**Доступ к Kafka из/вне Docker контейнеров**

При локальной разработке с Kafka могут возникнуть проблемы с доступом, если она запущена в Docker контейнере, и требуется обращение как изнутри Docker так и снаружи.

В Kafka запись и чтение осуществляется в ноду лидера для заданной партиции. Клиент может обратиться к любому узлу Kafka, чтобы узнать по какому адресу нужно делать операцию записи/чтения, даже если Kafka представлена одним узлом. Здесь и возникает путаница, если у вас несколько сетей (например, внешняя и внутренняя). Для внешнего пользователя нужно предоставить внешний адрес, а для внутреннего внутренний, соответственно.

Для решения этой проблемы будут использоваться два порта:

9092 - для доступа извне, в моем случае это хосте (мой компьютер)

**29092** - порт для обращения к Kafka других контейнеров (например, **Debezium**) внутри Docker.Пример такой собранной конфигурации (docker-compose.yaml):

version: '2'

services:

zookeeper:

image: confluentinc/cp-zookeeper:5.4.0

environment:

ZOOKEEPER\_CLIENT\_PORT: 2181

ZOOKEEPER\_TICK\_TIME: 2000

kafka:

image: confluentinc/cp-kafka:5.4.0

ports:

- 9092:9092

environment:

KAFKA\_BROKER\_ID: 1

KAFKA\_ZOOKEEPER\_CONNECT: zookeeper:2181

KAFKA\_LISTENERS: PLAINTEXT://0.0.0.0:**29092**,PLAINTEXT\_HOST://0.0.0.0:9092

KAFKA\_ADVERTISED\_LISTENERS: PLAINTEXT://kafka:**29092**,PLAINTEXT\_HOST://localhost:9092

KAFKA\_LISTENER\_SECURITY\_PROTOCOL\_MAP: PLAINTEXT:PLAINTEXT,PLAINTEXT\_HOST:PLAINTEXT

KAFKA\_INTER\_BROKER\_LISTENER\_NAME: PLAINTEXT

KAFKA\_OFFSETS\_TOPIC\_REPLICATION\_FACTOR: 1

connect:

image: **debezium**/connect:1.4

ports:

- 8083:8083

links:

- kafka

environment:

- BOOTSTRAP\_SERVERS=PLAINTEXT://kafka:29092

- GROUP\_ID=1

- CONFIG\_STORAGE\_TOPIC=dbz\_connect\_configs

- OFFSET\_STORAGE\_TOPIC=dbz\_connect\_offsets

- STATUS\_STORAGE\_TOPIC=dbz\_connect\_statuses