Kamlesh Kumar Pant

USeReady  Bangalore-Karnataka

Sales-force Streaming Events and Kafka Integration

Contents

[**Solution Architecture** 2](#_Toc88140656)

[**Solution Design/Approach** 2](#_Toc88140657)

[Salesforce: 2](#_Toc88140658)

[Zookeeper and Kafka Cluster 3](#_Toc88140659)

[Application (Python) 3](#_Toc88140660)

SALESFORCE -CDC & KAFKA CONNECTOR

This solution is primarily intend for Citibank , where Citi bank requires all the changes related to users -any addition/modification or role changes etc should be reflected in real time to consuming applications.

To Achieve this, application capitalizing salesforce event bus mechanism to capture the CDC Data.

# **Solution Architecture**

Diagram

Description automatically generated

# **Solution Design/Approach**

To capture CDC data and storing or redirecting them to HDFS/Databases (Mysql) or Kafka Topics, application (Python) needs to capitalize Salesforce Streaming Events service for capturing real time changing data via SF event bus.

Following steps to be executed across platforms

## Salesforce:

1. Create a connected app in Salesforce
2. Enable Oauth Process in the newly created app
3. Find the Org Domain URL for application interaction and update the application configuration file for corresponding key/values
4. Record Consumer Key, Consumer Secret and update the application configuration file for corresponding key/values
5. Record Refresh Token using Postman and Salesforce [OAuth 2.0 Web Server Flow for Web App Integration (salesforce.com)](https://help.salesforce.com/s/articleView?id=sf.remoteaccess_oauth_web_server_flow.htm&type=5) and update the application configuration file for corresponding key/values

## Zookeeper and Kafka Cluster

1. Download Kafka application
2. Download GitBash.
3. Enable Zookeeper
4. Enable Kafka Cluster
5. Create Kafka Topic and update the application configuration file for corresponding key/values

## Application (Python)

1. Change Configuration key/value details in the configuration File as per the need

|  |
| --- |
| "SALESFORCE": {  "DomainURL": "",  "AUTHENTICATION": {  "AuthURL": "https://login.salesforce.com/services/oauth2/token",  "UserName": "",  "Password": "",  "GrantType": "password",  "ClientID": "",  "ClientSecret": "",  "RefreshToken":""  },  "GETAPI": {  "GetAccountDetail": {  "EndPointURL": "<>/services/data/v52.0/query/?q=",  "QueryOrHeaderParameters": "SELECT+Name,Type+FROM+Account"  },  "GetUserDetail": {  "EndPointURL": <>/services/data/v52.0/query/?q=",  "QueryOrHeaderParameters": "SELECT+Name,Email,UserRoleId,ProfileId,Phone+FROM+User "  },  "GetUserProfile": {  "EndPointURL": "<>/services/data/v52.0/sobjects/Profile/",  "QueryOrHeaderParameters": ""  },  "GetUserRole": {  "EndPointURL": "<>/services/data/v52.0/sobjects/UserRole/",  "QueryOrHeaderParameters": ""  },  "SOBJECTS\_DETAILED\_DATA": {  "EndPointURL": "<>/services/data/v52.0/sobjects/",  "SOBJECTS": [  "Profile",  "UserRole",  "Contact1",  "User"  ]  }  },  "POSTAPI": {  "PostAccountDetail": {  "EndPointURL": "",  "HeaderParameters": "",  "BodyParameters": ""  }  },  "STREAMAPI": {  "SALESFORCE\_PUSHTOPICS": [  "TCRMAccount",  "TCRMContact"  ]  }  } |

1. Python application will connect Salesforce using python salesforce streaming service library, application can connect Salesforce Server using refresh\_token authentication or plain username/password authentication
2. Application will work in asynchronous mode and whenever there is event in the salesforce event bus, application will capture that data and parse it and identify which object change it is i.e., User data or account data or contact data or any object in particular accordingly it will format the data in json format and sends to Kafka Topic or MySQL server or it can be any target HDFS/AWS-S3 etc.