Lab Steps

Task 1: Sign in to AWS Management Console

1. Click on the  button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   * Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
   * Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**
4. Select Maybe later in New AWS Console Home page pop-up

A screenshot of a computer

Description automatically generated

**Note: There is no Validation Test for this Lab.**

Task 2: Creating a S3 Bucket

Let us create a S3 Bucket, where we will store the output of the analytics.

1. Make sure you are inthe **US East (N. Virginia) us-east-1** Region.
2. Navigate to **S3** by clicking on the **Services** menu, under the **Storage** section.
3. Click on .
4. In the General Configuration,
   * **Bucket name** :Enter ***whiz-demo-logs***
   * **Note:** S3 Bucket names are globally unique, choose a name that is available.
5. **Region**:Select **US East (N. Virginia) us-east-1** (i.e same region as the Kinesis data stream).
6. Leave other settings as default.
7. Click on **Create bucket**.

A screenshot of a computer

Description automatically generated with medium confidence

Task 3: Creating a Kinesis Firehose Delivery stream

Since we’ll be storing the output analytics into an S3 bucket, let us configure a firehose delivery stream.

1. Make sure you are inthe **US East (N. Virginia) us-east-1** Region.
2. Navigate to **Kinesis** by clicking on the **Services** menu, under the **Analytics** section by clicking [here](https://console.aws.amazon.com/kinesis/home?region=us-east-1#/home).
3. Under **Get Started**, select **Kinesis Data Firehose** and click on 
4. Under **Choose source and destination**,
   * Source : Choose **Direct PUT**
   * Destination : Choose **Amazon S3**
5. Under **Delivery stream name**, give a name of your choice.
   * **Delivery stream name** : Enter ***demo-delivery-stream***
6. Leave the **Transform and convert records** as default.
7. Under the **Destination settings**,
   * Click **Browse**.
   * From the pop-up, select the S3 Bucket we have created earlier, in my case, **whiz-demo-logs**.
   * Click **Choose**.
8. Expand the **Buffer hints, compression and encryption** section.
   * Under the **Buffer size**, make it to **1 MiB**.
   * Under the **Buffer interval**, make it to **60 seconds**.
9. Expand the **Advanced settings**,
   * Under **Permissions**, let it be as default, i.e **Create or update IAM role**. New IAM role with required permission would be created and will be assigned to this Kinesis delivery stream.
10. Click **Create delivery stream**. Wait till the delivery stream is created and is in **Active** state.

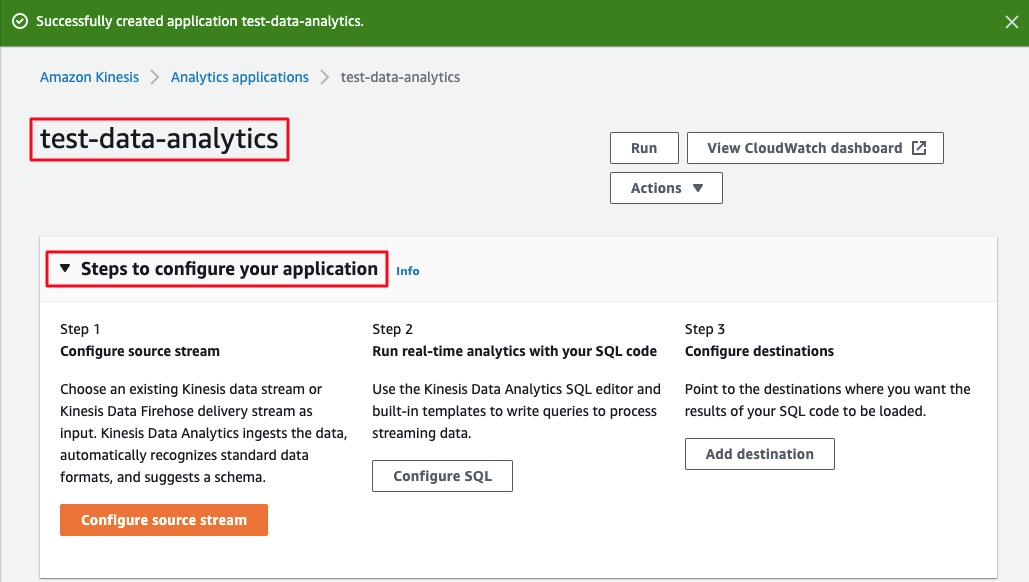
A screenshot of a computer

Description automatically generated with medium confidence

Task 4: Creating a Kinesis Data Analytics application

Let us create a data analytics application. With this application, we can get insights in real time and analyze streaming data.

1. In the Kinesis, on the left side click the  icon to expand the dashboard.
2. Click **Analytics application** and then again click on **SQL applications (legacy)**under Analytics application.
3. Then click on 
4. Under **Application Configuration**,
   * **Application name** : Enter ***test-data-analytics***
   * **Description** : Enter any description
   * **Tags** - Click on **Add Tag**.
     + **Key** : Enter ***Name***
     + **Value** : Enter ***test-data-analytics***
5. Click on .



1. The test application has been provisioned. Expand the **Steps to configure your application.**We’ll be taken through a 3 step process.
   * **Step 1: Configure source stream**
   * **Step 2: Run real-time analytics with your SQL code**
   * **Step 3: Configure destinations**

Task 5: Configuring an Input stream

Let us configure an Input stream for the Kinesis Analytics application..

1. Expand the **Steps to configure your application**.
2. Click the step 1, **Configure source stream**.
3. Under the **Source**, we can create a data stream or a data firehose delivery system but we’ll be creating a demo stream.
4. By using the demo stream we will continuously receive test generated data.
5. Click **Use demo stream**.
6. A demo stream will be provisioned by the AWS where it creates a Kinesis IAM Role, Kinesis stream and a method to auto-generate and populate the Kinesis stream.

A screenshot of a computer

Description automatically generated with low confidence

1. Keep **Record preprocessing with AWS Lambda** and **IAM role for reading source stream** as default.
2. Under **Schema**, click **Discover schema**.
3. Schema discovery can generate a schema using records from the source. You can observe that the Kinesis analytics stream is populated with stock ticker data. A sample data of the incoming stream is displayed within a table. The table column represents each of the detected attributes in a single record.
4. Click .
5. Once the provisioning is completed, you can see that step 1 is in the **Configured** state.

Task 6: Configuring a real-time analytics query

Let us use the Kinesis Data Analytics SQL editor and built-in templates to write queries to process streaming data.

1. Under the step 2, **Run real-time analytics with your SQL code**,  click **Configure SQL**.
2. We’ll be taken to Kinesis Analytics SQL editor. In this SQL editor we can create our own queries. Once we write the SQL query and run the application, it’ll be continually applied to our incoming source data stream.
3. For demo purposes, we will be using the pre-built SQL query from the templates.
4. Click **Add SQL from templates**.
5. In the drop-down, we’ll be able to see many templates.
6. Here, I have selected the template, **Aggregate function in a tumbling time window**. Once you select the template you’ll be presented with the SQL code. Go through the SQL code explanation.
7. Scroll down and select **Append SQL to the editor**.
8. Let us save and run the real time streaming analytics SQL query by clicking **Save and run application**. This will take some time to update and run the application.
9. Click on the **Output**. Select **DESTINATION\_SQL\_STREAM**. You will be able to see that results are added every 5-10 seconds.
10. In the **Output**, select **error\_stream** where you can see the errors encountered in the stream. Since it is a demo stream, we wouldn’t find any.
11. Now, we have configured the real time analytics with SQL code. Scroll down and click on **Back** to continue with the third step.
12. Expand the steps to configure your application and you can see that step 2 is in the **Configured** state.

Task 7: Setting up a delivery stream to store results in S3 bucket

Let us point to the destinations where you want the results of your SQL code to be loaded.

1. Under step 3, **Configure destinations**, click **Add destination**.
2. Since we’ll be storing the output analytics into an S3 bucket, let us select the firehose delivery stream as the destination.
3. Under **Destination** select **Kinesis Data Firehose Delivery Stream**.
4. Click **Browse** and select the delivery stream that we created earlier. If the delivery stream is not present, you can click the **Refresh** icon.
5. Once you select the delivery stream, click **Choose**.
6. Under **Access permissions for writing output stream**, leave it as default.
7. Under **In-application stream name**, select **DESTINATION\_SQL\_STREAM** from the drop-down as we will be storing that particular output in the S3 bucket.
8. Leave the **Output format** as **JSON**.
9. Click **Save changes**.
10. This will take sometime to update the data analytics application.
11. Now we have a Kinesis pipeline from the source (data stream) to the destination (delivery stream) where AWS auto-generates the stock ticker data and puts it to the source stream. Our configured aggregation count query runs continually over the incoming data using a 10 second tumbling time window. The analytics results are present in JSON format and it’ll be delivered into the configure S3 bucket.via the Firehose delivery system.

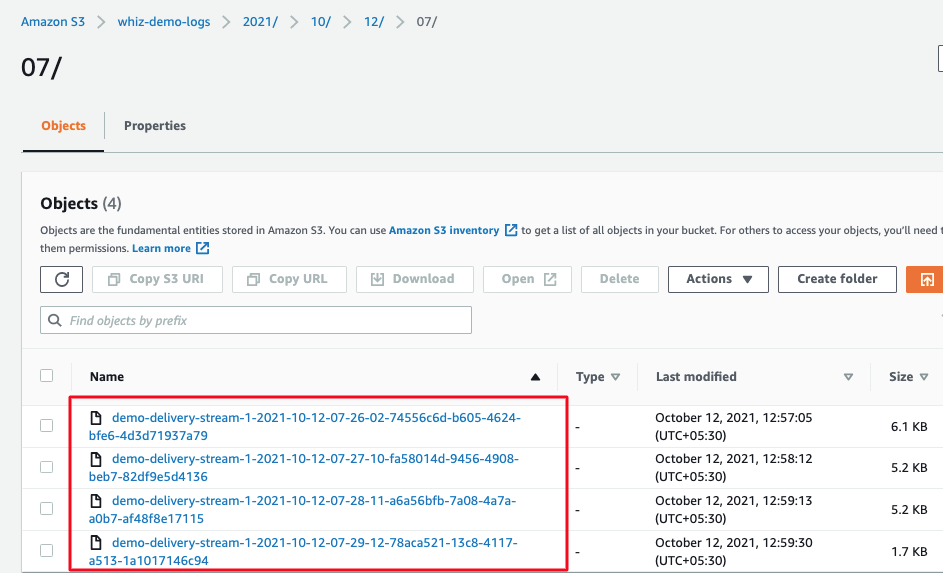
Task 7: Testing the destination

Now we have a Kinesis pipeline from the source (data stream) to the destination (delivery stream) where AWS auto-generates the demo stock data and puts it to the source stream. Our configured aggregation count query runs continually over the incoming data using a 10 second tumbling time window. The analytics results are present in JSON format and it’ll be delivered into the configured S3 bucket via the Firehose delivery system.

1. Make sure you are inthe **US East (N. Virginia) us-east-1** Region.
2. Navigate to **S3** by clicking on the **Services** menu, under the **Storage** section.
3. Click the bucket we have created and used in the Firehose delivery stream. In my case, **whiz-demo-logs**.
4. We have to wait till the logs are displayed in the S3 bucket as we have changed the buffer size and buffer interval while creating the Firehose delivery system. Meanwhile, keep refreshing the bucket by clicking on the refresh icon periodically.

**Note:** Wait for 3-5 minutes, if you are not able to see the logs.

1. You will see a hierarchy of folders with **year** > **month** > **date** > **hour**.
2. Click on the date or hour to see the results created.



1. Click on the result and select **Open** and save the file.
2. Open the file in any text editor in your local computer and check the analytical results.

A screen shot of a computer

Description automatically generated with medium confidence

Task 10: Delete AWS Resources

Deleting Kinesis Analytics application

1. Make sure you are in the **US East** (**N.Virginia) us-east-1** Region.
2. Navigate to **Kinesis** by clicking on the **Services** menu, under the **Analytics** section.
3. On the left panel, click **Analytics application** and then again click on **SQL applications (legacy)**under Analytics application.
4. Select the created application and click on the **Delete** button.
5. Confirm by typing ***delete***and click .

A screenshot of a computer error

Description automatically generated with low confidence

Deleting Kinesis Data Streams

1. Make sure you are in the **US East** (**N.Virginia) us-east-1** Region.
2. Navigate to **Kinesis** by clicking on the **Services** menu, under the **Analytics** section.
3. On the left panel, click on the **Data streams**.
4. Select the created data stream and click on the **Actions** button.
5. Select **Delete** from the drop-down.
6. Confirm by typing ***delete***and click .

A screenshot of a computer

Description automatically generated with low confidence

Deleting Kinesis Delivery Streams

1. Make sure you are in the **US East** (**N.Virginia) us-east-1** Region.
2. Navigate to **Kinesis** by clicking on the **Services** menu, under the **Analytics** section.
3. On the left panel, click on the **Delivery streams**.
4. Select the created delivery stream.
5. Select **Delete**.
6. Confirm by typing the delivery stream name in the field and click .

A picture containing text, screenshot, font, line

Description automatically generated

**Completion and Conclusion**

1. You have created an S3 bucket.
2. You have created a Kinesis Firehose Delivery stream.
3. You have created a Kinesis Data Analytics application.
4. You have configured an Input stream.
5. You have configured a real-time analytics query.
6. You have set up a delivery stream to store results in S3 bucket.
7. You have tested the destination by checking the analytical results stored in the S3 bucket.
8. You have deleted the AWS Resources.

**End Lab**

s