Streaming Yahoo Finance Data with AWS Lambda

Background

This project involves setting up a few AWS technologies to generate near real time finance data records for interactive querying. This will allow us to capture and analyze data that is being constantly generated in or near to real time. We will utilize the yfinance module that provides a suitable and effective way to download and view historical market data from Yahoo! finance. We will be taking a look at the following stocks:

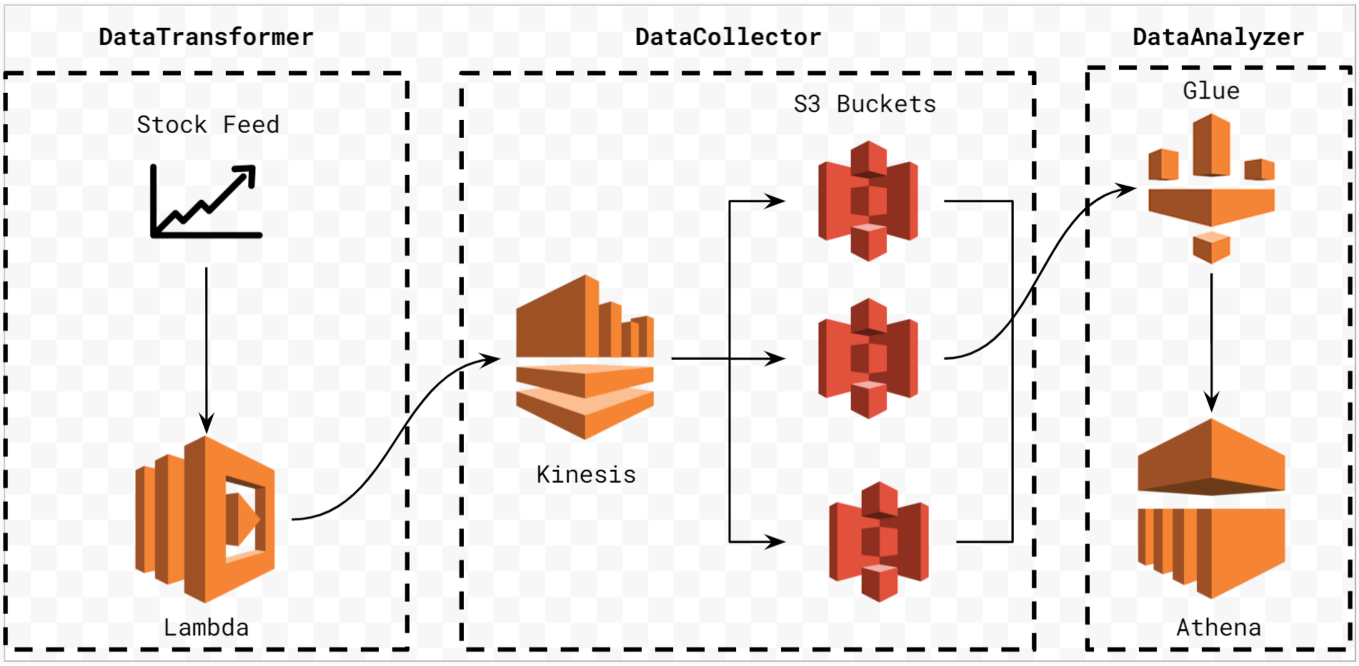
- Facebook (FB), Shopify (SHOP), Beyond Meat (BYND), Netflix (NFLX), Pinterest (PINS)
- Square (SQ), The Trade Desk (TTD), Okta (OKTA), Snap (SNAP), Datadog (DDOG)

Our aim is to leverage an AWS Lamda function as well as AWS Kinesis, S3, Glue and Athena to capture data over a specific time period and run a few queries as well as provide some analsyis and visualizations.

Setting up the Project

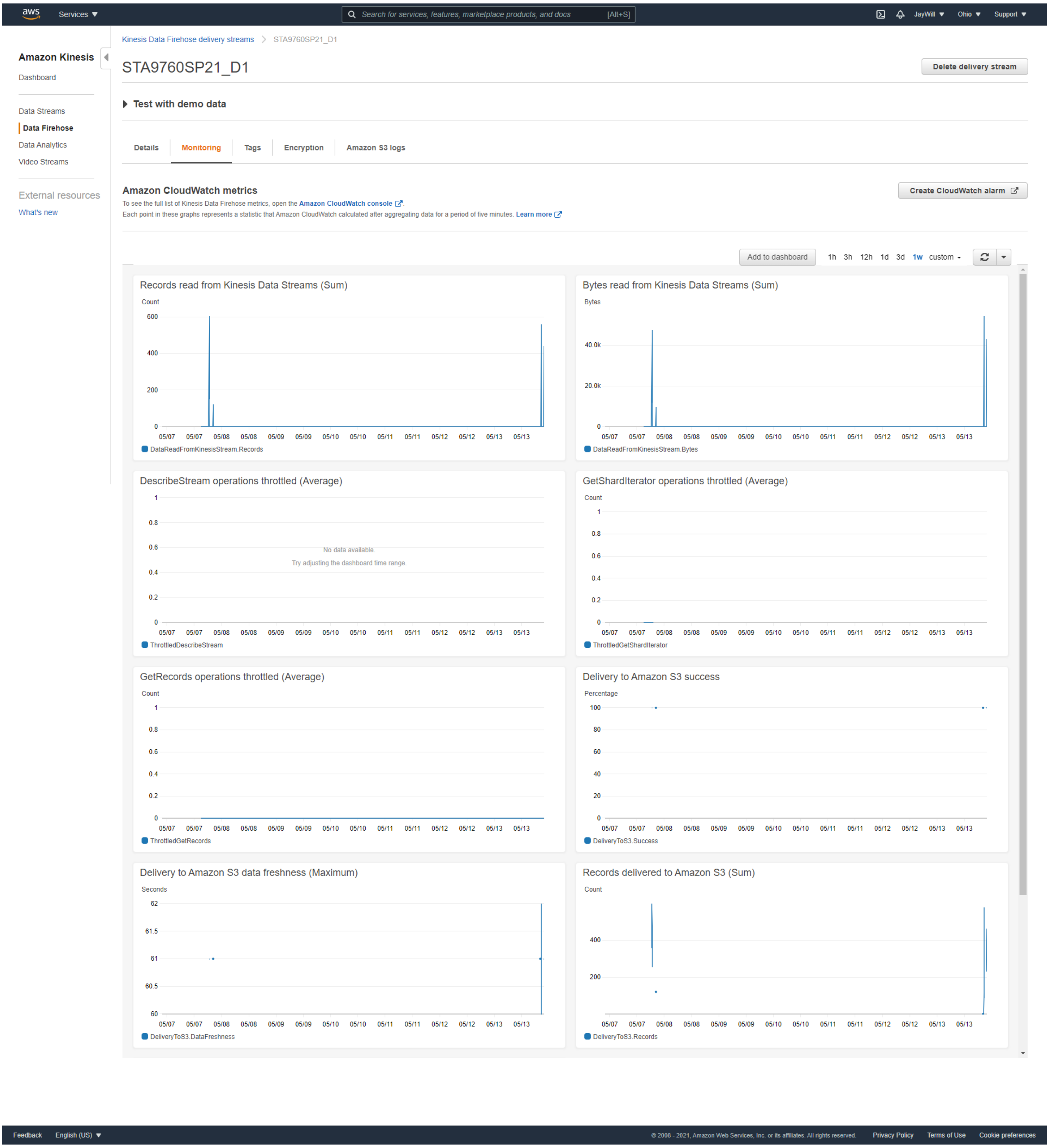
Part I: Design and Inital Setup

1. The following is an overview of the project infrastructure



2. Configure an Amazon Kinesis Stream
    - Ensure that you give a suitable name and then click into it and choose the destination as a one of your S3 buckets.
    - Ensure that the buffer size is set to **1** and the buffer interval set to **60**.

Below is an example of my Kinesis stream



To see the full list of Kinesis Data Firehose metrics, open the Amazon CloudWatch console.

Each point in these graphs represents a statistic that Amazon CloudWatch calculated after aggregating data for a period of five minutes. Learn more

No data available.

Try adjusting the dashboard time range.

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

3. Provision a Simple Lambda Service

    - From the AWS console search for "Lamda" and then select "Create a function"

   - Select " Author from scratch" and ensure that you give in an appropriate name and select **Python 3.8** for the Runtime.
   - Next click on "Add Trigger", select **API Gateway**. Ensure that on the configuration page **Rest API** is selected and security is set to **Open**.

4. Attach IAM Roles to LAmda function

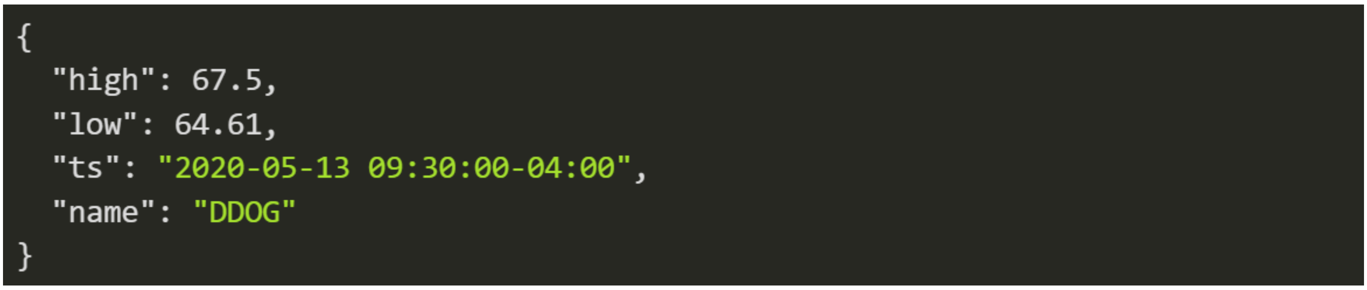
   - Enure that you go to Iam Roles and select the lambda function thgat was created .
   - Select **Attach policies** and ensure you add
     - AmazonKinesisFullAccess
     - AmazonKinesisFirehouseFullAccess
     - KinesisFirehoseServicePolicy....

5. Configure AWS Glue and AWS Athena

   - Confiure Athena to store your queries results inside a folder in your created S3 bucket.
   - Configure Glue choosing your S3 bucket as your data store and create an IAM ROLE to ensure that Athena, Glue and S3 can interact.
   - Add a database and under grouping and select **create a single schema for each S3 path** Is selected for Grouping behavior for S3 data.

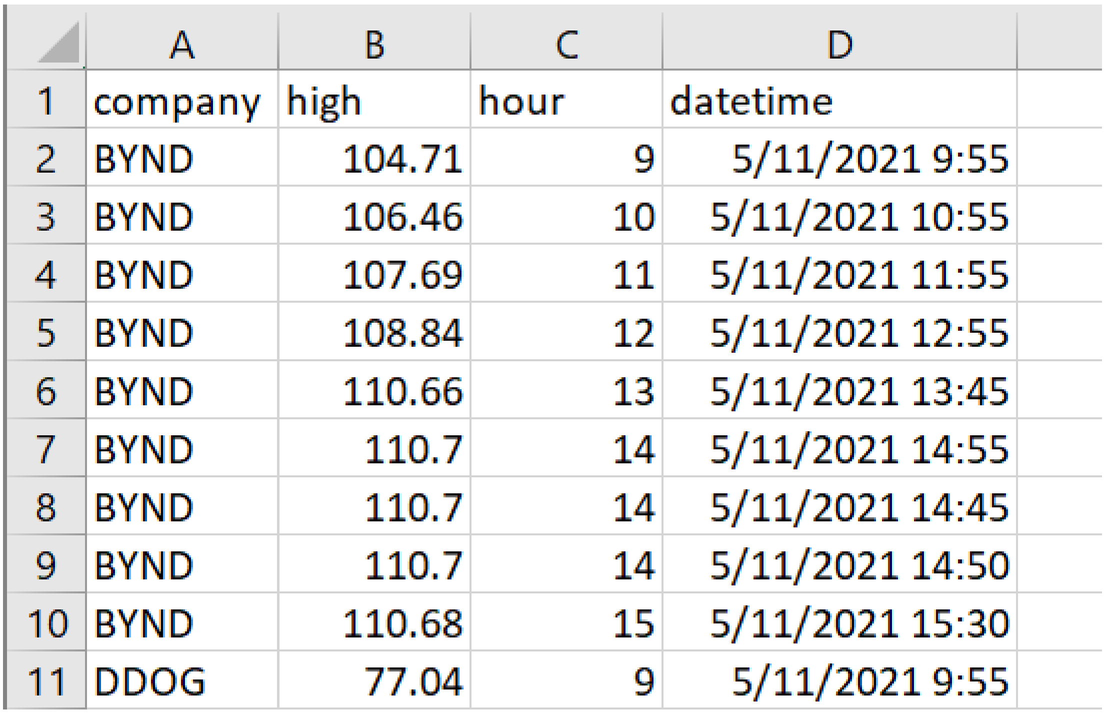
Part II: Data Transformation

In this section we have to develop a lambda function that utilizes the yfinance module to grab the stock information of the ten companies listed above. Speciically we want one full day's worth of the High and Low prices and this information should be collected at five minute intervals. Each singular record you attain should be a JSON object with the following setup



```
{
  "high": 67.5,
  "low": 64.61,
  "ts": "2020-05-13 09:30:00-04:00",
  "name": "DDOG"
}
```

Part III: Data Analysis Once you have collected your data for the stocks, provided everything was properly set up and connected, you can now use AWS Athena to run queries on your data. For this project a query was created to return the highest stock price for each company for every hour of the day. A csv should be generated which has the following look:



	A	B	C	D
1	company	high	hour	datetime
2	BYND	104.71	9	5/11/2021 9:55
3	BYND	106.46	10	5/11/2021 10:55
4	BYND	107.69	11	5/11/2021 11:55
5	BYND	108.84	12	5/11/2021 12:55
6	BYND	110.66	13	5/11/2021 13:45
7	BYND	110.7	14	5/11/2021 14:55
8	BYND	110.7	14	5/11/2021 14:45
9	BYND	110.7	14	5/11/2021 14:50
10	BYND	110.68	15	5/11/2021 15:30
11	DDOG	77.04	9	5/11/2021 9:55