Hands-on Lab: Working with streaming data using Kafka

Estimated time needed: **20** minutes

Objectives

After completing this lab you will be able to:

- Start the Zookeeper server for Kafka metadata management using docker
- Start the Kafka message broker service using docker
- Create a topic
- Start a producer
- Start a consumer

About Skills Network Cloud IDE

Skills Network Cloud IDE (based on Theia and Docker) provides an environment for hands on labs for course and project related labs. Theia is an open source IDE (Integrated Development Environment), that can be run on desktop or on the cloud. to complete this lab, we will be using the Cloud IDE based on Theia running in a Docker container.

Important Notice about this lab environment

Please be aware that sessions for this lab environment are not persistent. A new environment is created for you every time you connect to this lab. Any data you may have saved in an earlier session will get lost. To avoid losing your data, please plan to complete these labs in a single session.

Exercise 1 - start ZooKeeper

ZooKeeper is required for Kafka to work.

1. Start the ZooKeeper server.

```
docker run --name myzookeeper --restart always zookeeper
```

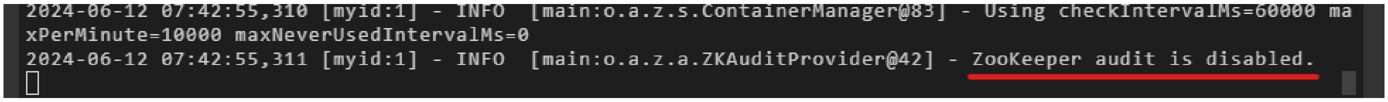
Copied! Executed!

Zookeeper runs on port `2181` at which the server will connect to the zookeeper. When ZooKeeper starts you should see an output that indicates that zookeeper is up and running.

You can be sure it has started when you see no error or exception that makes the server exits and return to command prompt.

ZooKeeper, is not required for Kafka to work in the latest version. However, you will use it to in this lab. ZooKeeper is responsible for the overall management of Kafka cluster. It monitors the Kafka brokers and notifies Kafka if any broker or partition goes down, or if a new broker or partition goes up.

Note: If you see **ZooKeeper audit is disabled** when starting the Zookeeper server, please ignore it and proceed further to next command. It will not hinder your progress.



```
2024-06-12 07:42:55,310 [myid:1] - INFO  [main:o.a.z.s.ContainerManager@83] - Using checkIntervalMs=60000 ma
xPerMinute=10000 maxNeverUsedIntervalMs=0
2024-06-12 07:42:55,311 [myid:1] - INFO  [main:o.a.z.a.ZKAuditProvider@42] - ZooKeeper audit is disabled.
```

Exercise 2 - Start the Kafka broker service

1. Start a new terminal.

2. Run the commands below. This will start the Kafka message broker service.

```
docker run --name mykafkaserver --link myzookeeper:zookeeper apache/kafka
```

Copied! Executed!

Kafka runs on port `9092`. When Kafka starts, you should see an output indicating that the server is up and running.

You can be sure it has started when you see no error or exception that makes the server exits and return to command prompt.

Exercise 3 - Create a topic

You need to create a topic before you can start to post messages.

1. Start a new terminal.
2. To create a topic named `news`, start a new terminal and run the command below.

```
docker exec -it mykafkaserver /opt/kafka/bin/kafka-topics.sh --create --topic news --bootstrap-server localhost:9092
```

Copied!

You will see the message: ‘Created topic news.’

Exercise 4 - Start Producer

You need a producer to send messages to Kafka.

1. Run the command below to start a producer.

```
docker exec -it mykafkaserver /opt/kafka/bin/kafka-console-producer.sh   --bootstrap-server localhost:9092   --topic news
```

Copied!

Once the producer starts, and you get the ‘>’ prompt, type any text message and press enter. Or you can copy the text below and paste. The below text sends three messages to kafka.

```
Good morning
Good day
Enjoy the Kafka lab
```

Copied!

Exercise 5 - Start Consumer

You need a consumer to read messages from kafka.

1. Open a new terminal.
2. Run the command below to listen to the messages in the topic `news`.

```
docker exec -it mykafkaserver /opt/kafka/bin/kafka-console-consumer.sh   --bootstrap-server localhost:9092   --topic news   --from-beginning
```

Copied!

You should see all the messages you sent from the producer appear here.

You can go back to the producer terminal and type some more messages, one message per line, and you will see them appear here.

Exercise 6 - Explore Kafka directories.

Kafka uses the directory /tmp//tmp/kraft-combined-logs to store the messages.

1. Open a new terminal.
2. Run the following command to open the kafka server in bash mode.

```
docker exec -it mykafkaserver /bin/bash
```

Copied!

4. Explore the root directory of the server.

```
ls
```

Copied!

5. You will observer that there is a `tmp` folder. The `kraft-combine-logs` inside the `tmp` folder contains all the logs. You can check the ones generated for topic `news` that you have created by running the following command.

```
ls /tmp/kraft-combined-logs/news-0
```

Copied!

This is where all the messages are stored.

6. To exit from the bash, use the following command:

```
exit
```

Copied!

Exercise 8 - Clean up

1. To stop the producer, in the terminal where you are running producer press `CTRL+C`.

2. To stop the consumer, in the terminal where you are running producer press `CTRL+C`.

3. Stop the zookeeper and kafka server instance, run the following command.

```
docker stop myzookeeper mykafkaserver
```

Copied!

Practice exercises

1. Problem:

   Create a new topic named `weather`.

- ▶ Click here for Hint
- ▶ Click here for Solution

2. Problem:

   Post messages to the topic `weather`.

- ▶ Click here for Hint
- ▶ Click here for Solution

3. Problem:

   Read the messages from the topic `weather`.

- ▶ Click here for Hint
- ▶ Click here for Solution

Authors

Lavanya T S

Other Contributors

Rav Ahuja