cluster Type**

**Number of Compute Nodes\***  Single Node clusters consist of a single node which performs both leader and compute functions.

**Maximum** 1

**Minimum** 1

# Additional Configuration

**Additional Configuration**

Provide the optional additional configuration details below.

Cluster Parameter Group: **default-*redshift-1.0*** Parameter group to associate with this cluster.

Encrypt Database: ☒ None ☐ KMS ☐ HSM [Learn more about database encryption](#)

Configure Networking Options

Choose a VPC: **Default VPC (vpc-2cb3148)** The identifier of the VPC in which you want to create your cluster.

Cluster Subnet Group: **default** Selected Cluster Subnet Group may limit the choice of Availability Zones.

**Publicly Accessible** ☒ Select Yes if you want the cluster to be accessible from the public internet. Select No if you want it to be accessible only from within your private VPC network.

Choose a Public IP Address: **No** Select Yes if you want to select your own public IP address from a list of elastic IP (EIP) addresses that are already configured for your cluster's VPC. Select No if you want Amazon Redshift to provide an EIP for you instead.

Availability Zone: **No Preference** The EC2 Availability Zone that the cluster will be created in.

Optionally, associate your cluster with one or more security groups.

VPC Security Groups: **default (sg-d37f3086)** List of VPC Security Groups to associate with this cluster.  
default\_elb\_6a9b664d-06f1-3a26-8f9b-bd3eedc2cbed (sg-KinesisDataVisSampleApp-Ec2SecurityGroup-87b86GKXGJNL)  
launch-wizard-1 (sg-eb7eb78d)

Optionally, create a basic alarm for this cluster.

Create CloudWatch Alarm: ☐ Yes ☒ No Create a CloudWatch alarm to monitor the disk usage of your cluster.

Optionally, associate up to 10 IAM roles with this cluster.

Available Roles: **Choose a role**

**Cancel** **Previous** **Continue**

# Review

Redshift Dashboard

Clusters

Snapshots

Security

Parameter Groups

Reserved Nodes

Events

Connect Client

AWS

Services

Tools

Help

Feedback

Ray Zhu

N. Virginia

Support

CLUSTER DETAILS

NODE CONFIGURATION

ADDITIONAL CONFIGURATION

REVIEW

You are about to launch a cluster with the following specifications:

Cluster Properties

These attributes specify the name of your cluster, what type of virtual hardware it will run on, how many nodes it will contain, and the availability zone in which it will be located.

Cluster Identifier: **rg-streaming**

Node Type: **dc1.large**

Number of Compute Nodes: **1** (leader and compute run on a single node)

Availability Zone: **us-east-1a**

Database Configuration

These properties specify the database name, port, and username you will use to connect to the database. The parameter group contains configuration values used by the database.

Database Name: **rg-streamingdb**

Database Port: **5439**

Master User Name: **rg-zhu**

Cluster Parameter Group: **default.redshift5-1.0**

Security, Access, and Encryption

These settings control whether your cluster will be created in an existing VPC to allow for simpler integration with other AWS Services, and the security groups which define access rules to your cluster.

Virtual Private Cloud: **vpc-3db3118d**

Cluster Subnet Group:

Publicly Accessible: **yes**

Elastic IP: **not used**

VPC Security Group: **sg-e377208d**

Encrypt Database: **no**

CloudWatch Alarms

CloudWatch alarms are used to notify if metrics for your cluster are within a certain threshold. All recipients under the SNS topic specified for your alarm will receive notifications once an alarm is triggered.

Basic alarms will not be created for this cluster.

⚠ Unless you are eligible for the free trial, you will start accruing charges as soon as your cluster is active.

Applicable charges:

The on-demand hourly rate for this cluster will be **\$0.25**, or **\$0.25/node**. If you have purchased reserved nodes in this region for this node type that are active, your costs will be discounted. Additional nodes will be billed at the on-demand rate.

If you are eligible for a free trial, you will receive 750 hours of free usage for each month of the trial, applied across all running dc1.large nodes across all regions. Regardless of when you start your trial, you will receive two full months of free usage. Once your trial expires or your usage exceeds 750 hours/month, you can shut down your cluster, avoiding any charges, or keep it running at our standard On-Demand Rate.

For more information, see [Amazon Redshift Free Trial FAQ](#), [Amazon Redshift Pricing](#), and [Reserved Nodes Documentation](#).

Cancel

Previous

Launch Cluster

# Configure VPC Security Group

The screenshot displays the AWS Redshift console interface for the 'raystreaming' cluster. The left sidebar contains navigation links: Redshift Dashboard, Clusters, Snapshots, Security, Parameter Groups, Reserved Nodes, Events, and Connect Client. The main content area shows the cluster's configuration, status, and database properties. The 'VPC Security Groups' field is highlighted with a red box, indicating the current security group configuration.

**Cluster:** raystreaming

**Endpoint:** raystreaming.cfyxscd70m.us-east-1.redshift.amazonaws.com:5439 (authorized)

**Cluster Properties**

|                            |  |
|----------------------------|--|
| Cluster Name               | raystreaming                             |
| Cluster Type               | Single Node                              |
| Node Type                  | dc1.large                                |
| Nodes                      | 1  |
| Zone                       | us-east-1b                               |
| Created Time               | August 3, 2016 at 11:23:23 AM UTC-7      |
| Cluster Version            | 1.0.1062                                 |
| VPC ID                     | vpc-2db31f68 (View VPCs)                 |
| Cluster Subnet Group       | default                                  |
| <b>VPC Security Groups</b> | <b>default (sg-d37d06d6)</b><br>(active) |
| Cluster Parameter Group    | default:redshift-1.0 (in-sync)           |

**Cluster Status**

|                              |           |
|------------------------------|-----------|
| Cluster Status               | available |
| Database Health              | healthy   |
| In Maintenance Mode          | no        |
| Parameter Group Apply Status | in-sync   |
| Pending Modified Values      | None      |

**Cluster Database Properties**

|                     |   |
|---------------------|---|
| Port                | 5439  |
| Publicly Accessible | Yes   |
| Database Name       | streamingdb   |
| Master Username     | raychu  |
| Encrypted           | No  |
| JDBC URL            | jdbc:redshift://raystreaming.cfyxscd70m.us-east-1.redshift.amazonaws.com:5439/streamingdb   |
| ODBC URL            | Driver={Amazon Redshift JDBC Driver};Server=raystreaming.cfyxscd70m.us-east-1.redshift.amazonaws.com;Database=streamingdb;UID=raychu;PWD={insert_your_master_username_here};Port=5439 |

**Backup, Audit Logging, and Maintenance**

|                                     |                 |
|-------------------------------------|-----------------|
| Automated Snapshot Retention Period | 1               |
| Cross-Region Snapshots Enabled      | No              |
| Audit Logging Enabled               | No              |
| Maintenance Window                  | Tue 08:30-09:00 |

# Configure VPC Security Group



US East (N. Virginia)  
52.70.63.192/27

US West (Oregon)  
52.89.255.224/27

EU (Ireland)  
52.19.239.192/27

The screenshot shows the AWS Management Console interface. In the left-hand navigation pane, the 'Security Groups' link is selected under the 'Network & Security' section. The main content area displays the 'Edit inbound rules' dialog for a security group named 'sg-437D3864'. The dialog has a table with the following columns: Type, Protocol, Port Range, and Source. The first row shows 'Redshift' as the Type, 'TCP' as the Protocol, '5438' as the Port Range, and 'Custom' as the Source. The 'Source' field is highlighted with a red rectangle, and the IP address '52.70.63.192/27' is visible in the dropdown menu. Below the table, there are 'Add Rule', 'Cancel', and 'Save' buttons. The background shows the 'Security Groups' page with a table of existing security groups.

| Type     | Protocol | Port Range | Source                 |
|----------|----------|------------|------------------------|
| Redshift | TCP      | 5438       | Custom 52.70.63.192/27 |

# Connect to Redshift DB

Connect to Redshift

Connect to Server

aginity

Need help?  
Just ask [support.aginity.com](https://support.aginity.com).

Connection Properties

New... Save Delete... Rename... Copy

Saved: RayStreamingRedshift

Server: st-1.redshift.amazonaws.com Ssl Mode: Prefer

User ID: rayzhu Password: \*\*\*\*\* ☒ Save password

Database: streamingdb

Port: 5439

Red Shift Native

OK Cancel

# Create a Redshift Table

```
create table TrafficViolation(  
  dateofstop date,  
  timeofstop timestamp,  
  agency varchar(100),  
  subagency varchar(100),  
  description varchar(300),  
  location varchar(100),  
  latitude varchar(100),  
  longitude varchar(100),  
  accident varchar(100),  
  belts varchar(100),  
  personalinjury varchar(100),  
  propertydamage varchar(100),  
  fatal varchar(100),  
  commlicense varchar(100),  
  hazmat varchar(100),  
  commvehicle varchar(100),  
  alcohol varchar(100),  
  workzone varchar(100),  
  state varchar(100),  
  veichletype varchar(100),  
  year varchar(100),  
  make varchar(100),  
  model varchar(100),  
  color varchar(100),  
  violation varchar(100),  
  type varchar(100),  
  charge varchar(100),  
  article varchar(100),  
  contributed varchar(100),  
  race varchar(100),  
  gender varchar(100),  
  drivercity varchar(100),  
  driverstate varchar(100),  
  distate varchar(100),  
  arresttype varchar(100),  
  geolocation varchar(100));
```

# Create a Redshift Table

The screenshot displays the Aginity Workbench for Redshift interface. The title bar indicates the connection to a Redshift cluster named 'raystreaming' in the 'us-east-1' region. The interface includes a menu bar (File, Edit, View, Query, Results, Object, Tools, Window, Help) and a toolbar with buttons for Connect, Execute, Explain, and Abort. The 'Object Browser' on the left shows a tree of databases, with 'rayshu.streamingdb (120 MB)' selected. The main query editor contains the SQL statement: `select * from TrafficViolation1`. Below the editor, the 'Output' tab shows 'Result 1' with a table of data. The table has columns: dateofstop, timeofstop, agency, subagency, description, location, latitude, longitude, accident, belts, and person. The status bar at the bottom shows 'Ready', 'Pos: 31 (row: 1, col: 32)', the user 'rayshu', and query statistics: '1 query, 2.136 sec., 0 row'.

Aginity Workbench for Redshift - [Query Analyser (raystreaming.chyau1d7lrm.us-east-1.redshift.amazonaws.com:5439 - Ray(streaming)redshift) - Untitled]

File Edit View Query Results Object Tools Window Help

Connect Execute Explain Abort Database: streamingdb Path: rayshu

Object Browser

- Databases (5)
  - rayshu.streamingdb (120 MB)
  - rdshb.dev
  - rdshb.paid\_harvest
  - rdshb.template0
  - rdshb.template1
- Security

Tab 1 x Tab 2 x New

```
select * from TrafficViolation1
```

Output Result 1 x

Drag a column header here to group by that column.

| dateofstop | timeofstop | agency | subagency | description | location | latitude | longitude | accident | belts | person |
|------------|------------|--------|-----------|-------------|----------|----------|-----------|----------|-------|--------|
|------------|------------|--------|-----------|-------------|----------|----------|-----------|----------|-------|--------|

Query Set

Ready Pos: 31 (row: 1, col: 32) rayshu 1 query, 2.136 sec., 0 row



## **Step 2 Set Up Firehose Delivery Stream**

# Destination

**Amazon Kinesis** | **Create Delivery Stream**

**Streams**  
**Firehose**  
Analytics

**Step 1: Destination**  
Step 2: Configuration  
Step 3: Review

### Destination

Select the destination where your streaming data will be delivered.

Destination: Amazon Redshift

Delivery stream name: myredshiftstream

### Intermediate S3 Bucket

Firehose delivers data to your S3 bucket first and then issues Redshift COPY command to load the data into your Redshift cluster.

S3 bucket: firehose-test-bucket

S3 prefix: s3 prefix

### Redshift Cluster

Redshift cluster: mydstreamg

Redshift database: dstreamingdb

Redshift table: TrafficLocation

Redshift table columns: Redshift table columns

Redshift username: mydhu

Redshift password: [masked]

Redshift COPY options: json 'auto'

Optional parameters for Redshift COPY command. [Learn more](#)

Retry duration (sec): 30

Retry duration can range from 0 seconds to 720 seconds in 1 second increments.

**COPY command**

```
COPY TrafficLocation FROM 's3://firehose-test-bucket/' MANIFEST 'CREDENTIALS'
'aws_access_key_id'='aws-access-key-id' 'aws_secret_access_key'='aws-secret-access-key'
MANIFEST json 'auto'.
```

[View instructions](#)

\*Required

Cancel Next

# Configuration

**Amazon Kinesis**

- Kinesis Streams
- Kinesis Firehose**

**Create Delivery Stream**

Step 1: Destination  
**Step 2: Configuration**  
Step 3: Review

### Configuration

Configure buffer, compression, logging and IAM role options for your delivery stream.

#### S3 Buffer

Firehose buffers incoming data before delivering to your S3 bucket. You can configure buffer size and buffer interval. The first satisfied condition will trigger the data delivery to your S3 bucket.

Buffer size\* 1  
Buffer size can range from 1MB to 128MB in 1MB increments.

Buffer interval\* 60  
Buffer interval can range from 50s to 900s in 1 second increments.

#### S3 Compression and Encryption

Firehose can compress and encrypt the data before delivering to your S3 bucket.

Data compression UNCOMPRESSED ⓘ

Data encryption No Encryption ⓘ

#### Error Logging

Firehose can log data delivery errors to CloudWatch Logs. If enabled, a CloudWatch Log Group and corresponding Log Stream(s) are created on your behalf. [Learn more](#).

☒ Enable ☐ Disable

#### IAM Role

Firehose needs an IAM role to access your specified resources, such as the S3 bucket and KMS key. [Learn more](#).

IAM role\* firehose\_delivery\_role ⓘ

\*Required

Cancel Previous **Next**

# Review

The screenshot shows the 'Review' step of the 'Create Delivery Stream' wizard in the AWS Management Console. The left sidebar shows 'Amazon Kinesis' with 'Streams' and 'Firehose' (selected) under 'Analytics'. The main area has a breadcrumb 'Create Delivery Stream' and three steps: 'Step 1: Destination', 'Step 2: Configuration', and 'Step 3: Review' (active). The 'Review' section contains two expandable sections: 'Destination' and 'Configuration'. The 'Destination' section lists: Destination (Amazon Redshift), Delivery stream name (rayshiftstream), S3 bucket (firehose-test-bucket), S3 prefix (none), Redshift cluster (raystreaming), Redshift database (streamingdb), Redshift table (TrafficVisitation), Redshift table columns (none), Redshift username (raychi), Redshift COPY options (json 'auto'), and Retry duration (sec) (30). The 'Configuration' section lists: S3 buffer size (1), S3 buffer interval (60), S3 compression (UNCOMPRESSED), S3 encryption (No Encryption), Error logging (Enabled), and IAM role (firehose\_delivery\_role). At the bottom are 'Cancel', 'Previous', and 'Create Delivery Stream' buttons.

**Amazon Kinesis** | **Create Delivery Stream**

Step 1: Destination  
Step 2: Configuration  
**Step 3: Review**

### Review

Review your destination and configuration before creating your delivery stream

**Destination**

|                        |                      |
|------------------------|----------------------|
| Destination            | Amazon Redshift      |
| Delivery stream name   | rayshiftstream       |
| S3 bucket              | firehose-test-bucket |
| S3 prefix              | none                 |
| Redshift cluster       | raystreaming         |
| Redshift database      | streamingdb          |
| Redshift table         | TrafficVisitation    |
| Redshift table columns | none                 |
| Redshift username      | raychi               |
| Redshift COPY options  | json 'auto'          |
| Retry duration (sec)   | 30                   |

**Configuration**

|                    |                        |
|--------------------|------------------------|
| S3 buffer size     | 1                      |
| S3 buffer interval | 60                     |
| S3 compression     | UNCOMPRESSED           |
| S3 encryption      | No Encryption          |
| Error logging      | Enabled                |
| IAM role           | firehose_delivery_role |

Cancel Previous **Create Delivery Stream**

## **Step 3 Send Data to Firehose Delivery Stream**

# Sample Data

US Government Open Data: <https://catalog.data.gov/dataset/traffic-violations-56dda>

09/30/2014,23:51:00,MCP,"1 st district, Rockville",DRIVER FAILURE TO STOP AT STEADY CIRCULAR RED SIGNAL,PARK RD AT HUNGERFORD DR,,,No,No,No,No,No,No,No,No,No,No,MD,02 -

Automobile,2014,FORD,MUSTANG,BLACK,Citation,21-202(h1),Transportation

Article,No,BLACK,M,ROCKVILLE,MD,MD,A - Marked Patrol,

03/31/2015,23:59:00,MCP,"2nd district, Bethesda",HEADLIGHTS (\*),CONNECTICUT AT METROPOLITAN AVE,,,No,No,No,No,No,No,No,No,No,No,MD,02 -

Automobile,2003,HONDA,2S,BLUE,ESERO,55\*,No,HISPANIC,M,SILVER SPRING,MD,MD,A - Marked Patrol,

09/30/2014,23:30:00,MCP,"5th district, Germantown",FAILURE TO DISPLAY TWO LIGHTED FRONT LAMPS WHEN REQUIRED,OBSERVATION @ RIDGE ROAD,,,No,No,No,No,No,No,No,No,No,No,MD,02 -

Automobile,2009,TOYOTA,CAMRY,RED,Warning,22-226(a),Transportation

Article,No,BLACK,F,GERMANTOWN,MD,MD,A - Marked Patrol,

03/31/2015,23:59:00,MCP,"5th district, Germantown",DRIVER FAILURE TO STOP AT STOP SIGN LINE,W/B PLYERS MILL RD AT METROPOLITAN AVE,39.0338233333333,-

77.07544,No,No,No,No,No,No,No,No,No,No,MD,02 - Automobile,2007,ACURA,MDX,BLACK,Warning,21-

707(a),Transportation Article,No,WHITE,F,KENSINGTON,MD,MD,A - Marked Patrol,"(39.0338233333333, -

77.07544)"

# Send Data

```
PutRecordRequest putRecordRequest = new PutRecordRequest();  
putRecordRequest.setDeliveryStreamName(deliveryStreamName);
```

```
String data = line + "\n";
```

```
Record record = createRecord(data);  
putRecordRequest.setRecord(record);
```

```
FirehoseClient.putRecord(putRecordRequest);
```

## **Step 4 Query and Analyze the Data from Redshift**



# Query Data

The screenshot displays the Aginity Workbench for Redshift interface. The title bar indicates the connection to a Redshift instance named 'raystreaming'. The main window is divided into three panes: Object Browser, Query Editor, and Output.

**Object Browser:** Shows a tree view of the database structure. The 'Databases (5)' folder is expanded, showing 'raychu.streamingdb (120 MB)', 'redshift\_dev', 'redshift\_prod\_harvest', 'redshift\_template1', and 'redshift\_template2'. The 'Security' folder is also visible.

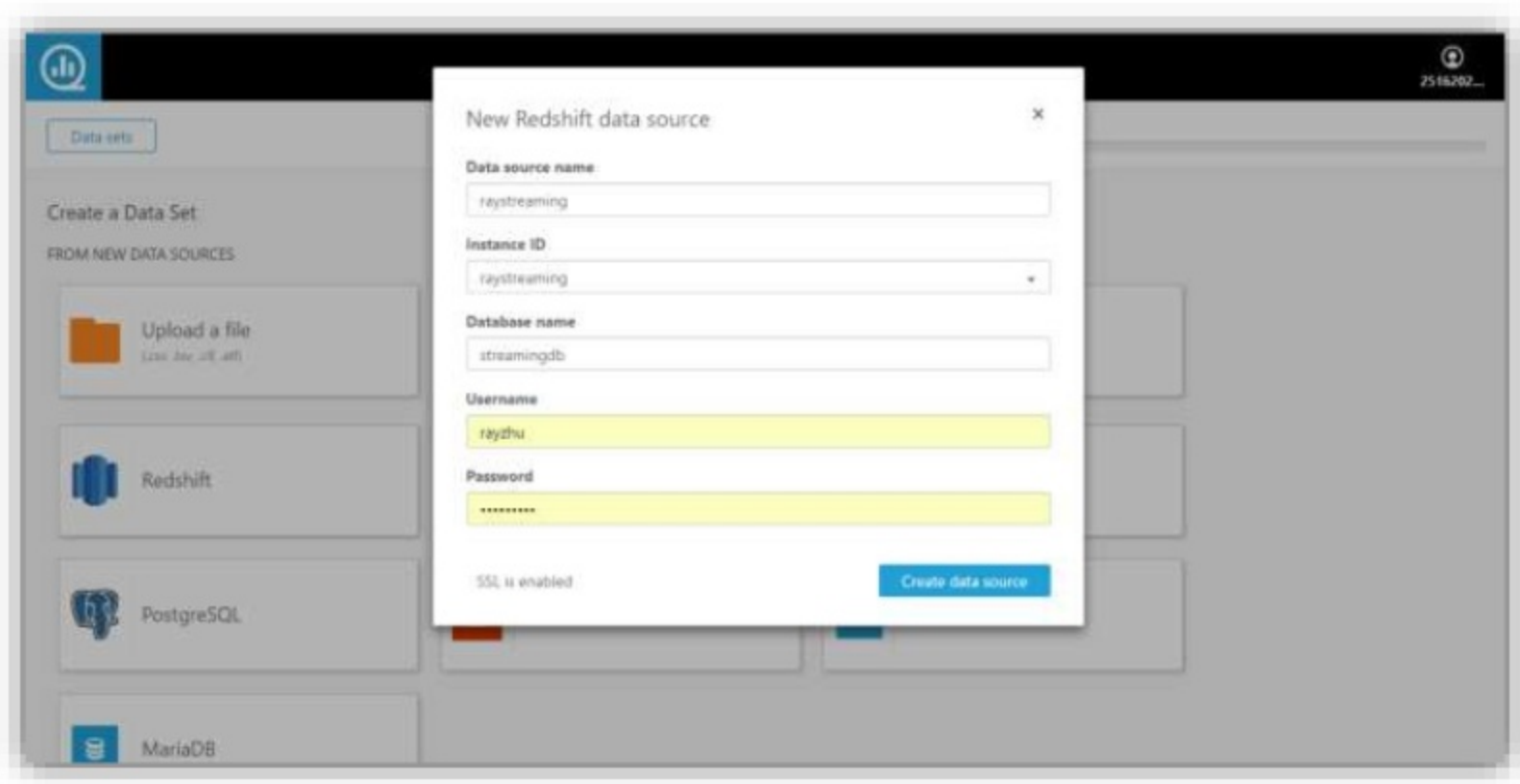
**Query Editor:** Contains a SQL query: `select * from TrafficViolation limit 15`.

**Output:** Displays the results of the query in a table format. The table has 12 columns: `dateofstop`, `timeofstop`, `agency`, `subagency`, `description`, `location`, `latitude`, `longitude`, `accident`, `injury`, `fatal`, and `person`. The results show 15 rows of traffic violation data.

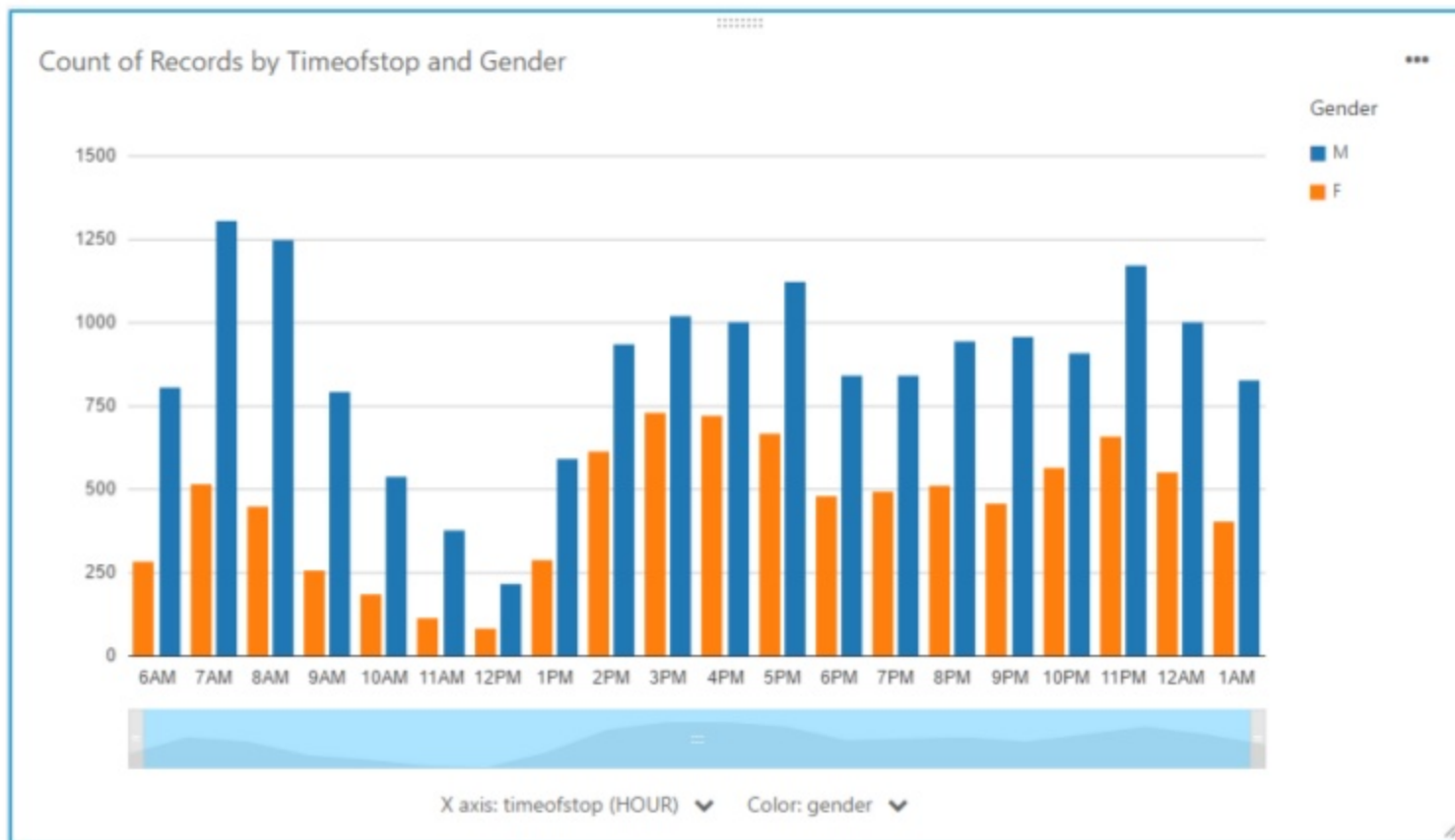
| dateofstop          | timeofstop          | agency | subagency    | description            | location        | latitude        | longitude      | accident | injury | fatal | person |
|---------------------|---------------------|--------|--------------|------------------------|-----------------|-----------------|----------------|----------|--------|-------|--------|
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 6th district | GAITH. FAILURE TO DI.  | MONTGOMERY.     | 39.159508333333 | -77.205595     | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. FAILURE OF IN.    | DAMASCUS CE.    |                 |                | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. EXCEEDING TH.     | RT 118 @ CAW.   | 39.170183333333 | -77.278645     | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 2nd district | Beth. FAILURE TO ST.   | NORFOLK AVE.    | 38.991253333333 | -77.0900666666 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. EXCEEDING TH.     | 118 @ DAWSO.    | 39.170266666666 | -77.275405     | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. DRIVER FAILU.     | RT 108 AT RT 1. |                 |                | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 4th district | Whe. DRIVER USING.     | 12141 GEORGI.   | 39.055216666666 | -77.0491333333 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 2nd district | Beth. FAILURE TO ST.   | NORFOLK AVE.    | 38.989566666666 | -77.08813      | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 3rd district | Salve. DRIVER USING.   | CASTLE BLVD I.  | 39.063383333333 | -76.9421633333 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. DRIVER USING.     | RIDGE RD AT I.  |                 |                | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 3rd district | Salve. FAILURE OF IN.  | ROBEY RD AN.    | 39.079646666666 | -76.9444216666 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. DRIVER READI.     | SB 124 AT RT 1. |                 |                | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 6th district | GAITH. STOP LIGHTS (*) | GOSHEN RD I.    | 39.158286666666 | -77.1919516666 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 5th district | GER. EXCEEDING TH.     | MIDDLEBROOK.    | 39.177833333333 | -77.2427666666 | No       | No     | No    | No     |
| 2015-04-11 00:00:00 | 1900-01-01 19:00:00 | MCP    | 4th district | Whe. FAILURE TO M.     | GEORGIA AVE.    | 39.036708333333 | -77.05147      | No       | Yes    | No    | No     |

The bottom status bar shows 'Ready', 'Pos: 39 (row: 1, col: 40)', 'raychu', and '1 query, 2.234 sec, 15 rows'.

# Connect via QuickSight

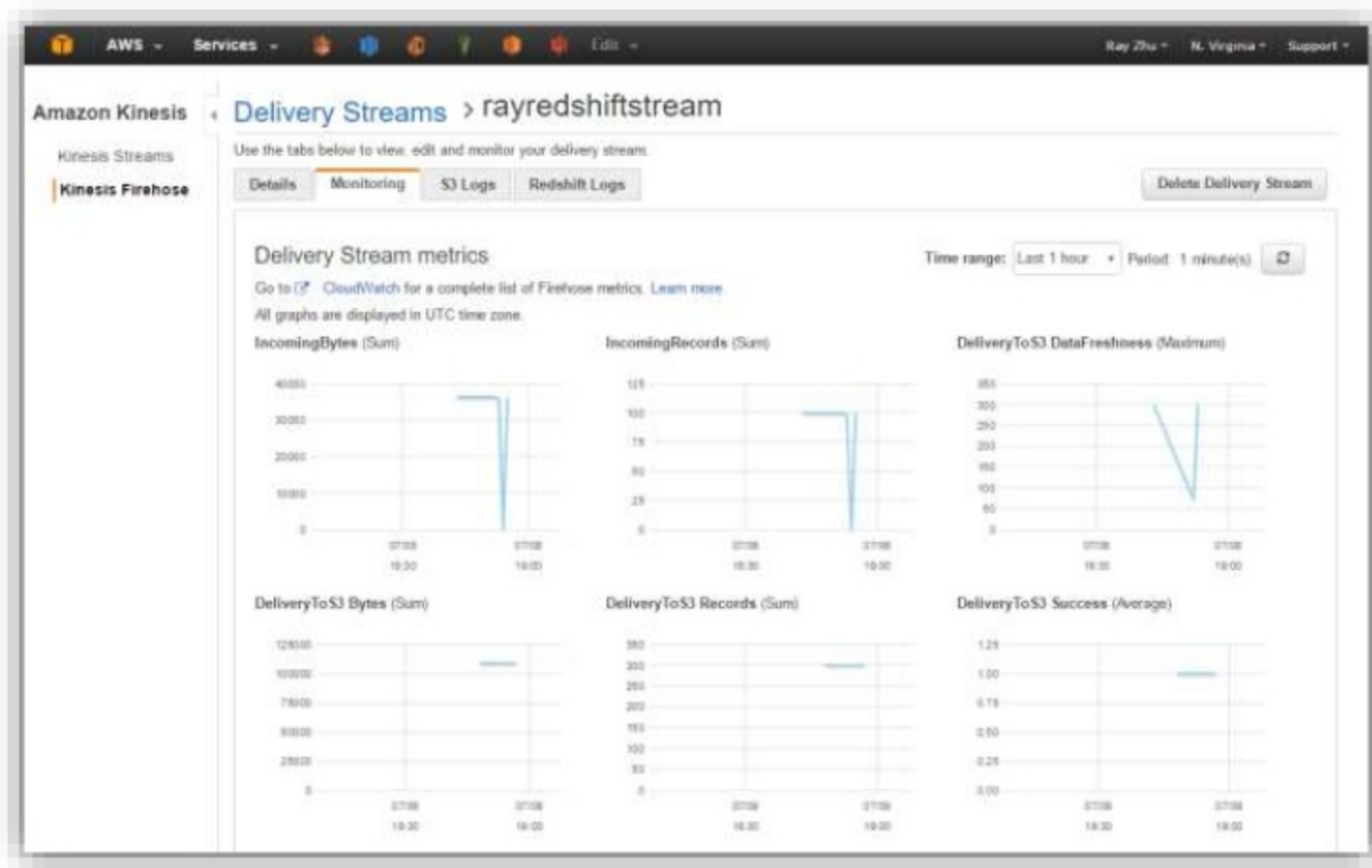


# Visualize Data

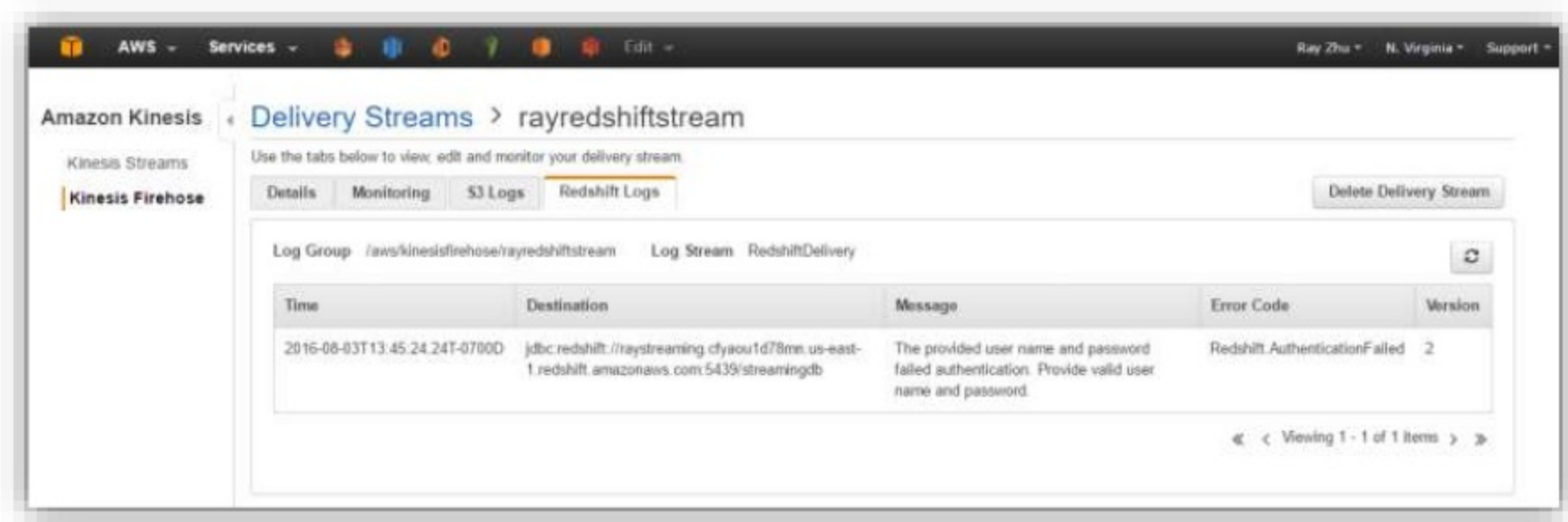


# **Step 5 Monitor Streaming Data Pipeline**

# Monitor with CloudWatch Metrics



# Monitor with CloudWatch Logs



The screenshot shows the Amazon Kinesis console interface. The top navigation bar includes the AWS logo, 'Services', and a user profile 'Ray Zhu' in 'N. Virginia'. The left sidebar shows 'Amazon Kinesis' with 'Kinesis Streams' and 'Kinesis Firehose' options. The main content area is titled 'Delivery Streams > rayredshiftstream'. Below this, a message states 'Use the tabs below to view, edit and monitor your delivery stream.' There are four tabs: 'Details', 'Monitoring', 'S3 Logs', and 'Redshift Logs', with the last one being active. A 'Delete Delivery Stream' button is located to the right of the tabs. Below the tabs, the 'Log Group' is '/aws/kinesisfirehose/rayredshiftstream' and the 'Log Stream' is 'RedshiftDelivery'. A refresh button is on the right. A table displays log entries with columns for Time, Destination, Message, Error Code, and Version. One entry is visible, showing an authentication failure. At the bottom right, it says 'Viewing 1 - 1 of 1 items'.

Amazon Kinesis **Delivery Streams** > rayredshiftstream

Kinesis Streams  
**Kinesis Firehose**

Use the tabs below to view, edit and monitor your delivery stream.

Details Monitoring S3 Logs **Redshift Logs** Delete Delivery Stream

Log Group /aws/kinesisfirehose/rayredshiftstream Log Stream RedshiftDelivery

| Time                          | Destination   | Message  | Error Code                    | Version |
|-------------------------------|---|--|-------------------------------|---------|
| 2016-08-03T13:45:24.247-07000 | jdbc:redshift://raystreaming.cfyaou1d78mn.us-east-1.redshift.amazonaws.com:5439/streamingdb | The provided user name and password failed authentication. Provide valid user name and password. | Redshift.AuthenticationFailed | 2       |

« < Viewing 1 - 1 of 1 items > »

**Q & A**

# Thank you!