Intermediate Status Report: Real-time Stock Data Streaming and Ingestion (Group 4)

Team Members: Sai Charishma Kurmala, Divya Nalam, Fnu Maria Poulose, Harika Boyina, Madhulika Dutta

1. **Progress towards the goal achieved so far**:
The project has made significant strides toward its goal of real-time stock data streaming and ingestion. Currently, we have successfully implemented a Python script that runs locally on a dedicated machine. This script retrieves real-time stock data, exemplified by IBM, from the Alpha Vantage API and ingests it into an Amazon Kinesis Data Stream.
More about the data source: The API will return the most recent 100 intraday OHLCV bars by default when the output size parameter is not set.
https://www.alphavantage.co/query?function=TIME_SERIES_INTRADAY&symbol=IBM&interval=1min&apikey=demo
The script is running as a background process using 'nohup' to ensure continuous data streaming. The stock in the line is ‘IBM’ and the interval is set to 1 min.
Learn more about Alpha Vantage API at : https://www.alphavantage.co/documentation/

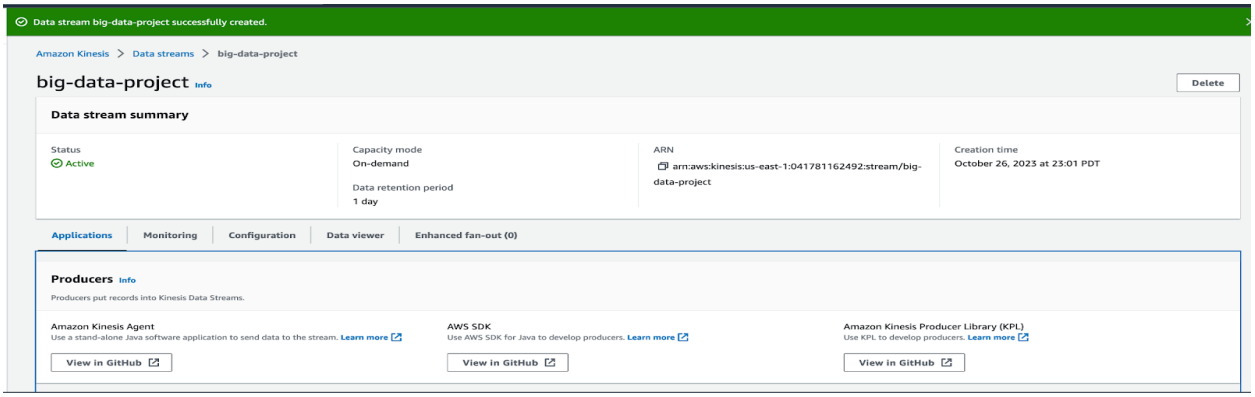
2. **Findings / Results so far**:
As of now, the script is effectively pulling real-time stock data, represented by IBM as a sample, from Alpha Vantage and streaming it into the designated Kinesis Data Stream. We have verified that the data is being ingested as expected, marking a critical milestone in the project's success.

3. **Difficulties being encountered and how we plan to resolve them:**
While the current setup is functional, we have encountered some challenges:
Continuous Machine Running: Running the script locally requires the machine to be always operational. Potential machine failures or the need to restart the script manually can disrupt data streaming. To address this, we plan to move the solution to an AWS Lambda function. Lambda functions are serverless, cost-efficient, and can automatically handle failures and scale as needed.
AWS Lambda Migration: Moving forward, we intend to migrate the script to AWS Lambda. This will eliminate the need for a dedicated machine, reduce operational overhead, and ensure continuous data streaming with minimal human intervention. AWS Lambda also offers event-driven execution, which aligns well with real-time data ingestion.
Cost Efficiency: The transition to AWS Lambda is expected to be cost-efficient as you only pay for the compute resources used during execution. Lambda functions automatically scale based on incoming requests, making it a cost-effective solution.

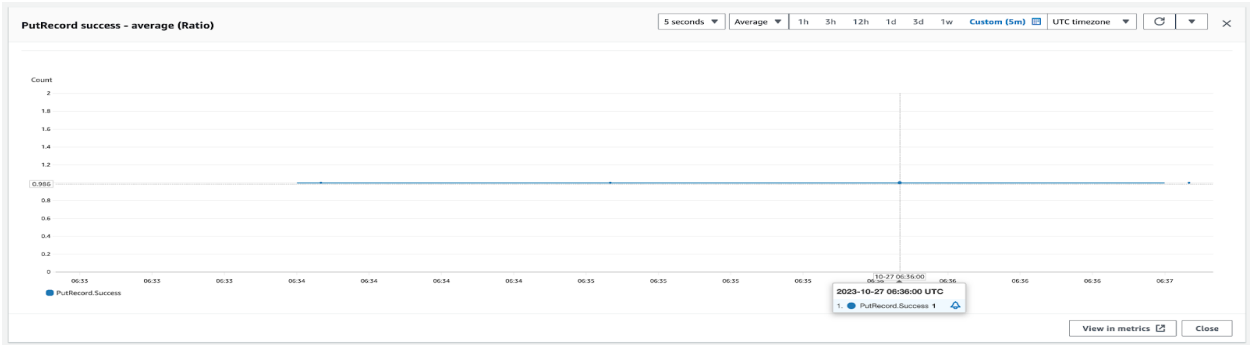
4. **Future Plans and Remaining Tasks:**
As we move forward, the project has several exciting developments and tasks ahead:
Kinesis Data Firehose: After ingesting data into the Kinesis Data Stream, we plan to introduce Amazon Kinesis Data Firehose into the architecture. Kinesis Data Firehose will act as a reliable, serverless way to prepare and load data into other AWS services.

Data Transformation and Analysis: A Lambda function will be integrated with Kinesis Data Firehose to clean, transform, and perform real-time analysis on the incoming stock data. This will enable us to gain valuable insights and make informed decisions.
Visualization with Kibana: To provide a visual representation of the analyzed data, we plan to utilize a visualization tool like Kibana. Kibana will help us create interactive and meaningful visualizations, making it easier to interpret stock market trends and patterns.
Monitoring and Optimization: We will implement robust monitoring and alerting mechanisms to ensure the reliability and performance of the entire data pipeline. Cost optimization strategies will also be applied to manage AWS service costs effectively.

5. **Applicability to Different Stocks:**
While we have used IBM as a sample stock symbol for this project, it's important to note that this solution can be applied to any stock symbol or financial instrument. The architecture is designed to be versatile and adaptable, allowing us to analyze and visualize real-time data for various stocks, enabling users to make informed decisions across the financial markets.
In summary, the project has made substantial progress, and the forthcoming enhancements will further solidify the solution's robustness, efficiency, and flexibility, allowing us to apply it to a wide range of financial instruments and deliver real-time financial insights.



The screenshot displays the creation of a data stream named "big-data-project" in Amazon Kinesis. The stream is currently "Active", operates in an "On-demand" capacity mode, and retains data for 1 day.



Above is a graph of the "PutRecord success - average (Ratio)" over a short time interval. Below are the other obtained graphs: