Audit Logs Part 2

Mirror audit logs into your dedicated cluster

# Objectives

Use cluster linking to bring the audit logs for your organization into your dedicated cluster. You can then use ksqlDB or other tools to shape, filter, and analyse the data.

# Labs

## Create the cluster link for your audit log

* Follow the steps in the [documentation](https://docs.confluent.io/cloud/current/multi-cloud/cluster-linking/audit-logs.html#use-cluster-linking-to-manage-audit-logs-on-ccloud) to set up a cluster link between the audit log cluster and your dedicated cluster. You can use the CLI as described in the documentation, or the Confluent Cloud UI to create the cluster link.
* Once the cluster link is established, consume some messages from the audit log from the UI or one of your clients set up in [Lab 2: Clients](https://docs.google.com/document/u/1/d/1SKlDYu7DgxgPlAGMYvWO6kMcqcqeGWv9yNrMtZEauXY/edit).

## 

## Use ksqlDB to shape the data

The reason why you might want to add your admin log into your dedicated cluster via cluster link is that you can filter and shape the events to your requirements.

* Create a ksqlDB cluster for your dedicated cluster.
  + Go to the side menu on the left-hand side of your cluster overview and create a small (1 CSU) cluster. It will take a few minutes to start up.
* The events are stored in a JSON format that needs to be interpreted first before you can run queries against it.

Here is some ksqlDB code you can use for that purpose:

SET 'auto.offset.reset' = 'earliest';

create stream audit\_source (

id varchar,

specversion varchar,

source varchar,

subject varchar,

type varchar,

time varchar,

datacontenttype varchar,

dataschema varchar,

data struct <

serviceName varchar,

resourceName varchar,

request STRUCT<

correlation\_id VARCHAR,

client\_id VARCHAR

>,

methodName varchar,

authenticationInfo struct <

principal varchar,

metadata struct <

mechanism varchar,

identifier varchar

>,

principalResourceId varchar,

identity varchar

>,

authorizationInfo STRUCT<

granted BOOLEAN,

operation VARCHAR,

resourceType VARCHAR,

resourceName VARCHAR,

patternType VARCHAR,

superUserAuthorization BOOLEAN

>,

requestMetadata STRUCT<

client\_address VARCHAR,

connect\_id varchar,

network\_id varchar

>,

result struct <

status varchar,

message varchar

>

>

) with (

KAFKA\_TOPIC='confluent-audit-log-events',

VALUE\_FORMAT='JSON',

TIMESTAMP = 'time',

TIMESTAMP\_FORMAT = 'yyyy-MM-dd''T''HH:mm:ss.SSSX'

);

create stream audit\_log with (KAFKA\_TOPIC='audit\_log', VALUE\_FORMAT='AVRO') as select \* from audit\_source;

You can now run some queries against the new Stream you created.

* Try a simple “select \* from audit\_log emit changes;”
* Something a little bit more sophisticated:  
  select DATA->METHODNAME, count(\*) COUNT from AUDIT\_LOG GROUP BY DATA->METHODNAME EMIT CHANGES;

# References

<https://docs.confluent.io/cloud/current/multi-cloud/cluster-linking/audit-logs.html#use-cluster-linking-to-manage-audit-logs-on-ccloud>

# Expected Outcomes

Create a working cluster link to mirror the audit log data into your dedicated cluster.

Use ksqlDB to “shape” the data.

# Check your understanding

This colour marks advanced questions.

* Why would you want to use a cluster link to bring your audit log data into your cluster?

# 