Kafka

Comprehensive Guide: Configuring Confluent Cloud Identity Pool with Azure AD Integration

Junaid Ahmed



# Introduction

This comprehensive guide will take you through the step-by-step process of configuring Confluent Cloud Identity Pool with Azure AD integration. The guide includes both Azure AD and Confluent Cloud configurations to achieve fine-grained access control to Kafka topics based on users' Azure AD group affiliations.

## Prerequisites

Azure AD Setup:

* Register your applications in Azure AD.
* Create Azure AD groups and assign applications to these groups.

Confluent Cloud Account:

* Sign up for a Confluent Cloud account.

**Azure AD Configuration**

1. Log in to Azure AD Portal
   * 1. Log in to the Azure AD portal using your Azure AD administrator credentials.
2. Register Applications
3. Register your Kafka producer and consumer applications:
4. Navigate to "Azure Active Directory" > "App registrations."
5. Click on "New registration" and provide necessary details.
6. Note down the "Application (client) ID", "Application (secret) ID" and "Directory (tenant) ID" for each application.
7. Configure Token Issuance
8. Configure Azure AD to include group claims in the JWT token:
9. In the application registration, go to "Token configuration."
10. Add the necessary group claims, e.g., "groups": ["Group1"].
11. Save the changes.
12. Create Groups and Assign Applications
13. Create Azure AD groups and assign applications to these groups:
14. In the Azure AD portal, navigate to "Groups."
15. Create groups, e.g., "Group1," "Group2," "Group3."
16. Assign the corresponding applications to their respective groups.

**Confluent Cloud Configuration**

* + - 1. Log in to Confluent Cloud

1. Log in to your Confluent Cloud account using your credentials.
   * + 1. Navigate to Access Section

In the Confluent Cloud Console, navigate to the "Access" section. This section is where you manage Identity Pools and Access Control Lists (ACLs).

* + - 1. Create Identity Pool
  1. Create a new Identity Pool:
  2. Click on "Identity Pools" in the "Access" section.
  3. Click on the "Create Identity Pool" button.
  4. Provide a name for the Identity Pool (e.g., "AzureAD-Identity-Pool").
  5. Configure other settings as needed.
     + 1. Configure Azure AD Integration
     1. Configure Azure AD integration for the Identity Pool:
     2. Within the Identity Pool configuration, look for Azure AD integration settings.
     3. Provide Azure AD details such as the tenant ID, client ID, and client secret.
     4. Establish the trust relationship between Azure AD and Confluent Cloud.
        1. Define Role Mapping
     5. Define role mappings based on Azure AD group claims:
     6. In the Identity Pool configuration, locate the "Role Mapping" or similar section.
     7. Add role mappings for different Azure AD groups.
     8. Example:
        1. Map members of Azure AD Group1 to the "Read-Only" role for Topic1.
        2. Map members of Azure AD Group2 to the "Write" role for Topic2.
        3. Map members of Azure AD Group3 to both "Read" and "Write" roles for Topic2.
        4. Create ACLs for Topics

1. Configure Access Control Lists (ACLs) for your Kafka topics based on the roles you defined:
2. In the Confluent Cloud Console, navigate to the "Topics" section.
3. For each topic, set ACLs to allow or deny access based on the roles configured in the Identity Pool.
4. Save Configuration
5. Save your Identity Pool and ACL configurations.
   * + 1. Test Access Control

Test the access control configuration by having users from different Azure AD groups authenticate with Confluent Cloud. Confirm that the assigned roles match the configured role mappings.

Conclusion

By following these comprehensive steps in both Azure AD and Confluent Cloud, you have successfully configured Confluent Cloud Identity Pool with Azure AD integration. This setup enables secure and controlled access to Kafka topics based on Azure AD group memberships.

Note

Always refer to the latest documentation for Azure AD and Confluent Cloud for any updates or additional features.