Installing KAFKA Single Node - Quick Start.

Apache Kafka is publish-subscribe messaging rethought as a distributed commit log. Kafka is a distributed, partitioned, replicated commit log service. It provides the functionality of a messaging system, but with a unique design. What does all that mean?

First let's review some basic messaging terminology:

- Kafka maintains feeds of messages in categories called topics.

- We'll call processes that publish messages to a Kafka topic producers.

- We'll call processes that subscribe to topics and process the feed of published messages consumers..

- Kafka is run as a cluster comprised of one or more servers each of which is called a broker.

  http://kafka.apache.org/documentation.html#introduction

Download and Extract

Download the `tgz` file and extract.

```
[kafka-admin@kafka Downloads]$ ls
jdk-7u75-linux-x64.rpm  kafka_2.9.2-0.8.2.0.tgz
[kafka-admin@kafka Downloads]$ sudo rpm -ivh jdk-7u75-linux-x64.rpm
...
[kafka-admin@kafka Downloads]$ sudo tar -xzf kafka_kafka_2.9.2-0.8.2.0.tgz -C /opt
[kafka-admin@kafka Downloads]$ cd /opt
[kafka-admin@kafka opt]$ sudo ln -s kafka_2.9.2-0.8.2.0 kafka
[kafka-admin@kafka opt]$ ls
kafka  kafka_2.9.2-0.8.2.0
[kafka-admin@kafka opt]$ sudo chmod kafka-admin:kafka-admin -R kafka
```

Now we are ready to start all the services required.

```
[kafka-admin@kafka opt]$ cd kafka
[kafka-admin@kafka kafka]$ ls
bin  config  libs  LICENSE  logs  NOTICE
[kafka-admin@kafka kafka]$ bin/zookeeper-server-start.sh config/zookeeper.properties
```

This will start us a zookeeper in `localhost` on port 2181. This configuration can be changed in the `config/zookeeper.properties` file. NOTE : If you want to run the zookeeper on a separate machine make sure the change in the `config/server.properties` so that the kafka server points to the correct zookeeper. By default it points to `localhost:2181`.

Next we start server.

```
[kafka-admin@kafka kafka]$ bin/kafka-server-start.sh config/server.properties
```

NOTE : If you want to start multiple make sure you make multiple copies of the `server.properties` file and change the below information.

1. `broker.id` is the unique identifier for the service.

2. `port` where this server is going to `listen` on.

3. `log.dir` where to right the log.
   config/server-1.properties: broker.id=1 port=9093 log.dir=/tmp/kafka-logs-1
   config/server-2.properties: broker.id=2 port=9094 log.dir=/tmp/kafka-logs-2

Now our server has started, lets assume we start only one server.

Creating Topics

To create a topic just execute below command, this will create a single partition.

```
[kafka-admin@kafka kafka]$ bin/kafka-topics.sh --create --zookeeper localhost:2181 \
                                    --replication-factor 1 --partitions 1 --topic test
```

To check topics currently running. Execute below command.

```
[kafka-admin@kafka kafka]$ bin/kafka-topics.sh --list --zookeeper localhost:2181
test
[kafka-admin@kafka kafka]$
```

We see currently we have only one topic. Now we are all set to send and recv messages.

Send some message

Open up a new terminal and fire up the Kafka producer script as below. And start typing some message `\n` or `cr` will be end of each message

```
[kafka-admin@kafka kafka]$ bin/kafka-console-producer.sh --broker-list localhost:9092 \
                                                                        --topic test

This is a message
This is a message2
```

Start a Consumer

Open a new terminal and start the consumer.

Option `--from-beginning` will give all the messages from the beginning. You will see 2 messages as we typed above `This is a message` and `This is a message2`.

```
[kafka-admin@kafka kafka]$ bin/kafka-console-consumer.sh --zookeeper localhost:2181 \
                                                --topic test --from-beginning
This is a message
This is a message2
```

Our single node Kafka cluster is Ready.