Managed Connectors

Create and configure a managed connector

# Objectives

Understand how to set up and configure a managed connector, in this case, a simple example of the Datagen Connector. Ensure the connector has the correct permissions to access just the topics it needs.

# Labs

## Setup

Ensure you have your cluster ready. If you followed the steps of the instructions of [Lab 3: ACLs and RBAC](https://docs.google.com/document/u/1/d/1DmuW9TcqhTosE-RmT0T_b94MWfrA4pxHH7Au6xv_0Lo/edit), your cluster is probably upgraded to a Standard cluster by now.

Install the **confluent** CLI if you have not done so already. See [references](#_agtdc8rgh3yv) for a link to the documentation on how.

## Datagen Connector with JSON Format

Install a simple connector using the Tutorial.

* Find and click on Connectors in the left-hand panel of your cluster overview.
* You should be presented with a list of Connector plugins. Find the “Sample Data” plugin, also known as the “Datagen Source” and click on the link.
* You are presented with a “Launch Sample Data” screen. Click on “Additional configuration” to fine-tune the settings.
* You first need to choose or create a topic on which to write to. Choose “Add new topic” and create a topic called “orders”, then continue.
* Pick “My Account” for now, then “Generate API Key & download”. Give the key a meaningful description.
* Pick JSON as the record format, and Orders as the template, then continue.
* Keep Connector Sizing at 1, then continue.
* Give the connector a meaningful name, such as “Order Datagen Connector”.
  + Inspect the JSON configuration. You might want to copy it somewhere locally for the final lab.
  + Investigate the advanced configurations and pricing.
* Launch the connector by pressing “Continue”.
* Note that you will receive an email and potentially a notification when the connector is running.
* You can either follow the tutorial or investigate the generated messages yourself under “Topics”.
* Point one of the clients you configured in [Lab 2: Clients](https://docs.google.com/document/u/1/d/1SKlDYu7DgxgPlAGMYvWO6kMcqcqeGWv9yNrMtZEauXY/edit) to the new topic and consume some messages on your local computer. You might have to adjust the ACLs to be able to read from the new topic. Reuse the API Key.

## AVRO Format and granular access

Add new data formats and granular access to a new connector.

* Create another Datagen Connector, this time with a new topic called “users” and a Service Account API Key option.
  + Choose a new Service Account named “user-generator”
  + Add all the required ACLs
* Pick AVRO for the format, and Users as the template
* Launch the connector
* Check out the messages in the Topics console
* Use kafka-avro-console-consumer or kcat (kafkacat) to consume the messages on your local computer. You will have to connect to the schema registry as well as to the cluster to be able to see the messages formatted correctly. Use the API Key from [Lab 2: Clients](https://docs.google.com/document/u/1/d/1SKlDYu7DgxgPlAGMYvWO6kMcqcqeGWv9yNrMtZEauXY/edit) as before.

## Using the **confluent** CLI to inspect the settings

Note, that words in **bold** are part of the command you will need to complete this lab.

* If you haven’t done so already, **login** to your Confluent Cloud organisation using your email address and password.
* **list** the existing environments, then **use** your current environment
* **list** the existing clusters, then **use** your current cluster
* Use **kafka acl list** to see the defined ACLs. The ACLs for the service user you created in the last part of the lab should be listed here.
* Use **connect cluster list** to see your connectors. You can use **describe** on a connector to see more details.

## Using the **confluent** CLI to create a connector

Create a connector using the CLI.

* If you copied your JSON file from the first part of this lab, you are in luck, otherwise, you have to either figure out the required settings yourself, or create another Connector up to the point where it shows you the JSON and copy that.
* We are going to use RBAC instead of ACLs. First, create a new Service User via the UI or the CLI, for example, “ConnectUser”.
* Assign RBAC roles to this user. Add two roles:
  + DeveloperWrite on a connector called “Stocks\_Datagen”
  + ResourceOwner on the topic “stocks”
* Create the topic “stocks” via the UI or the confluent CLI
* Create the connector “Stocks\_Datagen” from the confluent CLI using your JSON file. You will need the command  
  confluent connect cluster create --config-file <your-config-file>
* Check your connector is running in the UI and the confluent CLI.

Hint: if you want to check the role bindings of a Service User concerning resources like a Connector or a Topic, you need to add the –inclusive flag:  
  
confluent iam rbac role-binding list --principal "User:sa-wz2q1j" --inclusive

# References

<https://docs.confluent.io/confluent-cli/current/install.html>

<https://docs.confluent.io/cloud/current/access-management/access-control/rbac/manage-role-bindings.html>

<https://docs.confluent.io/cloud/current/connectors/cc-datagen-source.html>

# Expected Outcomes

Successfully deploy and configure a Managed Connector via the UI and the CLI.

Provide a connector with the correct permissions following the Least Privilege principle.

# Check your understanding

This colour marks advanced questions.

* What are the advantages of deploying a connector in the Confluent Cloud
  + What are the potential disadvantages?
* Why should you secure a connector with restricted access rather than open permissions?

# 