sudo apt update

sudo apt install openjdk-17-jdk -y

wget https://downloads.apache.org/kafka/4.0.0/kafka\_2.13-4.0.0.tgz

tar -xvzf kafka\_2.13-4.0.0.tgz

cd kafka\_2.13-4.0.0

cd kafka\_2.13-4.0.0/config

mkdir -p kraft

cd kraft

nano server.properties

Add this content to file :

# Unique identifier for each node in the cluster

node.id=1

# Define the roles for this node

process.roles=broker,controller

# Specify the quorum voters (node.id@host:port)

controller.quorum.voters=1@localhost:9093

# Define listeners for broker and controller

listeners=PLAINTEXT://localhost:9092,CONTROLLER://localhost:9093

controller.listener.names=CONTROLLER

inter.broker.listener.name=PLAINTEXT

# Map listener names to security protocols

listener.security.protocol.map=CONTROLLER:PLAINTEXT,PLAINTEXT:PLAINTEXT

# Directory where Kafka will store its data

log.dirs=/tmp/kraft-combined-logs

After saving create cluster

KAFKA\_CLUSTER\_ID="$(bin/kafka-storage.sh random-uuid)"

bin/kafka-storage.sh format --cluster-id="$KAFKA\_CLUSTER\_ID" --config config/kraft/server.properties

* Back to kafka directory :

bin/kafka-storage.sh format -t $(bin/kafka-storage.sh random-uuid) -c config/kraft/server.properties

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

* In a new terminal, create a topic named test-topic:

bin/kafka-topics.sh --create --topic test-topic --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1

* in another new terminal , but same kafka folder :

sudo apt install sqlite3

sudo apt install python3-pip

If this error occurs when installing python :

E: Could not get lock /var/cache/apt/archives/lock. It is held by process 9978 (unattended-upgr)

N: Be aware that removing the lock file is not a solution and may break your system.

E: Unable to lock directory /var/cache/apt/archives/

Then do :

sudo lsof /var/cache/apt/archives/lock

sudo kill -9 9978

sudo rm /var/cache/apt/archives/lock

sudo dpkg --configure -a

Then try again .

pip install kafka-python faker

nano ~/.bashrc

Add this at the end of file :

export PATH="$HOME/.local/bin:$PATH"

source ~/.bashrc  
  
nano producer.py:

from kafka import KafkaProducer

from faker import Faker

import json

import time

fake = Faker()

producer = KafkaProducer(

bootstrap\_servers='localhost:9092',

value\_serializer=lambda v: json.dumps(v).encode('utf-8')

)

while True:

data = {

'name': fake.name(),

'email': fake.email(),

'address': fake.address(),

'timestamp': time.time()

}

print(f"Sending: {data}")

producer.send('test-topic', value=data)

time.sleep(1)

consumer.py  
  
from kafka import KafkaConsumer

import json

import sqlite3

# Set up SQLite database

conn = sqlite3.connect('kafka\_data.db')

cursor = conn.cursor()

cursor.execute('''

CREATE TABLE IF NOT EXISTS users (

id INTEGER PRIMARY KEY AUTOINCREMENT,

name TEXT,

email TEXT,

address TEXT,

timestamp REAL

)

''')

conn.commit()

# Set up Kafka consumer

consumer = KafkaConsumer(

'test-topic',

bootstrap\_servers='localhost:9092',

value\_deserializer=lambda m: json.loads(m.decode('utf-8')),

auto\_offset\_reset='earliest',

enable\_auto\_commit=True

)

for message in consumer:

data = message.value

print(f"Received: {data}")

cursor.execute('''

INSERT INTO users (name, email, address, timestamp)

VALUES (?, ?, ?, ?)

''', (data['name'], data['email'], data['address'], data['timestamp']))

conn.commit()

python3 consumer.py

python3 producer.py

To check database : in the same kafka.version folder, do :

sqlite3 kafka\_data.db

.tables

.schema users

SELECT \* FROM users;

.exit