Goal

Use the Kafka REST Proxy through an Oracle APEX client application.
This Oracle APEX sample App, available on Github, includes a simple KAFKA producer and a KAFKA consumer. It can be used to get more familiar with the produce/consume process and the *commit* features subtilities.
About the sample demo, let's assume a collection of devices spreaded in several cities and the ability given to any operator to insert a manual message in the stream or polling events from a given topic.

Contenu Afficher

Prerequisites

Either install an on-premise Apache KAFKA cluster, or use a docker image or subscribe to Confluent platform.

If on-premise KAFKA installation, one must install, at least, the community version of Confluent REST Proxy and setup TLS in order to make calls from a free tiers APEX instance. (In case of apex.oracle.com, it's possible to call a http endpoint instead https) cf Oracle rules.

The APEX application is available on github. Export has been made with a version 23.2.

Installation of Kafka

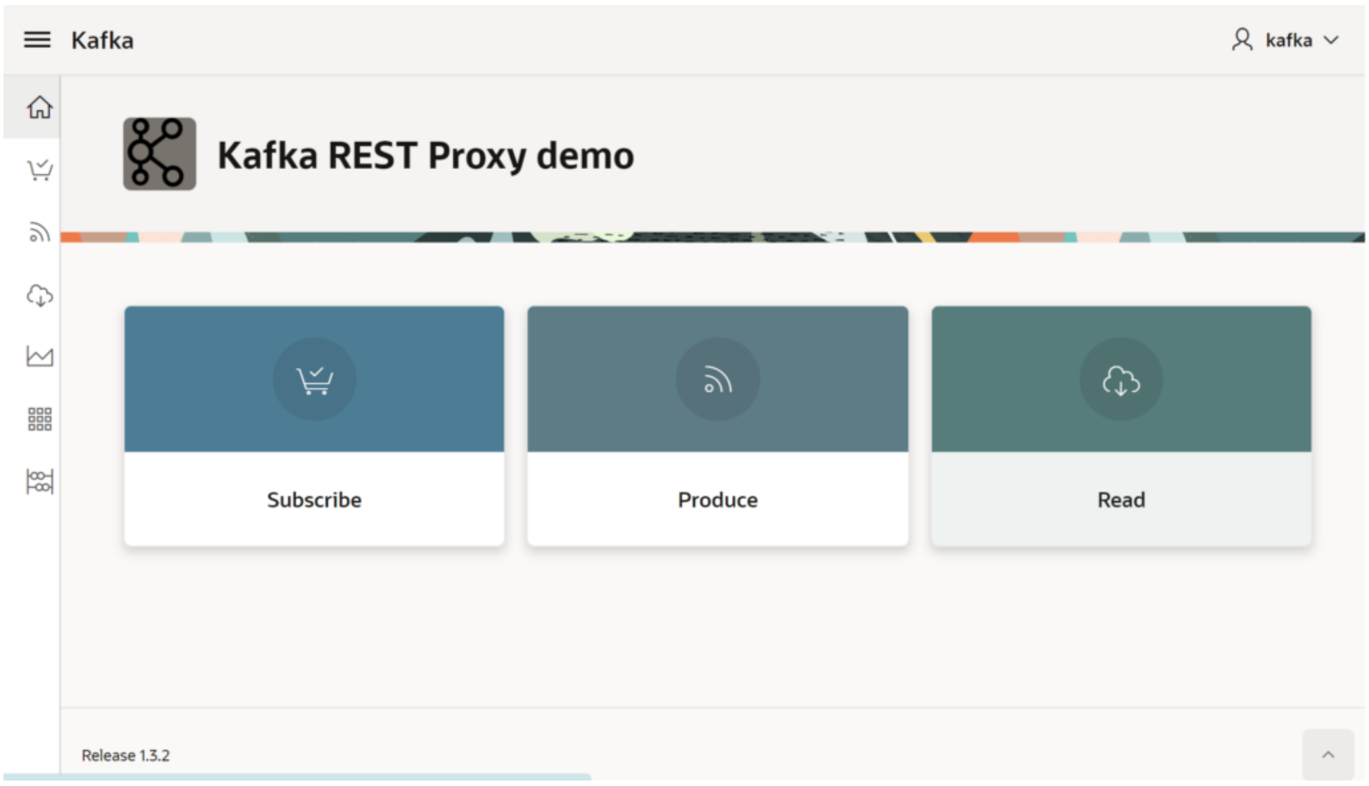
For a single Broker, on a Linux server, follow the links:

- Download Apache Kafka

- Install Kafka
- Install REST Proxy
- Create start/Stop scripts (cf appendices)

About setup TLS for the REST Proxy, read for instance the very good post from Ken Coenen and get infos related to openssl and keystore and adapt the file etc/kafka-rest/kafka-rest.properties.
Read REST Proxy Securit and adapt ssl.client.authentication.

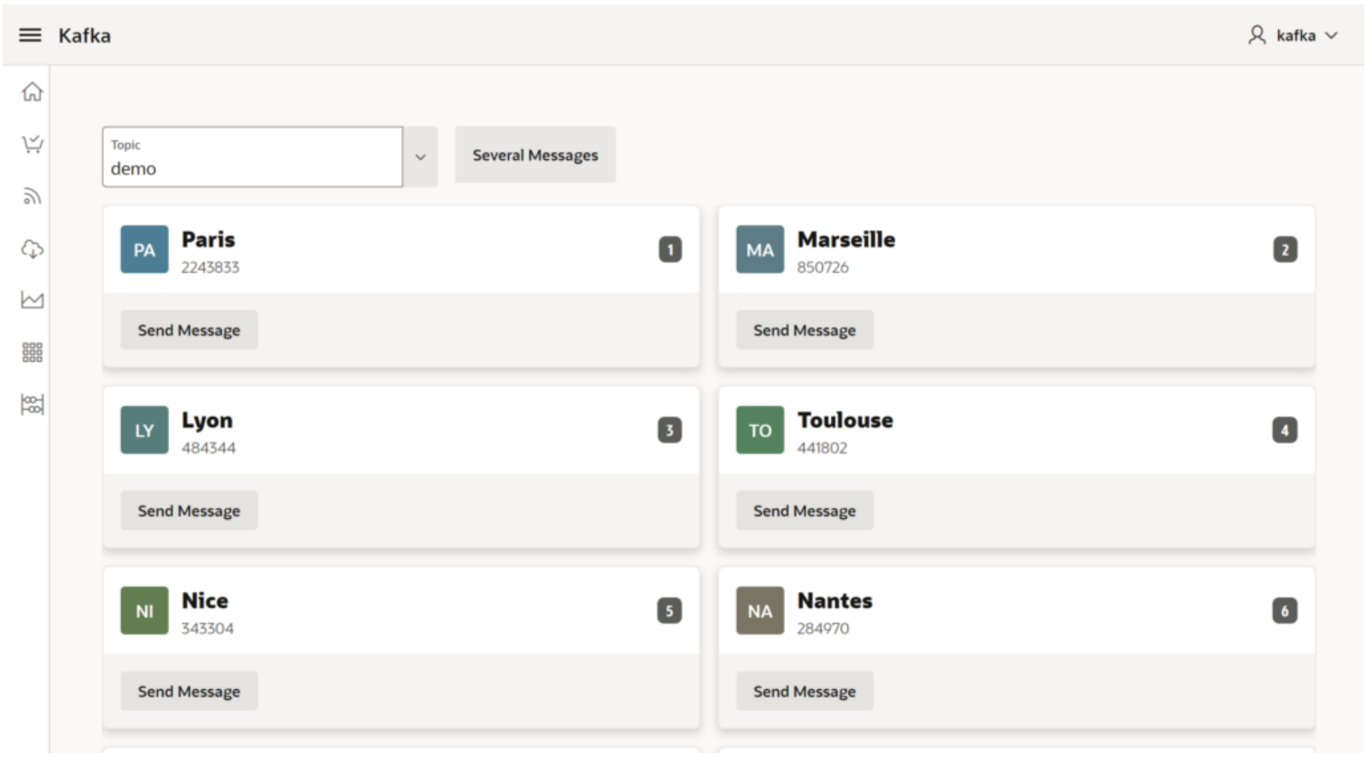
Description of application



A regular APEX app, named Kafka, relies on a package (KAFKA_PKG) which wraps calls to the REST Proxy. The material is available from Github.
Import it in a Oracle APEX instance >= 23.2.

During import process, set the REST Proxy endpoint and accept installation of supporting objects
Launch Kafka app, jump in *setup* option, check the *default consumer name* (ie: patrick) and the *Consumer Group name* (default: *my_json_consumer_group*).
The producer menu option proposes to add only one message or a batch of ten records based on the content of VILLES table. That can be changed in the package KAFKA_PKG.

Sending Messages



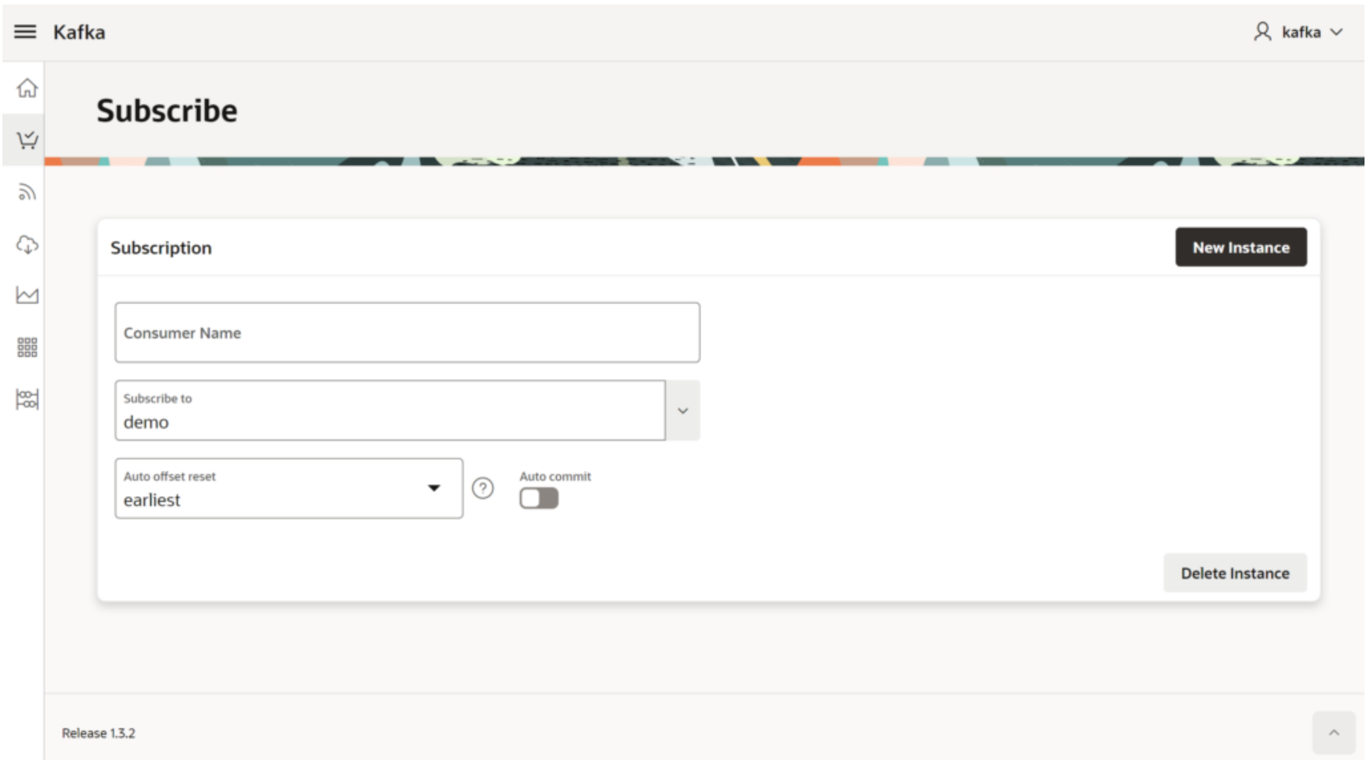
Topic: demo

PA Paris 2243833 1
MA Marseille 850726 2
LY Lyon 484344 3
TO Toulouse 441802 4
NI Nice 343304 5
NA Nantes 284970 6

Application offers following features:

- Listing existing topics
- Creating/deleting a consumer instance and subscribing to one topic
- Consuming records from an offset.

The records page relies on a data source and the other actions are implemented in a dedicated PLSQL package : KAFKA_PKG. This package is embeded as a supporting object in the APEX application.



Consumer Name

Subscribe to: demo

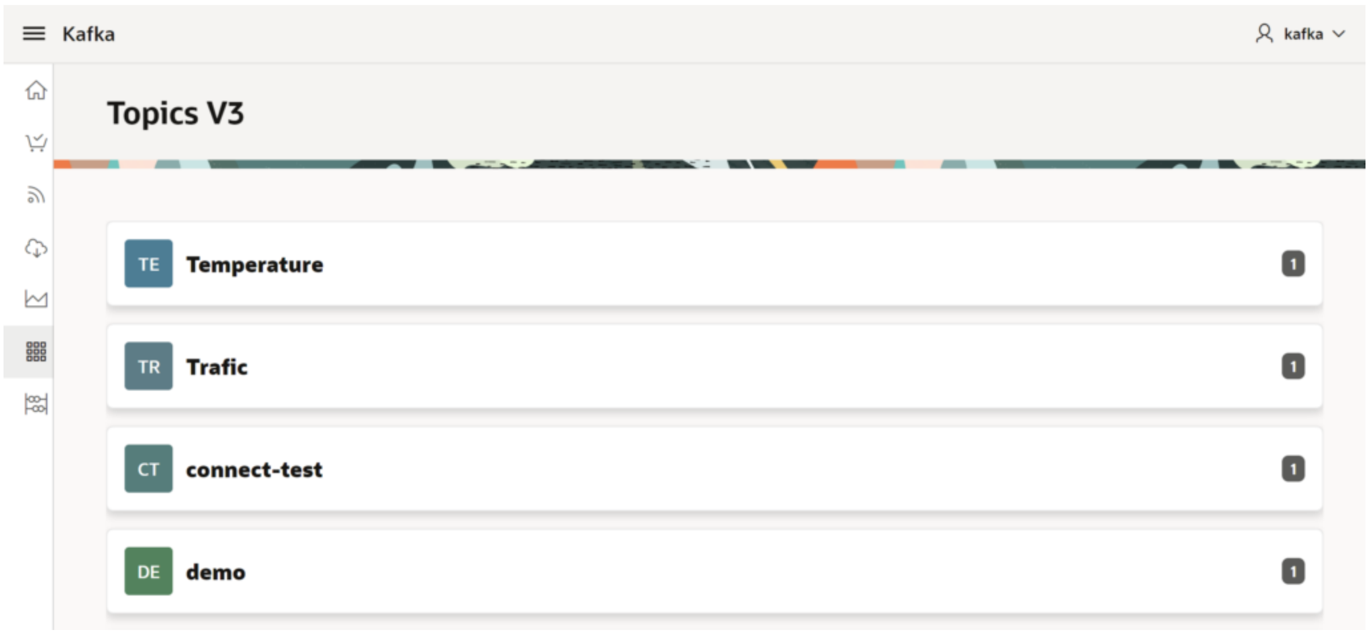
Auto offset reset: earliest

Auto commit

Notes about the consumer instance

When creating a new consumer instance in a consumer group, the max iddle session is set at the server side around 4 minutes. That means that we have to poll regularly, otherwise, we must re-create a new consumer instance. The sample application doesn't catch this situation, but there is a page which draws a chart on a regular basis and that prevents a too long idle time.

List of existing Topics



Clicking on a topic entry gives the lags between the last commit point and the last entry.

Appendices

Scripts for starting and stopping Apache KAFKA

It’s strangely tricky to start Kafka at boot, for obscure reasons of permissions, even as root …
I didnt’ want to dig in these details, not important in my context.

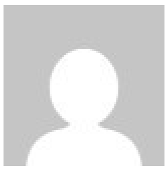
So I just mention two scripts to launch manually the needed modules.
Because I used Kafka with Zookeeper, the first one starts *zookeeper*, then a Kafka server in background.
(Another option is to use Kafka with KRaft)
The second script launches REST Proxy. We can choose to let it in foreground or as a daemon.

Copied from
https://stackoverflow.com/questions/34512287/how-to-automatically-start-kafka-upon-system-startup-in-ubuntu

These following scripts are available on the Github repository.

- Start/Stop/Status Zookeeper and Kafka
- Start/stop/Status REST Proxy

Author

- 

  Patrick

  GPM Factory