**Stock Market Project – Kafka and AWS**

Create an EC2 instance in AWS:

A screenshot of a computer

Description automatically generated

**Start EC2:**

cd "/Users/bhavanachinnamgari/Projects/Kafka\_project\_stocks"

ssh -i "kafka-stock-market-key.pem" [ec2-user@ec2-75-101-243-117.compute-1.amazonaws.com](mailto:ec2-user@ec2-75-101-243-117.compute-1.amazonaws.com)

A white background with black text

Description automatically generated

Download Kafka and unzip the folder:

wget <https://dlcdn.apache.org/kafka/3.9.0/kafka-3.9.0-src.tgz>

tar -xvf kafka\_2.12-3.9.0.tgz

A screenshot of a computer

Description automatically generated

**Check for Java JDK, install if not present:** sudo yum install java-1.8.0-openjdk

**Change path into folder:** cd kafka\_2.12-3.9.0

**Run zookeeper in Terminal-1:** bin/zookeeper-server-start.sh config/zookeeper.properties

**Start EC2 in Terminal-2 and run Kafka server:**

export KAFKA\_HEAP\_OPTS="-Xmx256M -Xms128M"

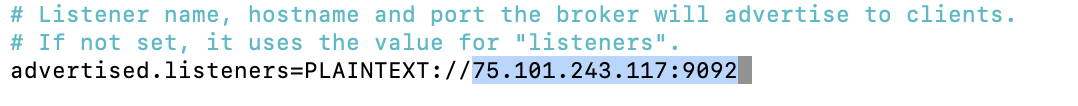
cd kafka\_2.12-3.9.0

bin/kafka-server-start.sh config/server.properties

A close-up of a computer code

Description automatically generated

Change properties file with EC2 port number: sudo nano config/server.properties



CTRL + X, Y, RETURN

Run both Zookeeper and Kafka again and check below in Kafka server:



A screenshot of a computer

Description automatically generated

Now, we have Zookeeper and Kafka setup done!

Let’s create Topic, Producer and Consumer. Start EC2 instances in new terminals and Goto Kafka folder

**Create the topic with replication factor and partitions: Terminal-3**

bin/kafka-topics.sh --create --topic stocks-test6 --bootstrap-server 75.101.243.117:9092 --partitions 1 --replication-factor 1

General command: bin/kafka-topics.sh --create --topic {topic name} --bootstrap-server {EC2 instance/localhost}:9092 --partitions 1 --replication-factor 1

# “stocks-test6” Topic gets created.

**Start Producer: Terminal-3**

bin/kafka-console-producer.sh --topic stocks-test6 --bootstrap-server 75.101.243.117:9092

**Start Consumer in a new Terminal-4**

cd kafka\_2.12-3.9.0

bin/kafka-console-consumer.sh --topic stocks-test6 --bootstrap-server 75.101.243.117:9092 --from-beginning

A screenshot of a computer screen

Description automatically generated

Producer and Consumer are working well!

Login to AWS using CLI with Access ID and Secret ID. Write the python code for the kafka project to display output in Jupyter/ VS code for testing. Then use stocks\_data.csv to populate them in S3 bucket.

Create S3 bucket to save the output:

A screenshot of a computer

Description automatically generated

Run for 5 sec and check the console if the json data is being populated.

A screenshot of a web page

Description automatically generated

Create a AWS Crawler using our S3 bucket

A screenshot of a computer

Description automatically generated

Create a Role to give access to AWS Glue:

A screenshot of a computer screen

Description automatically generated

Create a Database “stock-market-db” to store the table created by Crawler:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A Table is created under the database “stock-market-db”. Now we can execute queries using AWS Athena. Create another S3 bucket to save the query information.

A screenshot of a computer

Description automatically generated