Unlocking the Stream – a journey into the world of Kafka on Aiven

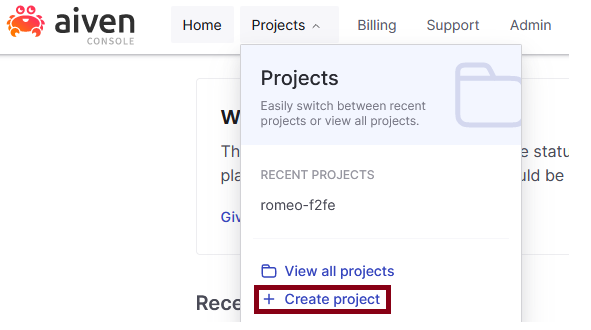
In today's data-driven world, the ability to harness real-time data streams has become a critical element in building responsive and agile applications. Apache Kafka, a distributed event streaming platform, has emerged as a powerful solution for ingesting, processing, and analyzing vast volumes of data in real-time.

When you couple the capabilities of Kafka with the reliability and convenience of Aiven's managed services, you open the door to a world of possibilities. In this blog post, we'll embark on a journey to explore the synergy between Kafka and Aiven, and discover how this dynamic duo can supercharge your data pipelines and applications.

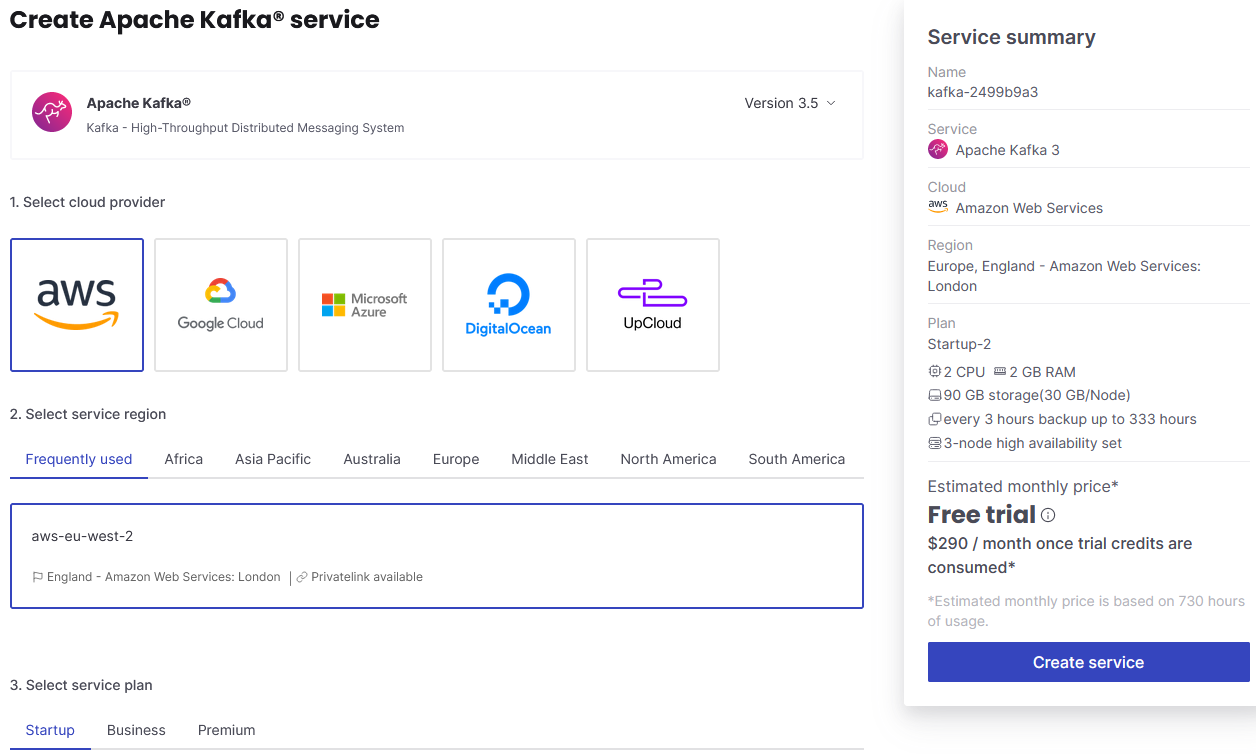
Whether you're a seasoned developer or just beginning your data streaming adventure, Kafka on Aiven promises an exciting ride through the realms of real-time data processing.

Setting up your services in Aiven

1. Log in to Aiven at <https://console.aiven.io/>
2. From the top menu click on Projects > Create a project.

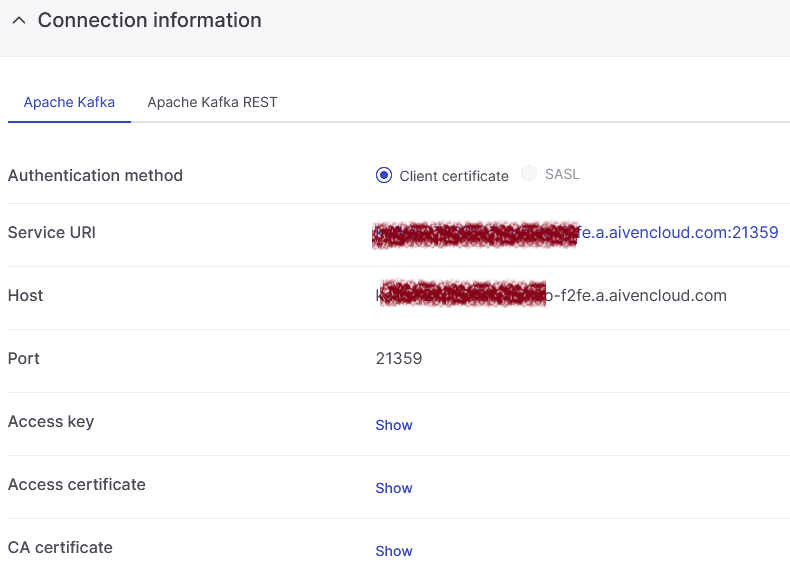


1. Click on the **Create service** button and select **Kafka**. Pick your desired cloud provider and region.



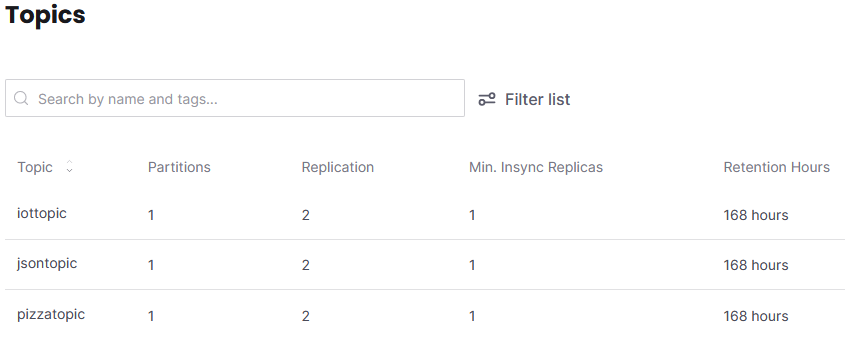
Note: The resources allocated to your Kafka cluster will vary according to the plan you select. For the scope of this exercise, the Startup plan will suffice.

1. The provisioning of the newly-created Kafka service takes up to 5 minutes. Once provisioned, you will be presented with the connectivity details.



From this screen you can also download your access key, access certificate and CA certificate. Store these locally on your laptop – you will need to refer their location in the Python script in the next section.

1. From the menu on the left you are also able to administer and create new topics.



Running a Kafka producer client

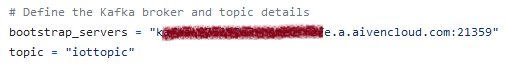
For the scope of this exercise, we will generate dummy IoT messages using a Python script that uses **confluent\_kafka** and **faker** packages. These messages will be sent to the **iottopic** of the Kafka service running in Aiven.

The full source code of the Python script is available in Github at <https://github.com/romeostoica/aiven-kafka>

To run the code, simply open the console on your laptop and type **python3 kafka-aiven.py**



Note: You will need to amend the URI of your Kafka service and the topic name. You will also need to amend the path to your local access key and certificate.

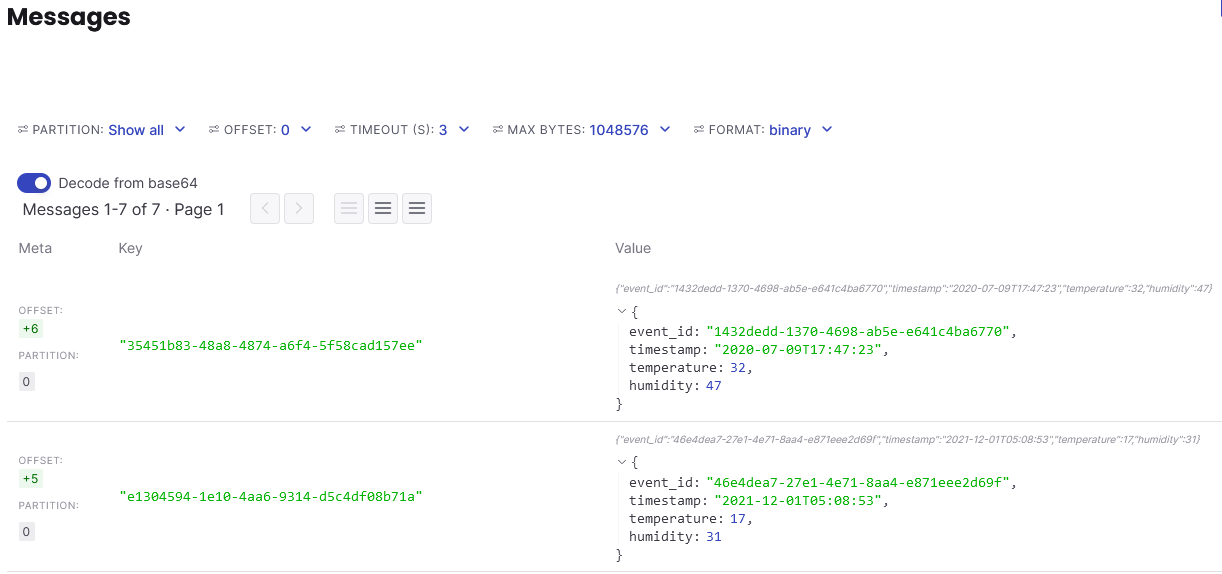
****

For each run, the Python script will generate a message in JSON format, which will be printed in the console and also be sent to the Kafka service you defined. The payload of the message will consist of dummy data (temperature and humidity) from a fictitious IoT sensor.

The test code has been tested on Python 3.6.9 and should be compatible with Python version 3.8 and newer.

Examining the IoT messages in Aiven

1. Back in the Aiven console, select **Topics** from the left side menu.
2. Click the topic you generated messages for.
3. Click the **Messages** button, then **Fetch messages** – you will see a list of all messages generated to that topic using the Python script



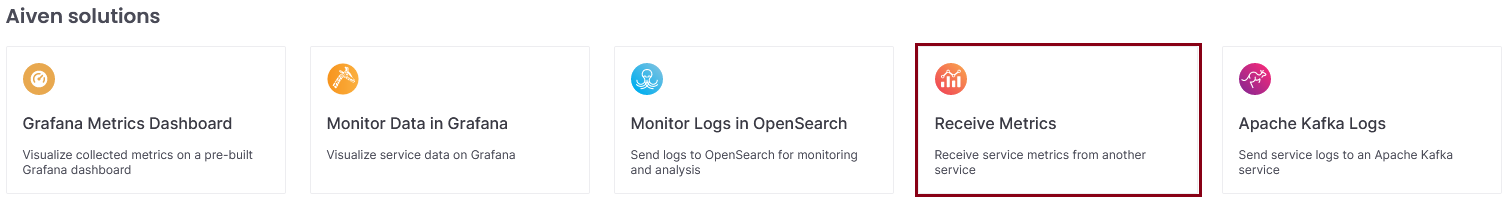
Setting up integrations for observability and monitoring

**InfluxDB** is a popular and lightweight time series database, typically used for storage and retrieval of time series data in fields such as operations monitoring, application metrics, IoT sensor data, and real-time analytics. With its lightning-fast storage and query capabilities, it transforms chaotic data streams into meaningful insights, empowering you to make informed decisions in real-time.

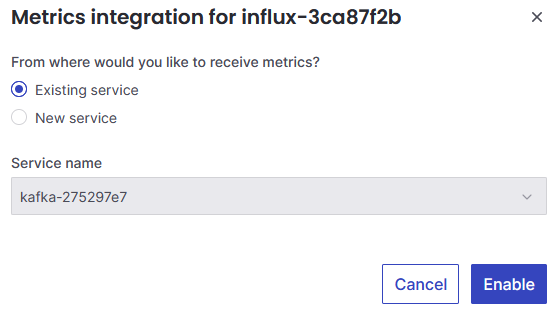
**Grafana** is a multi-platform open source analytics and interactive visualization web application. It provides charts, graphs, and alerts for the web when connected to supported data sources. It takes raw metrics and transforms them into stunning visualizations that provide at-a-glance insights.

Both InfluxDB and Grafana are available as fully-managed services on the Aiven platform.

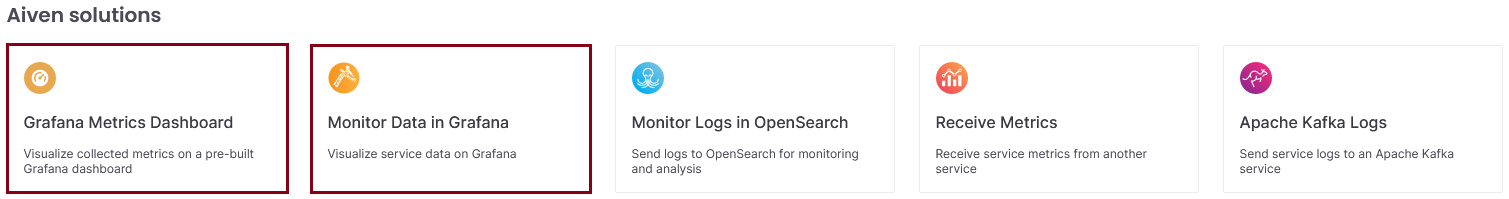
1. In the Aiven console, click again on the Create service button.
2. Similar to the previous steps to create a Kafka instance, provision an instance of **InfluxDB** and another of **Grafana**. It takes few minutes until the services are up and running.
3. To set up the integration between Kafka and InfluxDB – under your InfluxDB service, select Integrations from the left hand menu.
4. From the Aiven solutions listed, click on **Receive Metrics**.



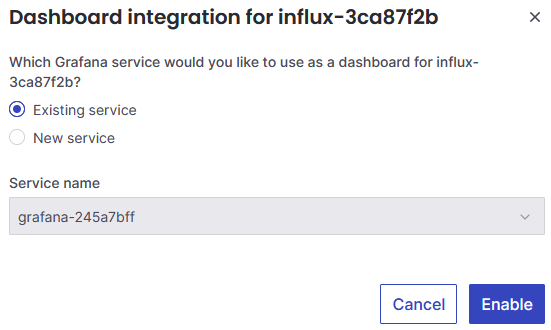
1. Select your existing Kafka service as the source for metrics in InfluxDB



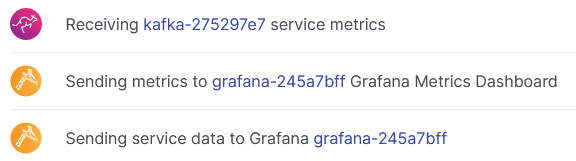
1. To set up the integration between InfluxDB and Grafana – under your InfluxDB service, select Integrations from the left hand menu.
2. From the Aiven solutions listed, click on **Grafana Metrics Dashboard** first, then do the same for **Monitor Data in Grafana**.



1. Select your newly-created Grafana service as the destination.



Upon the successful completion of the integration setup, you should see something similar to this on the Integrations homepage of your InfluxDB service.



Congratulations, you have just provisioned and configured your first Kafka-InfluxDB-Grafana setup on Aiven!