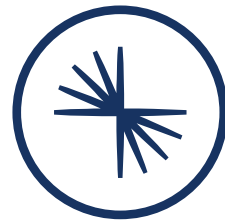


## 02: Access Confluent Cloud



# CONFLUENT Global Education

## Module Overview



This module contains five lessons:

- Confluent Cloud Console
- Confluent CLI
- Confluent Cloud APIs
- Kafka Client
- Confluent Cloud Security Basics

After this module you will be able to:

- Navigate in the Confluent Cloud UI to view topics, client metrics, data flows, schemas, connectors, etc.
- Use the Confluent Cloud CLI and REST API to perform operations
- Provide AuthN and AuthZ to Users and Services accounts

## 02a: Confluent Cloud Console - Demo

### Description

Your instructor will provide a demonstration of the Confluent Cloud console.


### Learning Objectives



Upon completion of this lesson and associated lab exercises, you will be able to:


- Demo Confluent Cloud Console including, where to find the different components, how to configure them, etc.


# Confluent Cloud Console - Demo

 CONFLUENT

## Welcome to Confluent Cloud

Log in with your email

 Sign in with Google

 Sign in with GitHub

Or

Next

Forgot password

Don't have an account? [Sign up and try it for free](#)

Copyright © Confluent, Inc. 2014-2022. [Privacy Policy](#) | [Terms & Conditions](#)  
Apache, Apache Kafka, Kafka, and associated open source project names  
are trademarks of the Apache Software Foundation



## 02b: Confluent CLI

### Description

Use the Confluent Cloud command line interface (CLI) to develop and manage a Confluent Cloud environment.

### Learning Objectives



Upon completion of this lesson and associated lab exercises, you will be able to:

- Define the Confluent CLI.
- Explain the installation options for Confluent Cloud.
- Identify important commands for Confluent Cloud users and administrators.

## Confluent CLI Overview

- The Confluent CLI enables developers to create, manage, and deploy their Confluent components.
- Confluent CLI traffic uses the following ports and endpoints:
  - Cloud operational requests are over HTTPS / port 443 and go to <https://confluent.cloud>.
  - Kafka protocol requests, for example, for produce/consume, go over port 9092 to Kafka brokers.
- Supported operating systems: macOS, Windows, and Linux.

## Confluent CLI Installation

- For **macOS** and **Linux**, run the following command:

```
brew install confluentinc/tap/cli
```

- For **Windows**:

1. Download the latest Windows ZIP file from:

```
https://github.com/confluentinc/cli/releases/latest
```

2. Unzip the following file:

```
confluent_X.X.X_windows_amd64.zip
```

3. Run `confluent.exe`

## Tarball or Zip Installation

1. Download and install the most recently released CLI binaries by platform:
2. Set the `PATH` environment to include the directory that you downloaded the CLI binaries in the previous step



## Important Commands for All Users

Command	Options	Description
<code>confluent help</code>	-	Get information about all available commands
<code>confluent login</code>	-	Login to your account
<code>confluent environment</code>	<code>list</code>	List all environments you have access to
	<code>use</code>	Set your current environment
<code>confluent kafka cluster</code>	<code>list</code>	List all clusters you have access to in the current environment
	<code>use</code>	Set your current cluster in the current environment

## Important Commands for Cloud Administrators

Command	Options	Description
<code>confluent iam user</code>	<code>delete</code> , <code>describe</code> , <code>invite</code> , <code>list</code>	Manage all the users within your organization
<code>confluent iam service-account</code>	<code>create</code> , <code>delete</code> , <code>list</code> , <code>update</code>	Manage CCloud Service Accounts
<code>confluent iam rbac role-binding</code>	<code>create</code> , <code>delete</code> , <code>list</code>	Manage role bindings for principals
<code>confluent api-key</code>	<code>create</code> , <code>delete</code> , <code>list</code> , <code>store</code> , <code>update</code> , <code>use</code>	Manage the API keys of all resources (cluster, SR, ksqlDB, Cloud metrics)
<code>confluent environment</code>	<code>create</code> , <code>delete</code> , <code>update</code>	Manage environments in your org.
<code>confluent kafka cluster</code>	<code>create</code> , <code>delete</code> , <code>describe</code> , <code>update</code>	Manage clusters in your current environment
<code>confluent kafka acl</code>	<code>create</code> , <code>delete</code> , <code>list</code>	Manage ACLs for your Service Accounts

## 02c: Confluent Cloud APIs

### Description

Use Confluent Cloud application programming interfaces (APIs) to programmatically extend and integrate.

### Learning Objectives



Upon completion of this lesson and associated lab exercises, you will be able to:

- Manage your own Confluent Cloud account or to integrate Confluent into your product:
  - Create a topic.
  - Update the topic configuration.
  - List cluster configuration.
  - List consumer groups and consumer lags.

## API Requests

- API requests **must** be made over HTTPS
- API requests without authentication will fail
  - Authentication is included in Headers as an `Authorization: Basic {key}`
  - `{key}` requires you to base64 encode `<api-key>:<api-secret>`:

```
$ echo -n "<api-key>:<api-secret>" | base64
```

## Create a Topic

Create a new topic `testTopic1`:

```
curl -H "Authorization: Basic ABC123ABC" -H 'Content-Type: application/json' \  
  --request POST \  
  --url 'https://pkc-lzvrd.us-west4.gcp.confluent.cloud:443/kafka/v3/clusters/lkc-vo9pz/topics' \  
  -d '{"topic_name": "testTopic1", "partitions_count": 5, "replication_factor": 3}'
```

## List All Topics

```
curl -H "Authorization: Basic ABC123ABC" --request GET \  
      --url 'https://pkc-lzvr4.us-west4.gcp.confluent.cloud:443/kafka/v3/clusters/lkc-vo9pz/topics'
```

## 02d: Kafka Client

### Description

Use the Confluent Cloud client to programmatically develop and manage Confluent Cloud environments and clusters.

### Learning Objectives



Upon completion of this lesson and associated lab exercises, you will be able to:

- Define the required properties to connect a Kafka client to Confluent Cloud.

## Connect Clients to Confluent Cloud

- Set the configuration in the client code:

```
bootstrap.servers=pkc-ep9mm.us-east-2.aws.confluent.cloud:9092
security.protocol=SASL_SSL
sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule   required username='{{
CLUSTER_API_KEY }}' password='{{ CLUSTER_API_SECRET }}';
sasl.mechanism=PLAIN
```

- If using Schema Registry in Confluent Cloud:

```
schema.registry.url=https://psrc-4nrnd.us-central1.gcp.confluent.cloud
basic.auth.credentials.source=USER_INFO
basic.auth.user.info={{ SR_API_KEY }}:{{ SR_API_SECRET }}
```

- Confluent Cloud Console provides the required configuration for your programming language:
  - In the Cluster → **Clients** → **New client**



## 02e: Confluent Cloud Security Basics

### Description

Understand the basic fundamentals of securing a Confluent Cloud environment and users.

### Learning Objectives



Upon completion of this lesson and associated lab exercises, you will be able to:

- Explain how to control access to Confluent Cloud.
- Define API keys with concrete examples.
- Explain role-based access control (RBAC) and access control lists (ACLs).
- Describe best practices for setting access control.

## API Keys

API keys control access to Confluent Cloud components and resources.

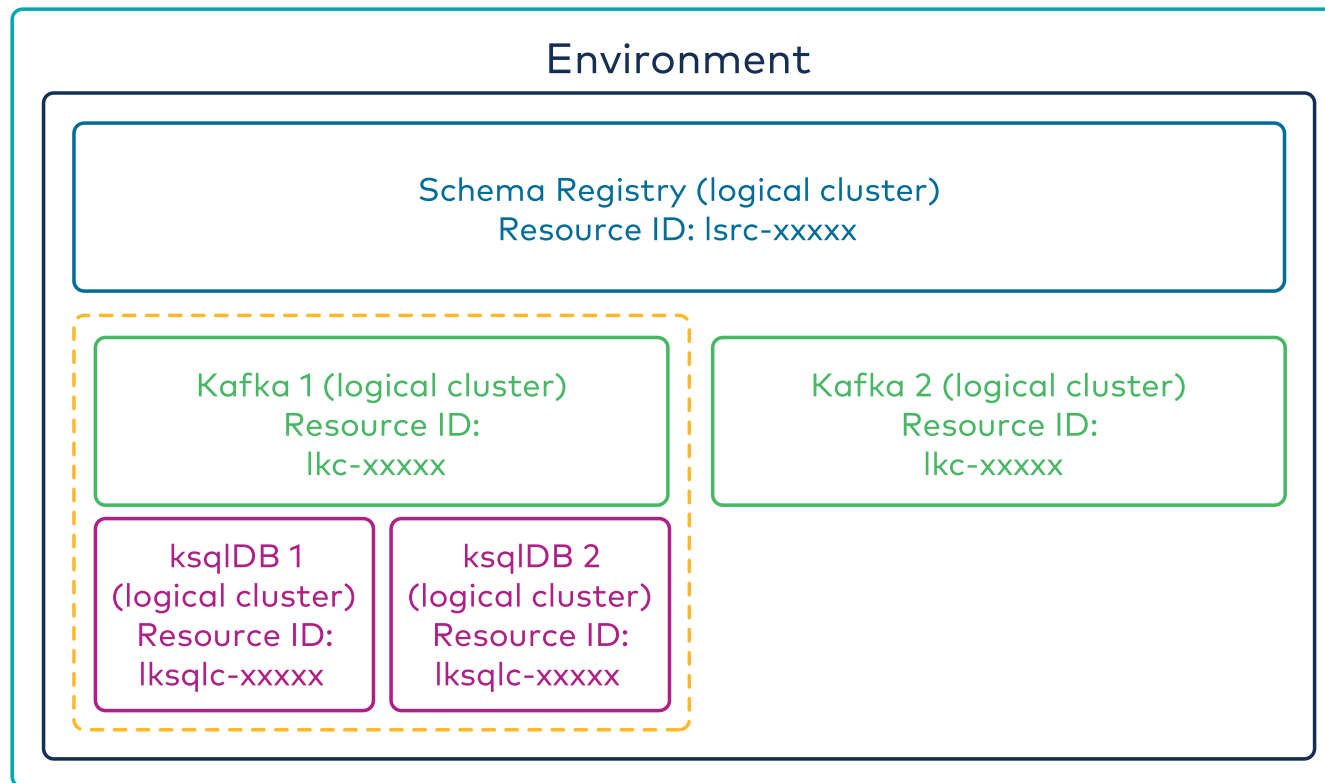
Each API key consists of a key and a secret (username and password).

There are four types of API keys:

- Kafka API keys
- Schema Registry API keys
- ksqlDB API keys
- Cloud API keys

## API Keys - Cluster Overview

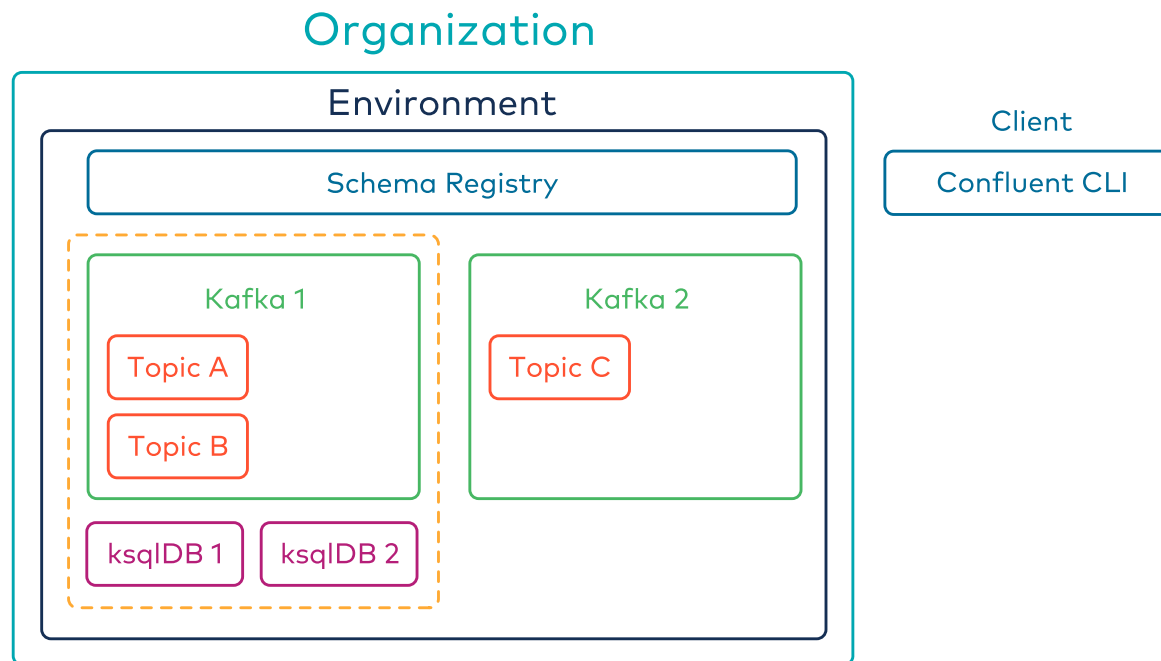
### Organization



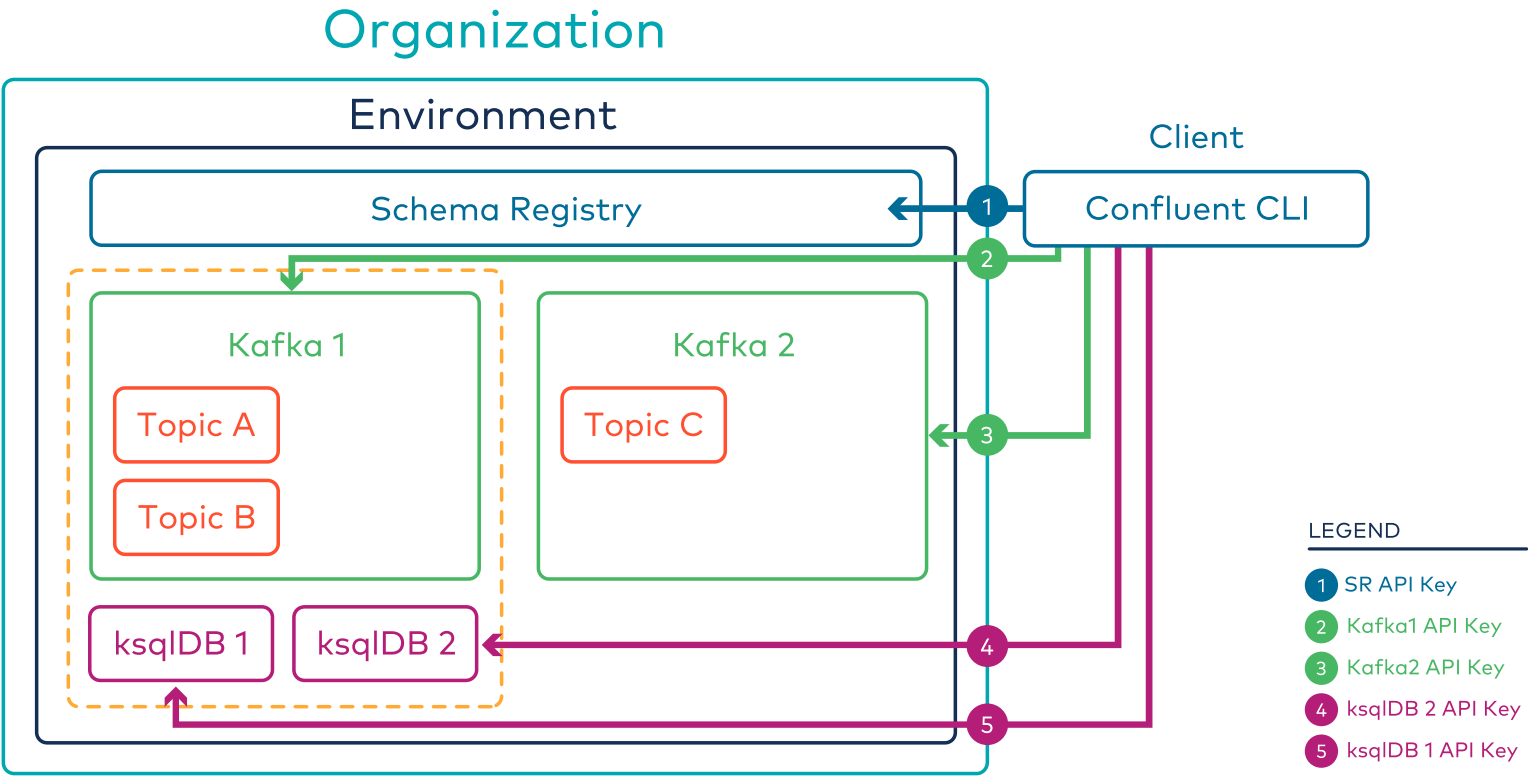
## API Keys - Confluent CLI Client (1)

### Question:

What API keys will be required if the Confluent CLI needs to register a schema to Schema Registry, produce to **Topic A** and **Topic C** and send queries to the two ksqlDB clusters?



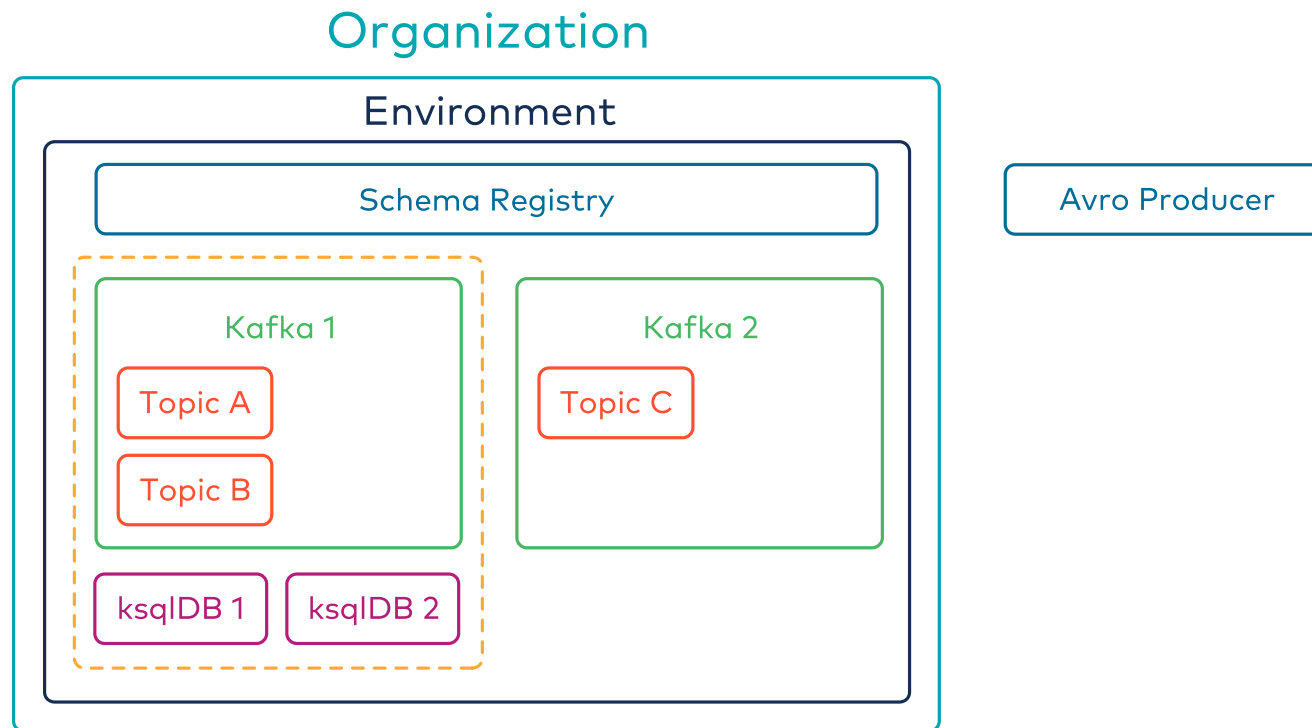
# API Keys - Confluent CLI Client (2)



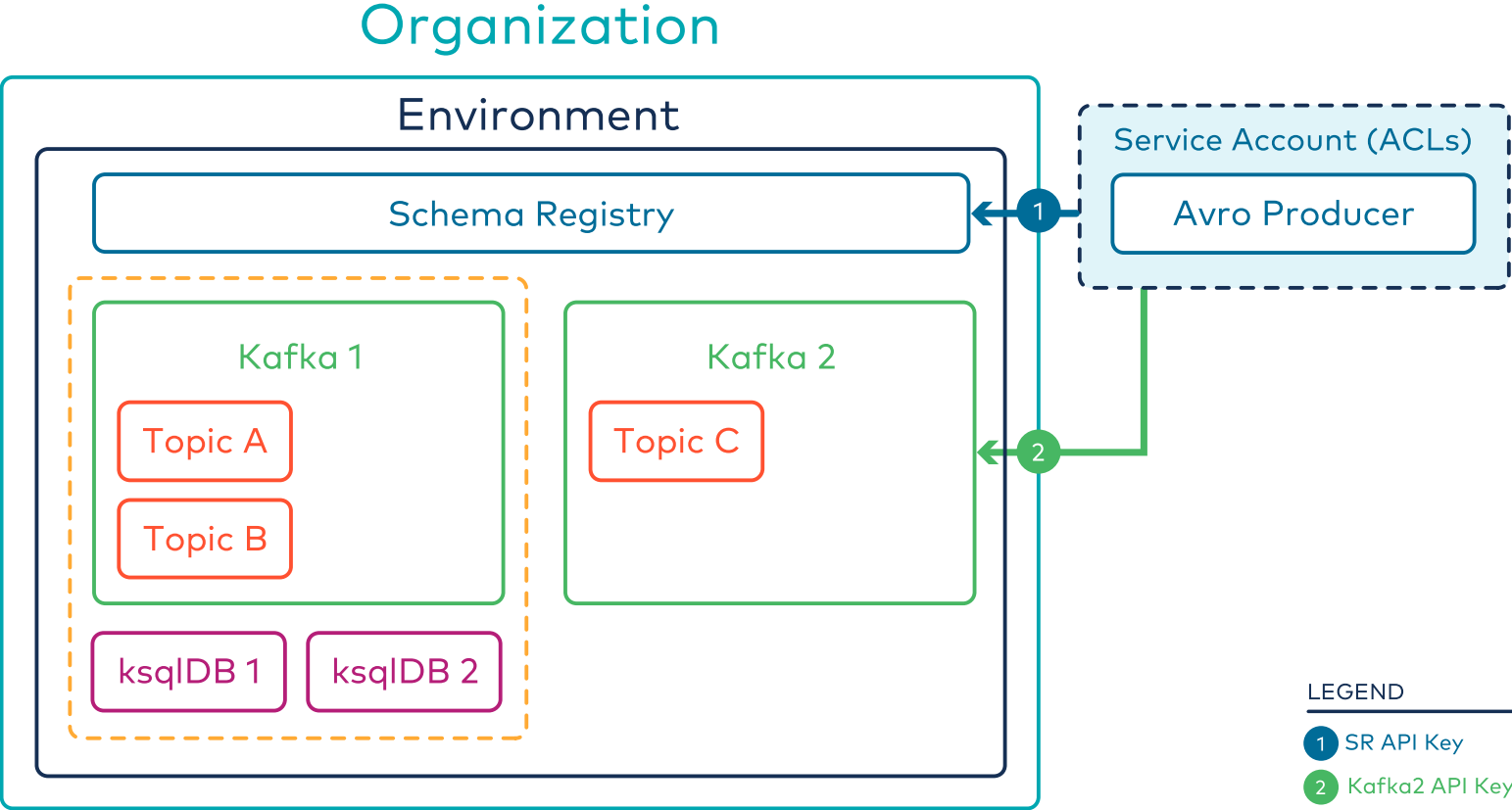
## API Keys - Kafka App (1)

### Question:

What API keys will be required if the Avro Producer needs to write messages to **Topic C**?



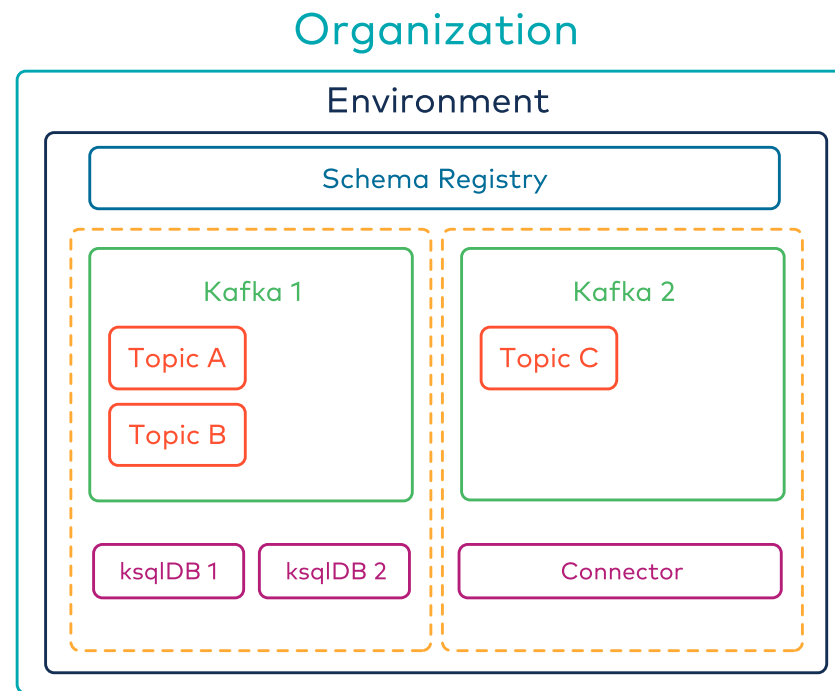
# API Keys - Kafka App (2)



## API Keys - Connector (1)

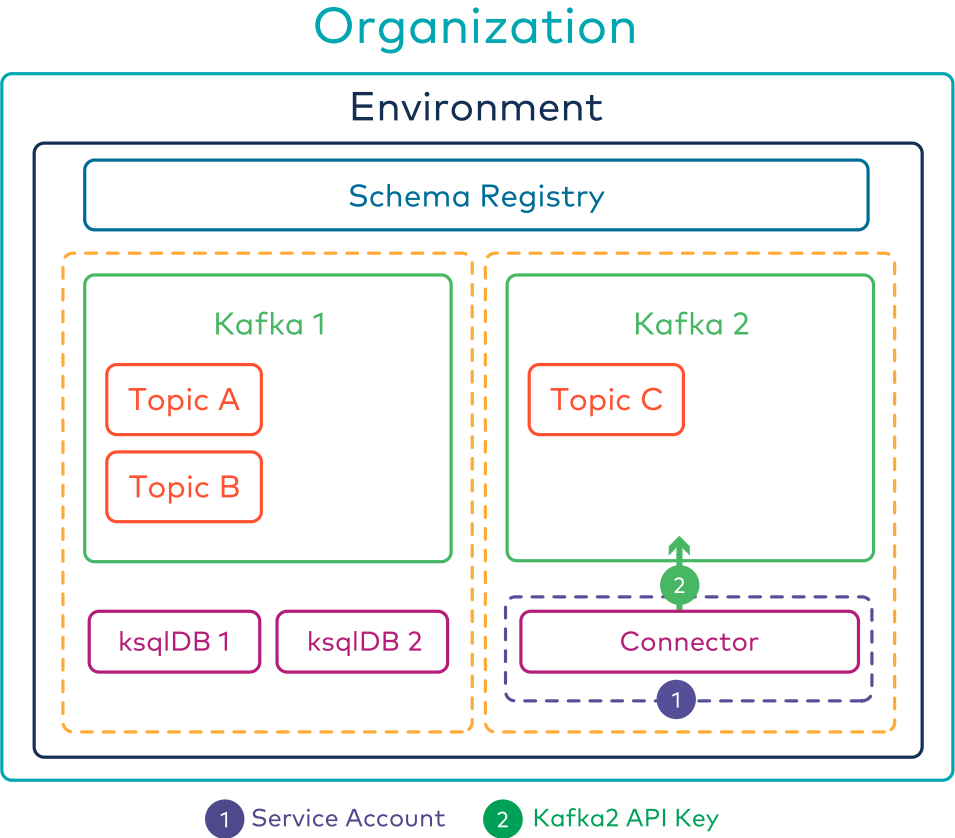
### Question:

You want to create a **Connector** associated to **Kafka 2** to write to **Topic C**. What API key/s will be required?





# API Keys - Connector (2)

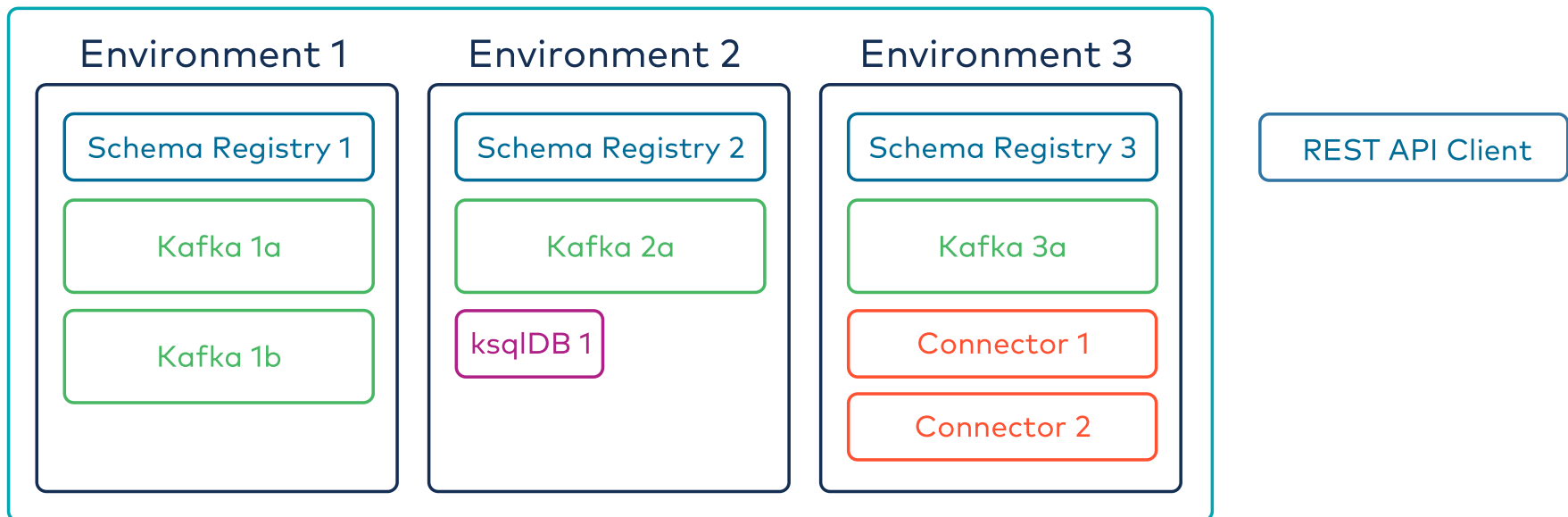


## API Keys - Cloud API Key (1)

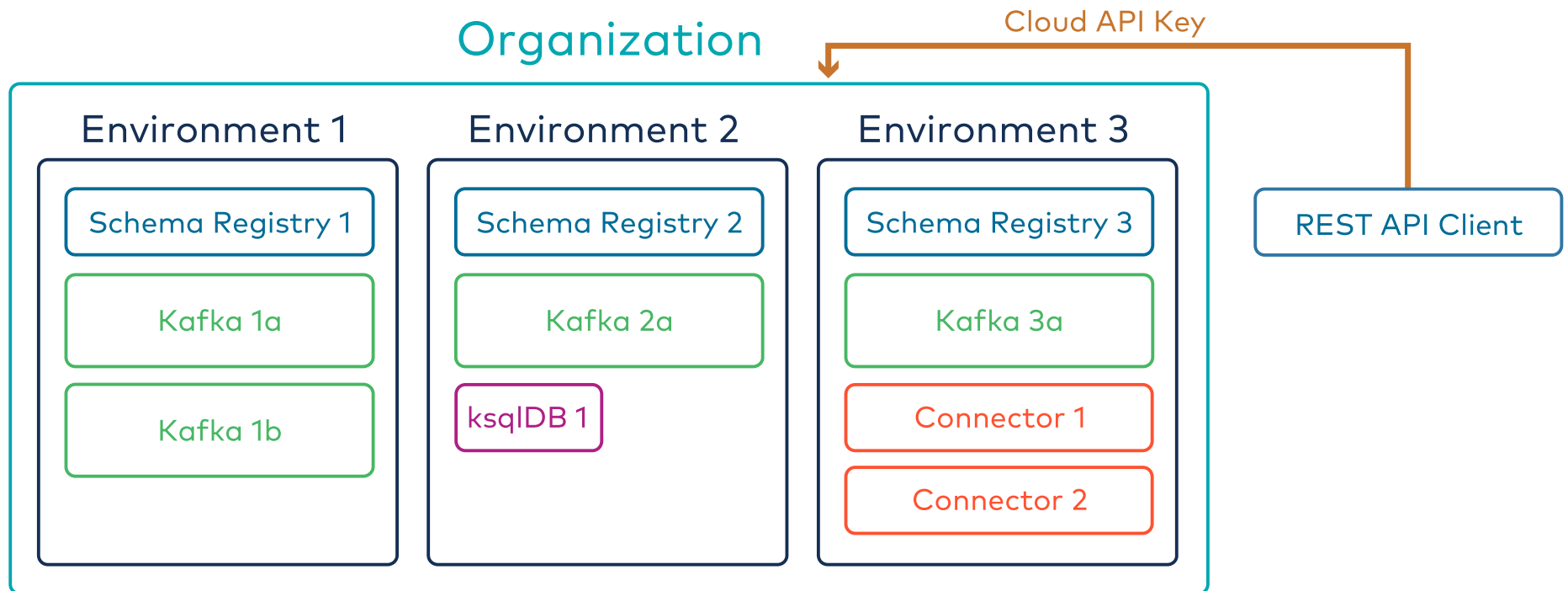
### Question:

You have an Application that wants to monitor the Confluent Cloud performance using the Metrics API. What API key/s will be required?

### Organization



## API Keys - Cloud API Key (2)

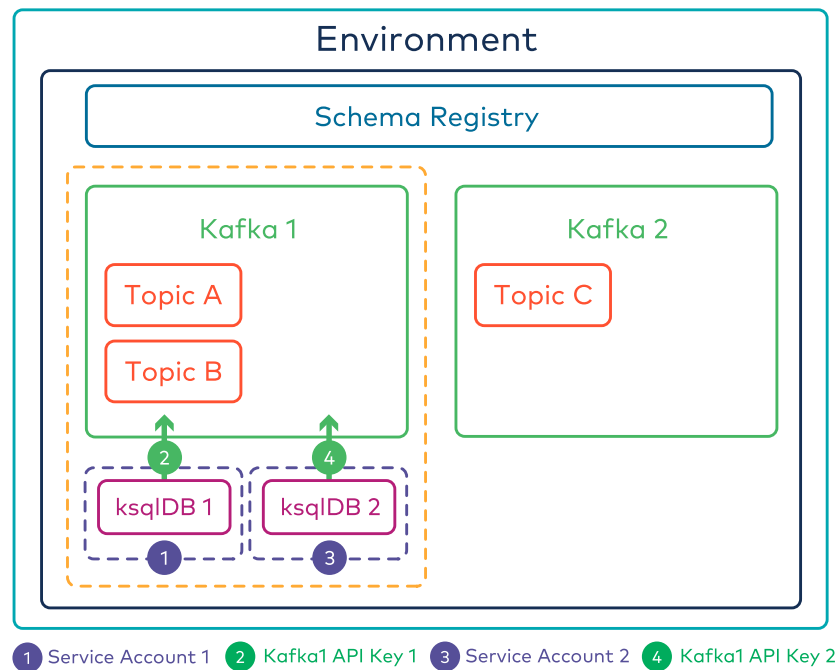


## API Keys - ksqlDB clusters to Kafka cluster

### Scenario:

You have two ksqlDB clusters associated with **Kafka 1**. These ksqlDB clusters need access to **Kafka 1** to consume the input data and to produce the output of the transformations.

### Organization



## Managing API Keys

All API keys can be managed using the Confluent CLI:

```
$ confluent api-key list
$ confluent api-key create --resource lksqlc-xxxxx --service-account sa-xxxxx
$ confluent api-key delete ESP40MLMFQRNEXOC
$ confluent api-key list --service-account sa-xxxxx
```



All API keys can be managed using Confluent Cloud Console, except ksqlDB API keys.

## Best Practices for Using API Keys

- Ensure only service account API keys are used in production
- User account API keys recommended only for development and testing
- Delete unneeded API keys and service accounts
- Rotate API keys regularly:
  1. Create a new API key
  2. Update the resource or application to use the new API key
  3. Delete the old API key

## Role-Based Access Control (RBAC)

Access Control for **User** and **Service** accounts based on predefined roles

- Can control access to:
  - Organization
  - Environment
  - Kafka cluster
  - Schema Registry
  - ksqlDB cluster
  - Pipelines
  - Network
- There are 15 predefined roles:
  - `OrganizationAdmin`
  - `EnvironmentAdmin`
  - `CloudClusterAdmin`
  - `Operator`
  - `KsqlAdmin`
  - `NetworkAdmin`
  - `MetricsViewer`
  - and more...

## Access Control Lists (ACLs)

- Provide secure access to your Confluent Cloud resources and data
- Can be applied to **User** and **Service** accounts
- Prefix matching supported

Cluster	Consumer group	Topic	Transactional ID
Topic name* vehicle-sensors			
Pattern type* LITERAL			
Operation* WRITE		Permission* ALLOW	
<a href="#">Delete ACL</a>			



## Lab Module 02: Accessing Confluent Cloud

Please work on **Lab Module 02: Accessing Confluent Cloud**.

Refer to the Exercise Guide.

