DATA AND
ARTIFICIAL INTELLIGENCE

Big Data on AWS

Real-Time Analytics on Streaming Data with Amazon Kinesis and Amazon Elasticsearch Service

Objectives

- To assist Facebook in continuously monitoring the systems to detect sentiment changes in a social media feed and responding to the sentiment in near real time

Prerequisites

- Amazon Kinesis
- Amazon Elastic Search
- Kinesis Analytics
- SQL
- Kibana

Industry Relevance

- Amazon kinesis: It provides real-time insights and allows to respond quickly to new information by collecting, processing, and analyzing real-time, streaming data.
- Amazon Elastic Search: It is a managed service that makes it easy to deploy, operate, and scale Elasticsearch in the AWS Cloud.
- Kinesis Analytics: It enables you to quickly author SQL code that continuously reads, processes, and stores data in near real-time.
- SQL: It is a standard language for storing, manipulating, and retrieving data in databases.
- Kibana: It is a free and open user interface that allows to visualize the Elasticsearch data and navigate the Elastic Stack.

Problem Statement

Data streaming is used everywhere; from social networks like Facebook to mobile and web applications, IoT devices, instrumentation in data centers, and other sources. As the speed and volume of data increases, the need to perform data analysis in real time with machine learning algorithms and extract a deeper understanding from the data becomes even more important.

Tasks to Perform

1. Create a Kinesis delivery stream
   - Open the amazon kinesis streams console
   - Create a new kinesis stream and give a name that indicates raw incoming stream data, for example, RawStreamData
2. Simulate streaming application to detect anomalies
3. Open Elasticsearch service and create a new domain
   a. Give a unique name to the domain
   b. In the configure cluster screen, use the default settings
   c. In the set-up access policy screen, choose allow access to the domain from specific IP(s)
   d. Enter the public IP address of your computer

Tasks to Perform

4. Configure Kinesis firehose to export the results to Amazon ES
    - Open the Amazon Kinesis Firehose console and choose Create Delivery Stream
    - Choose Amazon Elasticsearch Service from the Destination dropdown menu
    - Type a stream name, and choose the Amazon ES domain that you created in Step 4
    - Provide an index name and ES type. In the S3 bucket dropdown list, choose Create New S3 bucket, then select Next

Tasks to Perform

5. Update the buffer size and existing IAM role for the process
   - In the configuration, change the Elasticsearch buffer size to 1 MB and the buffer interval to 60s. Use the default settings for all other fields. This shortens the time for the data to reach the ES cluster
   - Under IAM Role, choose Create/Update an existing IAM role. The best practice is to create a new role every time. Otherwise, the console keeps adding policy documents to the same role. Eventually, the size of the attached policies causes IAM to reject the role, but it does it in a non-obvious way, where the console basically quits functioning
   - Choose Next to move to the review page

Tasks to Perform

- Review the configuration, and then choose Create Delivery Stream
- Run the Python file for 1–2 minutes, and then press Ctrl+C to stop the execution. This loads some data into the stream for you to visualize in the next step

Tasks to Perform

6. Open the Amazon Kinesis Analytics console and create a new application
   - Open the Amazon Kinesis Analytics console and create a new application. Give the application a name, and then choose Create Application
   - On the next screen, choose Connect to a source. Choose the raw incoming data stream that you created earlier. (Note the stream name Source_SQL_STREAM_001 because you will need it later.)
   - Use the default settings for everything else. When the schema discovery process is complete, it displays a success message with the formatted stream sample in a table, as shown in the following screenshot. Review the data, and then choose Save and continue

Tasks to Perform

7. Connect to the source for further analysis
8. Launch SQL_Editor and start the application
9. Load the processed data into the Kinesis Firehose delivery stream
    - On the application configuration page, choose Connect to a destination
    - Choose the stream name that you created earlier and use the default settings for everything else. Then, choose Save and Continue
    - On the application configuration page, choose Exit to Kinesis Analytics applications to return to the Amazon Kinesis Analytics console

Tasks to Perform

- Run the Python script again for 4–5 minutes to generate enough data to flow through Amazon Kinesis Streams, Kinesis Analytics, Kinesis Firehose, and finally into the Amazon Elastic Search domain
- Open the Kinesis Firehose console, choose the stream, and then choose the Monitoring
- As the processed data flow into Kinesis Firehose and Amazon ES, the metrics appear on the Delivery Stream metrics page. Keep in mind that the metrics page takes a few minutes to refresh with the latest data
- Open the Amazon Elasticsearch Service dashboard in the AWS Management Console

Tasks to Perform

7. Visualize the data using Kibana
   - Use the ES domain link to go to the cluster detail page, and then choose the Kibana link
   - In the Kibana dashboard, choose the Discover tab to perform a query
   - You can also visualize the data using the different types of charts offered by Kibana

Project Outcome

- The aim of the project is to assist Facebook to do a continuous monitoring system to detect sentiment changes in a social media feed to respond to the sentiment in near real time.

Submission Process

1. Complete the project in the Simplilearn lab
2. Complete each task listed in the problem statement
3. Take screenshots of the results for each question and the corresponding code
4. Save it as a document and submit using the assessment tab
5. Tap the "Submit" button (this will present you with three choices)
6. Attach three files and then click "Submit“

Note: Be sure to include screenshots of the output

Thank You