# Salesforce Pub/Sub Adapter for SAP Integration Suite

Version 1.3.0 – May 2026

THE BEST RUN SAP

# Contents


1 Salesforce Pub/Sub Introduction ..... 4

1.1 Objective ..... 4

1.2 Coding Samples ..... 4

1.3 Internet Hyperlinks ..... 4

1.4 Overview ..... 4

1.5 Features ..... 5

2 Installation and Configuration ..... 6

2.1 Prerequisites ..... 6

2.2 Procedure ..... 6

2.2.1 Adapter Installation by creating a New Integration Flow ..... 6

2.2.2 Adapter Installation without Creating a New Integration Flow ..... 7

3 Getting Started: Salesforce Pub/Sub Adapter ..... 9

3.1 Architecture Overview ..... 9

3.2 Authentication ..... 10

3.2.1 Creating Credentials in Security Material ..... 10

3.2.2 Creating User Credentials ..... 10

3.2.3 Creating Secure Parameter ..... 11

3.2.4 Adding Key Pair to Keystore ..... 12

4 Salesforce Pub/Sub Adapter Configuration ..... 13

4.1 Sender Adapter ..... 13

4.1.1 General ..... 13

4.1.2 Connection ..... 14

4.1.3 Processing ..... 16

4.2 Receiver Adapter ..... 18

4.2.1 General ..... 18

4.2.2 Connection ..... 19

4.2.3 Processing ..... 21

5 Operations ..... 22

5.1 Sender Adapter ..... 22

5.2 Receiver Adapter ..... 23

5.2.1 Get Schema ..... 23

5.2.2 Publish ..... 24

6 Reference ..... 26
6.1 Initialize OAuth Username-Password in Salesforce ..... 26
6.2 Initialize OAuth JWT Bearer in Salesforce ..... 29
6.2.1 Create a Connected App in Salesforce ..... 29
6.2.2 Create Security Certificate and JKS file using OpenSSL ..... 31
6.3 Creating OAuth Client Credentials ..... 32
6.3.1 Creating OAuth2 Client Credentials in Security Material ..... 32

# 1 Salesforce Pub/Sub Introduction

## 1.1 Objective

This is the official guide for the Salesforce Pub/Sub Adapter for SAP Integration Suite. This guide covers all relevant information for integration developers to start working with the Salesforce Pub/Sub adapter. Read this guide carefully before using the Adapter.

## 1.2 Coding Samples

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. The correctness and completeness of the Code given herein are not guaranteed.

## 1.3 Internet Hyperlinks

The documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. The availability and the correctness of this related information or the ability of this information to serve a particular purpose are not warranted.

## 1.4 Overview

Salesforce is a SaaS Cloud solution that covers four main services; Marketing, Sales, Orders, and Support. The Salesforce Pub/Sub API provides a single interface for publishing and subscribing to platform events, including real-time event monitoring events, and change data capture events.

Salesforce PubSub adapter allows establishing a connection with the Salesforce Pub/Sub API to allow subscription to and publishing of events.

## 1.5 Features

The Salesforce Pub/Sub Adapter provides the following key features:

- Allows you to **subscribe** to the channels to receive events.
- Supports **publishing** of events to channels with defined schemas.
- Retrieves schema from the Salesforce channel using the **Get Schema** operation.
- Offers secure authentication options using **OAuth 2.0 Username-Password**, **OAuth 2.0 JWT Bearer**, and **OAuth 2.0 Client Credentials**.
- Using the Sender Adapter, you can retrieve events from a **specific position (Custom), earliest position (Earliest)**, or **latest position (Latest)**.
- Provides an error-handling mechanism in case of subscription failure, allowing you to re-subscribe to the channel and retrieve events from the **earliest position (Earliest)** or **latest position (Latest).**

# 2 Installation and Configuration

This section describes the prerequisites and procedure to install the Salesforce Pub/Sub adapter.

## 2.1 Prerequisites

> The Salesforce Pub/Sub adapter is available as part of your Standard license for SAP Integration Suite. For more information, see SAP Note.

Before you start working with the adapter, you must deploy it to your SAP Integration Suite tenant.

## 2.2 Procedure

You can deploy the adapter using the following methods:

> The installation procedure below is compatible with Apache Camel 2, Apache Camel 3, and the Edge Integration Cell (EIC) platform.

### 2.2.1 Adapter Installation by creating a New Integration Flow

The Salesforce Pub/Sub adapter is available for selection in the sender/receiver adapter list and can be deployed in the **Design** tab directly as you use it in an Integration flow.

**Purpose**

To install an adapter for use in your Integration flow.

**Procedure**

1. Go to **Design** workspace and select the integration package where you want to create a new Integration flow.
2. Click **Edit** to make the package editable.
3. Go to the **Artifacts** tab. Click **Add** and select **Integration Flow**.
4. Enter **Name** and **ID** for your flow. Additionally, select **Runtime Profile** from the drop-down and choose **Sender** and **Receiver** systems from the list . Finally, click **Add** to create the integration flow.
5. Go to the newly created integration flow and click **Edit** to make it editable.

i. For the Sender, in the integration flow add a **Connector** between the **Sender box** and the **Start**.
ii. For the Receiver, in the integration flow, click **End** to add a **Connector** between the **End** and the **Receiver Box**.

6. A drop-down with the available adapters appears. The **SalesforcePubSub** adapter should show up in the list.
7. Select the **SalesforcePubSub** adapter from the list. The adapter is now imported which *triggers* an adapter deployment. Once the Salesforce Pub/Sub Adapter is deployed, a success message is displayed.
   After the above steps are done, the Salesforce Pub/Sub Adapter is successfully deployed in your Design workspace of the SAP Integration Suite tenant.

## 2.2.2 Adapter Installation without Creating a New Integration Flow

The following procedure explains how the Salesforce Pub/Sub adapter is migrated from the Discover workspace to the Design workspace of the SAP Integration tenant.

This method is useful for scenarios where integration flow packages are migrated from development to a higher environment such as Production.

The Salesforce Pub/Sub adapter can be imported into the Design workspace without creating an integration flow. Use the Transport Management Service (TMS) to import/transport the Salesforce Pub/Sub adapter to a higher environment. Alternatively, If the TMS is not available in the landscape, the adapter package can be imported to the Design workspace by copying it from the Discover workspace.

To copy the integration package from the Discover workspace and import the Salesforce Pub/Sub adapter to the Design workspace, follow these steps:

1. Go to **Discover** workspace.
2. In the search box, search for **Salesforce PubSub adapter for SAP Integration Suite package**.
3. Select the package and click **Copy**. This copies the package from the Discover workspace to the Design workspace.
4. Go to Design workspace and select the copied **Salesforce PubSub adapter for SAP Integration Suite package**.
5. In the Actions tab of the selected package, click Deploy. This completes the adapter deployment to the Design workspace.

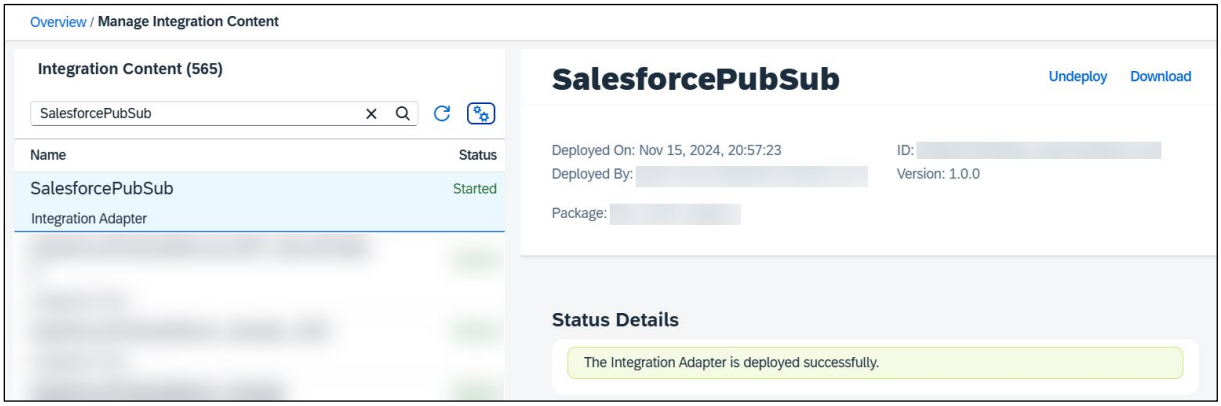
After the adapter deployment is complete, you can check the status in the **Monitor** section.

## Purpose

To check the status of the deployed adapter:

## Procedure

1. Under the **Monitor** tab, click **Integrations and APIs**. This opens the **Overview** page.
2. On the **Overview** page, go to **Manage Integration Content** section and click **All**. This opens **Integration Content** page with a list of all the deployed adapters.
3. Here, you can check and confirm the deployment status of your adapter.



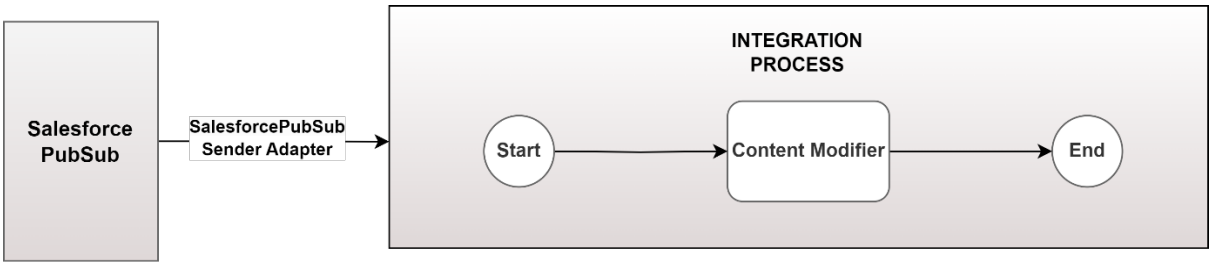
# 3 Getting Started: Salesforce Pub/Sub Adapter

This section explains how to configure the Salesforce Pub/Sub adapter for SAP Cloud Integration. You can find information about adapter architecture, application configuration, and authentication for the Salesforce Pub/Sub Adapter.

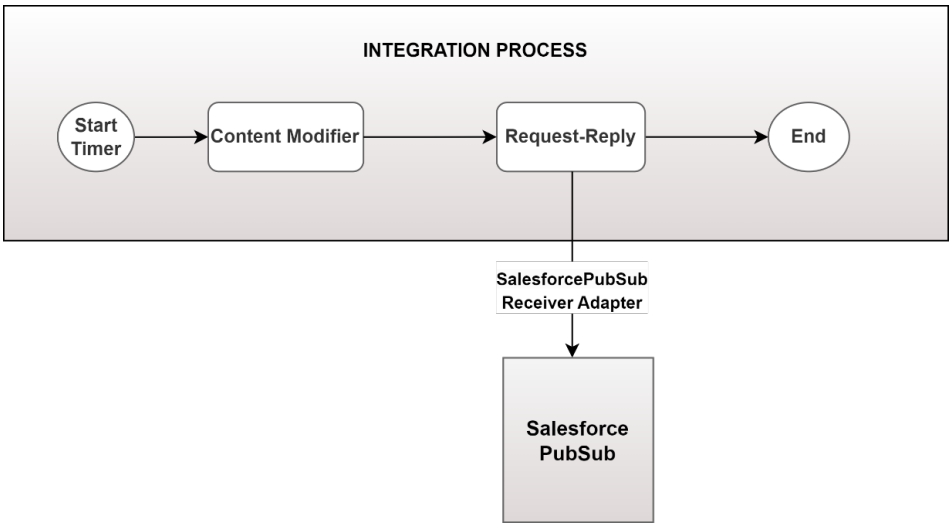
## 3.1 Architecture Overview

The Salesforce Pub/Sub adapter is designed to function as both a sender and receiver adapter.

**How the Salesforce Pub/Sub Sender Adapter Works:** The Salesforce Pub/Sub sender adapter supports subscribing to a topic and reading events. In such a scenario where the adapter is used as a sender adapter, Salesforce Pub/Sub acts as the initiator of the calls.




**How the Salesforce Pub/Sub Receiver Adapter Works:** The Salesforce Pub/Sub receiver adapter supports retrieving of schema and publishing the events. In such a scenario where the adapter is used as a receiver adapter, SAP Integration Suite acts as the initiator of the calls.



# 3.2 Authentication

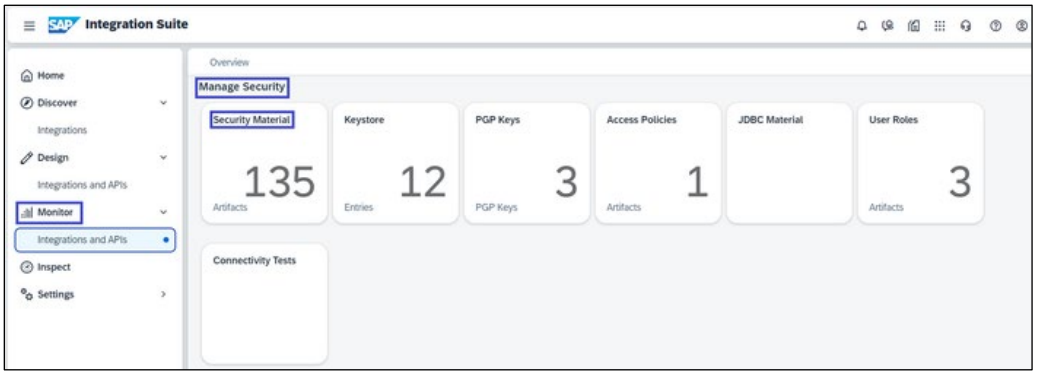
The Salesforce Pub/Sub Adapter supports **OAuth 2.0 Username-Password**, **OAuth 2.0 JWT Bearer, OAuth 2.0 Client Credentials** authentication mechanisms. The Adapter makes use of common security artifacts in SAP Cloud Integration. It is required to use Secure Parameter and User Credentials to safely store OAuth details, user-password combinations, and Key Pair for JWT. These security artifacts can then be accessed in the Adapter using aliases.

## 3.2.1 Creating Credentials in Security Material

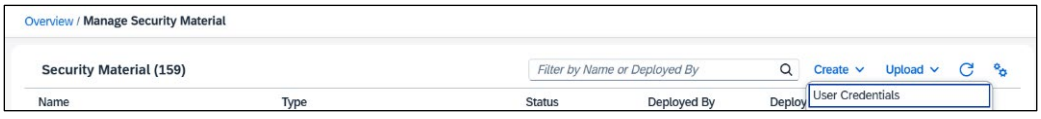
As there are three types of authentication methods. Multiple security artifacts need to be created for each type as follows:

## 3.2.2 Creating User Credentials

1. In SAP Integration Suite, navigate to **Monitor** > **Integrations and APIs**. This opens the **Overview** page.
2. On the **Overview** page, go to **Manage Security** section and click **Security Material**.




3. On **Manage Security Material** page, click **Create** to select **User Credentials** from the dropdown.




4. In the **Create User Credentials** popup, provide the below details.

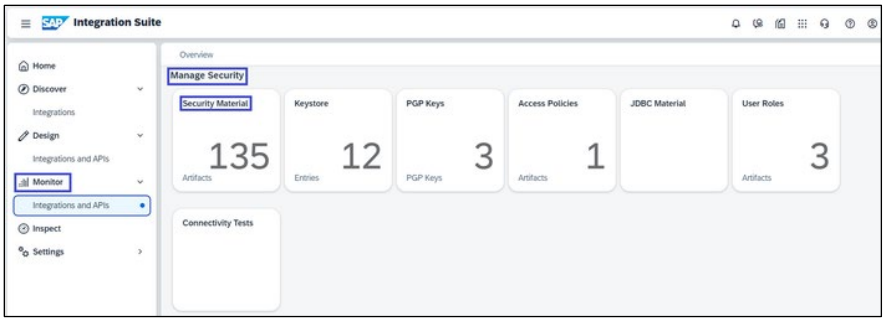
| Parameter | Description |
|---|---|
| **Name** | Specify the name for the security artifact.<br><br>The artifact name is used as an alias for the confidential data assigned by this parameter. |
| **Description** | Enter a description for the artifact (optional). |
| **Type** | Select **User Credentials** if creating credentials for Salesforce Pub/Sub. This allows you to configure a specific system to enable a connection with your integration flow artifact. |
| **User** | Specify the username used to invoke the receiver system. |
| **Password** | Specify password against which the user has to be authenticated. |

5. Click **Deploy** to complete the process.

When you refresh the **Manage Security Material** page, the new artifact is displayed (with Type **Credentials**) in the artifact table.

## 3.2.3 Creating Secure Parameter

1. In SAP Integration Suite, navigate to **Monitor > Integrations and APIs**. This opens the **Overview** page.
2. On the **Overview** page, go to **Manage Security** section and click **Security Material**.




3. On **Manage Security Material** page, click **Create** to select **Secure Parameter** from the dropdown.

4. In the **Create Secure Parameter** popup, provide the below details.

| Parameter | Description |
|---|---|
| **Name** | Specify the name of the security artifact. The artifact name is used as an alias for the confidential data. |
| **Description** | Enter a description for the artifact (optional). |
| **Secure Parameter** | Enter the confidential value of the attribute.<br><br>The permissible length of the secure parameter for Cloud Foundry is a maximum of 4096 characters. |
| **Repeat Secure Parameter** | Repeat the confidential value of the attribute. |

5. Click **Deploy** to complete the process.

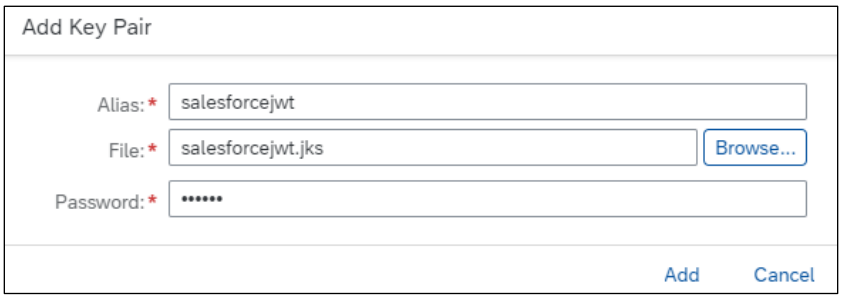
When you refresh the **Manage Security Material** page, the new artifact is displayed (with Type **Credentials**) in the artifact table.

## 3.2.4 Adding Key Pair to Keystore

To add **Key Pair** to Cloud Integration Keystore, follow the steps below:

1. Open the **Monitor** Tab in SAP Cloud Integration.
2. Select **Keystore** in the **Manage Security** section.


3. Select **Add** and choose **Key Pair**. Supply an Alias and the File and Password combination created in Section 6.2.2.



# 4 Salesforce Pub/Sub Adapter Configuration

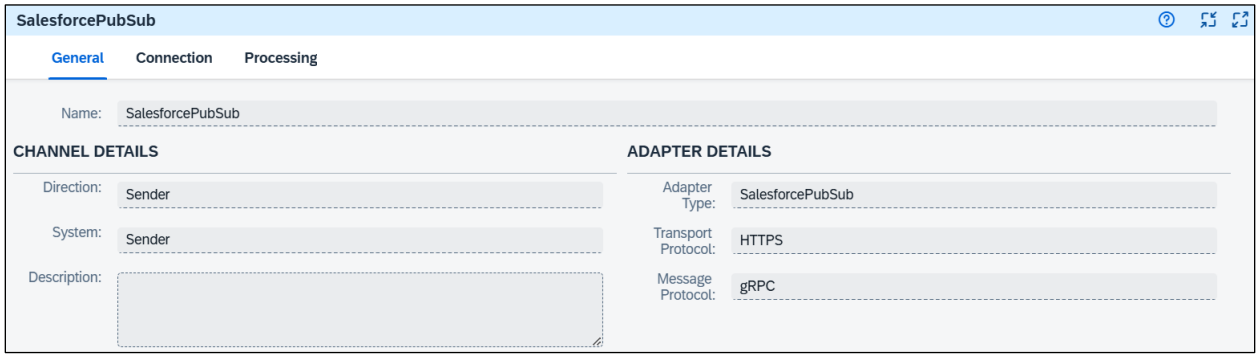
This section describes the parameters to be configured for your Salesforce Pub/Sub adapter. You need to configure the **General**, **Connection**, and **Processing** tabs. A description and example usage for every field has been added.

## 4.1 Sender Adapter

In this section, you will learn how to configure the Salesforce Pub/Sub sender adapter. On selecting the Salesforce Pub/Sub adapter from the list of adapters, you must configure the **Connection**, and **Processing** tabs.

### 4.1.1 General

The General tab provides an overview of basic adapter information including **Channel** and **Adapter** details.

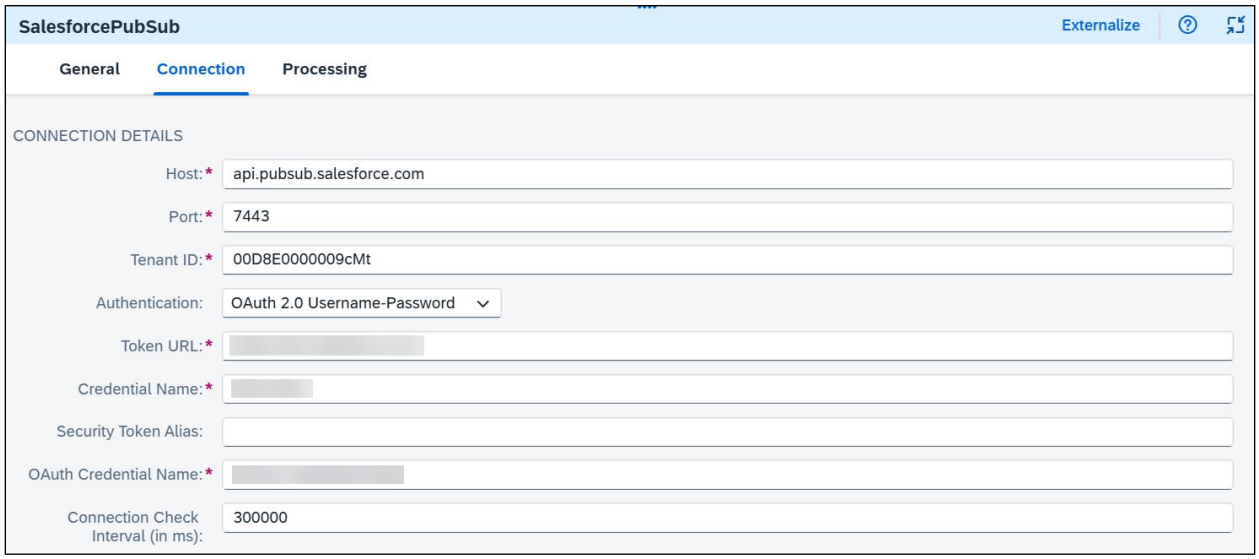



Only the Name and Description fields are editable.

| Parameter | Description |
|---|---|
| **Name** | Name of the adapter flow |
| **Description** | Description of the adapter |

## 4.1.2 Connection

The Connection tab contains connection and authentication parameters for Salesforce Pub/Sub.




The connection tab contains the following fields:

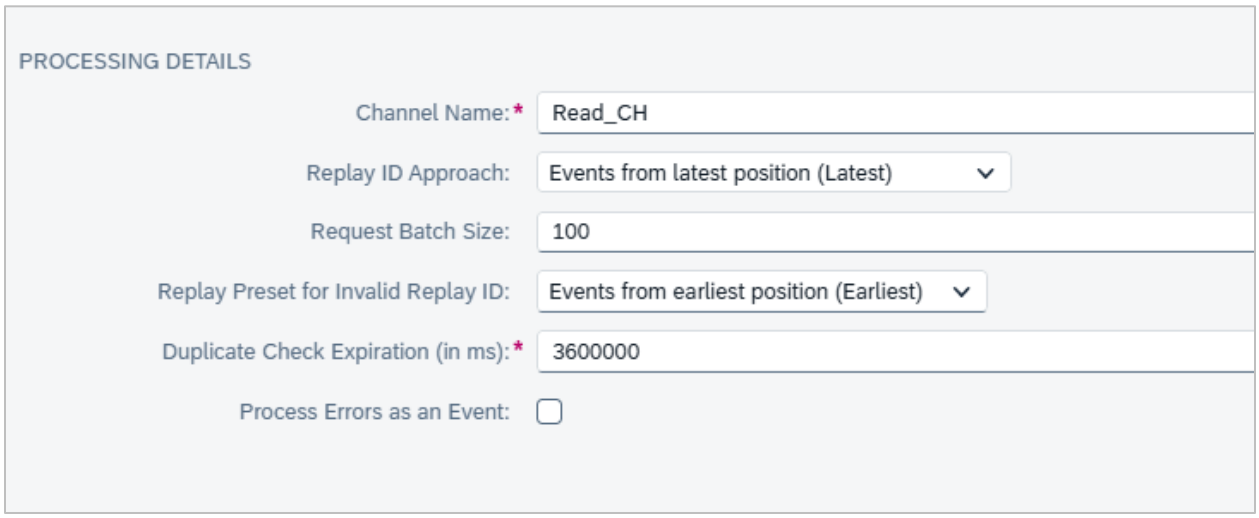
| Parameter | Description |
|---|---|
| **Host** | Specify the host to which the proxy requests are sent for the gRPC API (towards the gRPC host). The proxy requests to the gRPC host will go through the proxy.<br><br>Example: `api.pubsub.salesforce.com` |
| **Port** | Specify the port to which the proxy requests are sent for the gRPC API (towards the gRPC port). The proxy requests to the gRPC port will go through the proxy.<br><br>Example: `7443` |
| **Tenant ID** | Specify the Salesforce Organization ID. |
| **Connection Check Interval (in ms)** | Specify the interval in milliseconds to verify the connection validity of the gRPC. |

| Parameter | Description |
|---|---|
| **Authentication** | Select the authentication type to be used for the connection to Salesforce:<br>• **OAuth 2.0 Username-Password**<br>• **OAuth 2.0 JWT Bearer**<br>• **OAuth 2.0 Client Credentials** |
| The following fields are available when OAuth 2.0 Username-Password is selected. | |
| **Token URL**<br><br>(Also available when **OAuth 2.0 Client Credentials** is selected.) | Specify the login endpoint to your Salesforce instance url.<br><br>Example: https://login.salesforce.com for Salesforce production environment, https://test.salesforce.com for Salesforce Sandbox environment and https://{MyDomain}.my.salesforce.com in case MyDomain is enabled on the org. |
| **Credential Name** | Specify the User Credentials storing the username-password details. |
| **Security Token Alias** | Specify the Security Token Alias that refers to the secure parameter.<br><br>This can be omitted if your SAP CI IP address has been whitelisted in Salesforce. |
| **OAuth Credential Name** | Specify the User Credentials that stores the Consumer key-Client secret pair. |
| The following fields are available when OAuth 2.0 JWT Bearer is selected. | |
| **Audience** | Specify the login endpoint to your Salesforce instance URL.<br><br>Example: https://login.salesforce.com for Salesforce production environment, https://test.salesforce.com for Salesforce Sandbox environment, and https://site.force.com/customers if implementing for an Experience Cloud site. |
| **Subject Alias** | Specify the Secure Parmeter that stores the username of the Salesforce user. |
| **Issuer Alias** | Specify the Secure Parameter which indicates the OAuth client_id of the connected app in Salesforce for which the certificate was registered. |

| Parameter | Description |
|---|---|
| **Keystore Alias** | Specify the alias name of the added JKS file in Keystore as a Key Pair. It consists of a key and certificate to sign the JWT |
| **Expiration (in ms)** | Specifies the validity of the assertion in milliseconds. |
| The following field is available when **OAuth 2.0 Client Credentials** is selected. | |
| **OAuth2 Client Credentials** | The alias name of the deployed OAuth2 Client Credentials artifact that uses Client ID and Client Secret. |

## 4.1.3 Processing

The Processing tab lists all the operations that can be performed through the adapter.




| Parameter | Description |
|---|---|
| **Channel Name** | Specify the Salesforce channel name to perform the subscription.<br>The channel name is case-sensitive. |

| Parameter | Description |
|---|---|
| **Replay ID Approach** | Select the replay option in the first FetchRequest:<br><br>• **Events from a specific position (Custom)**<br>• **Events from earliest position (Earliest)**<br>• **Events from latest position (Latest)**<br><br>The default value is **Events from latest position (Latest)** which allows subscription only to new event messages and does not retrieve earlier event messages stored in the event bus. |
| **Replay ID** | Specify the Replay ID value. The Replay ID is populated by Salesforce and refers to the position of the event in the event stream.<br><br>Example: `8381402` |
| **Request Batch size** | Specify the number of events to be requested in the Salesforce fetchRequest. The default value is 100.<br><br>The maximum value for this field is 100. In case of values greater than 100, Salesforce will ignore it and treat it as 100.<br><br>After the adapter retrieves the number of events specified in the batch size, it automatically requests the next set of events using the same batch size. |
| **Replay Preset for Invalid Replay ID** | Select the replay option in case "sfdc.platform.eventbus.grpc.subscription.fetch.replayid.corrupted" error occurs:<br><br>• **Events from latest position (Latest)**<br>• **Events from earliest position (Earliest)**<br><br>The default value is **Events from earliest position (Earliest)** indicating the earlier event messages stored in the event bus. |
| **Duplicate Check Expiration (in ms)** | Specify the expiry time in milliseconds while handling the Duplicate check.<br><br>The default value is 3600000. |

| Parameter | Description |
| --- | --- |
| **Process Errors as an Event** | Enable this property to write error events in the exchange. If disabled, issues connecting to event streams will only be written to the logs. |

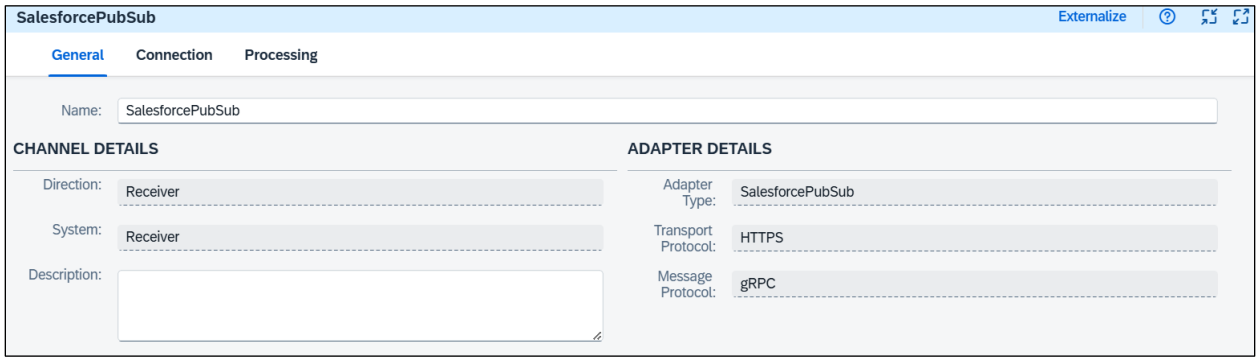
For optimal performance while using the sender adapter, it is recommended to ensure that the execution time of your interface is low. In case your scenario takes longer to execute, it is recommended to decouple the scenario.

# 4.2 Receiver Adapter

In this section, you will learn how to configure the Salesforce Pub/Sub receiver adapter. On selecting the Salesforce Pub/Sub adapter from the list of adapters, you must configure the **Connection**, and **Processing** tabs.

## 4.2.1 General

The General tab provides an overview of basic adapter information including **Channel** and **Adapter** details.




Only the Name and Description fields are editable.

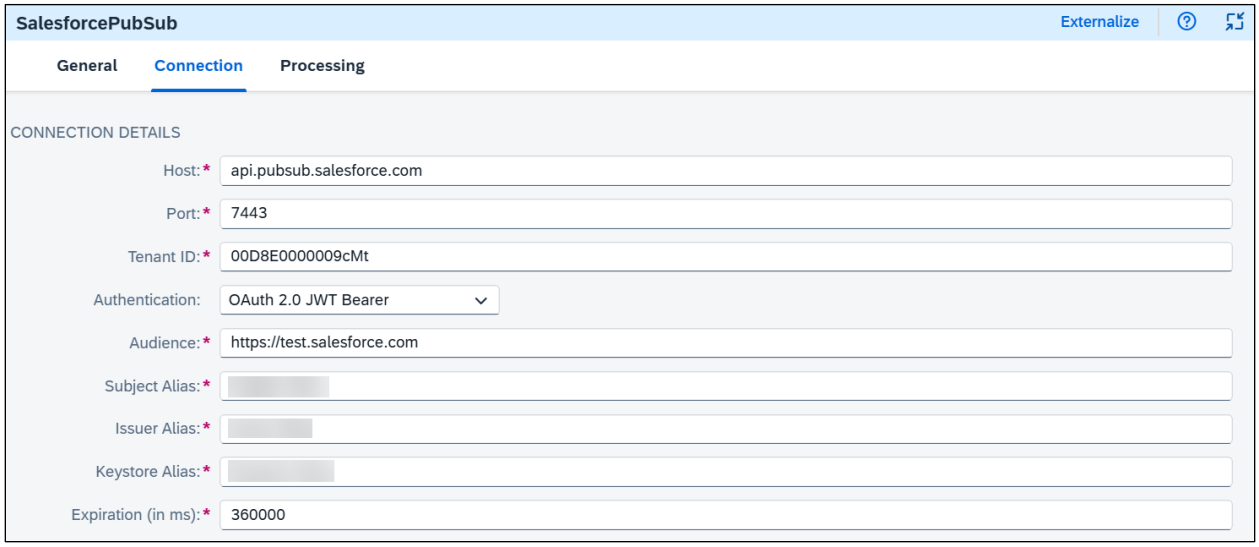
| Parameter | Description |
| --- | --- |
| **Name** | Name of the adapter flow |
| **Description** | Description of the adapter |

## 4.2.2 Connection

The **Connection** tab contains connection and authentication parameters for Salesforce Pub/Sub.

### Using Credentials

The Security artifact created in the previous section (Creating Credentials in Security Material) should be used in the **Connection tab** of the Adapter as shown in Figure below.




The connection tab contains the following fields:

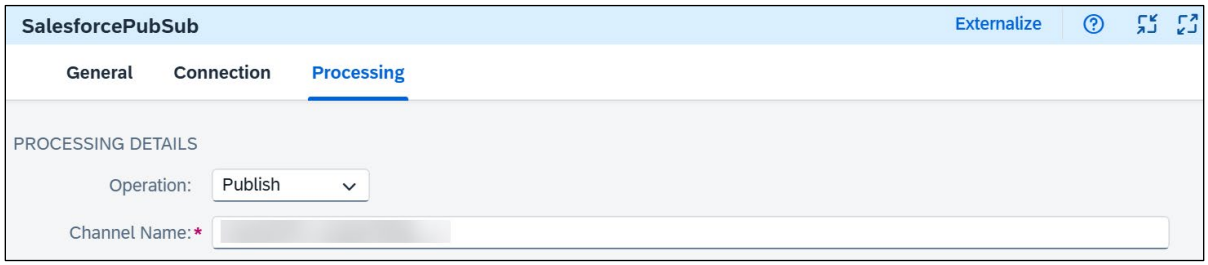
| Parameter | Description |
|---|---|
| **Host** | Specify the host to which the proxy requests are sent for the gRPC API (towards the gRPC host). The proxy requests to the gRPC host will go through the proxy.<br><br>Example: api.pubsub.salesforce.com |
| **Port** | Specify the port to which the proxy requests are sent for the gRPC API (towards the gRPC port). The proxy requests to the gRPC port will go through the proxy.<br><br>Example: `7443` |
| **Tenant ID** | Specify the Salesforce Organization ID. |

| Parameter | Description |
| --- | --- |
| **Authentication** | Select the authentication type to be used for the connection to Salesforce.<br><br>• **OAuth 2.0 Username-Password**<br>• **OAuth 2.0 JWT Bearer** |
| The following fields are available when OAuth 2.0 Username-Password is selected. | |
| **Token URL**<br><br>(Also available when **OAuth 2.0 Client Credentials** is selected.) | Specify the login endpoint to your Salesforce instance URL. Example: https://login.salesforce.com for Salesforce production environment, https://test.salesforce.com for Salesforce Sandbox environment, and https://{MyDomain}.my.salesforce.com in case MyDomain is enabled on the org. |
| **Credential Name** | Specify the User Credentials storing the username-password details. |
| **Security Token Alias** | Specify the Security Token Alias that refers to the secure parameter.<br><br>This can be omitted if your SAP CI IP address has been whitelisted in Salesforce. |
| **OAuth Credential Name** | Specify the OAuth Credential Name that refers to the user credentials (Consumer key-Client secret pair). |
| The following fields are available when OAuth 2.0 JWT Bearer is selected. | |
| **Audience** | Specify the login endpoint to your Salesforce instance URL.<br><br>Example: https://login.salesforce.com for Salesforce production environment, https://test.salesforce.com for Salesforce Sandbox environment and https://site.force.com/customers if implementing for an Experience Cloud site. |
| **Subject Alias** | Specify the Secure Parmeter that stores the username of Salesforce user. |
| **Issuer Alias** | Specify the Secure Parameter which indicates the OAuth client_id of the connected app in Salesforce for which the certificate was registered. |

| Parameter | Description |
|---|---|
| **Keystore Alias** | Specify the alias name of the added JKS file in Keystore as a Key Pair. It consists of a key and certificate to sign the JWT |
| **Expiration (in ms)** | Specifies the validity of the assertion in milliseconds. |
| The following field is available when **OAuth 2.0 Client Credentials** is selected. | |
| **OAuth2 Client Credentials** | The alias name of the deployed OAuth2 Client Credentials artifact that uses Client ID and Client Secret. |

## 4.2.3 Processing

The Processing tab lists all the operations that can be performed through the adapter.



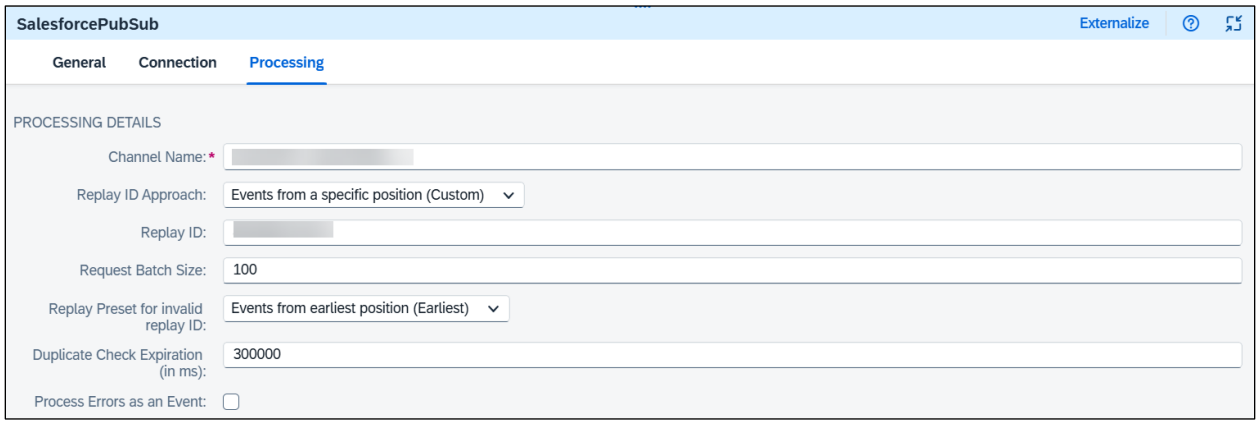

| Parameter | Description |
|---|---|
| **Operation** | Select the operation to perform:<br>• **Get Schema**<br>• **Publish** |
| **Channel Name** | Specify the Salesforce channel name to perform the operation.<br>The channel name should be case-sensitive. |

# 5 Operations

This section lists and describes the operations supported by the Salesforce Pub/Sub Sender and Receiver adapter.

## 5.1 Sender Adapter

The Salesforce Pub/Sub sender adapter allows you to **subscribe** to a channel and **retrieve** events by specifying the channel name.




| Parameter | Description |
|---|---|
| **Channel name** | Specify the channel name to perform the subscription. |
| **Replay ID approach** | Select the replay option as **Events from a specific position (Custom)** in the first FetchRequest. |
| **Replay ID** | Specify the Replay ID value. The Replay ID is provided by Salesforce and indicates the position of the event in the event stream. |
| **Request Batch Size** | Specify the number of events to be requested in the Salesforce FetchRequest.<br>The default value is 100. |
| **Replay Preset for invalid replay ID** | Select the replay option in the case of a 'sfdc.platform.eventbus.grpc.subscription.fetch.replayid.corrupted' error occurs.<br>The default value is 'EARLIEST', indicating the earlier event messages stored in the event bus. |

| Parameter | Description |
|---|---|
| **Duplicate Check Expiration (in ms)** | Specify the expiry time in milliseconds when handling the duplicate check.<br><br>The default value is 3600000 ms. |
| **Process Errors as an Event** | Enable this property to write error events in the exchange. If disabled, issues connecting to event streams will only be written to the logs. |

The field Replay ID approach has 3 possible values which are mentioned in the table below.

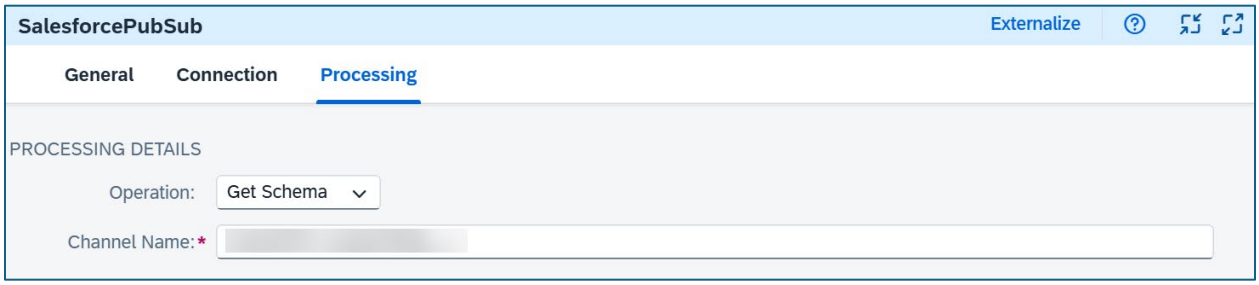
| Replay ID Approach | Behavior | When to Use |
|---|---|---|
| **Latest** | Subscribes to the latest events, starting from the most recent message. | Use this when you only need new events and don't require earlier messages stored in the event bus. |
| **Custom** | Subscribe from a specific replay ID (the last processed event or keepalive message). | Use this to catch up on missed events after a specific message, such as following a connection failure. |
| **Earliest** | Subscribes from the earliest available events. | Use to catch up on missed events and retrieve all stored events, especially if disconnected for over 3 days. |

# 5.2 Receiver Adapter

Following are the operations supported by the Salesforce Pub/Sub Receiver adapter.

## 5.2.1 Get Schema

This operation can be used to retrieve the schema from the Salesforce event. For more information about the operation, see Salesforce Pub/Sub documentation.



| Parameter | Description |
|---|---|
| **Operation** | Select the operation as Get Schema. |
| **Channel Name** | Specify the Salesforce channel name to perform the Get Schema operation. |

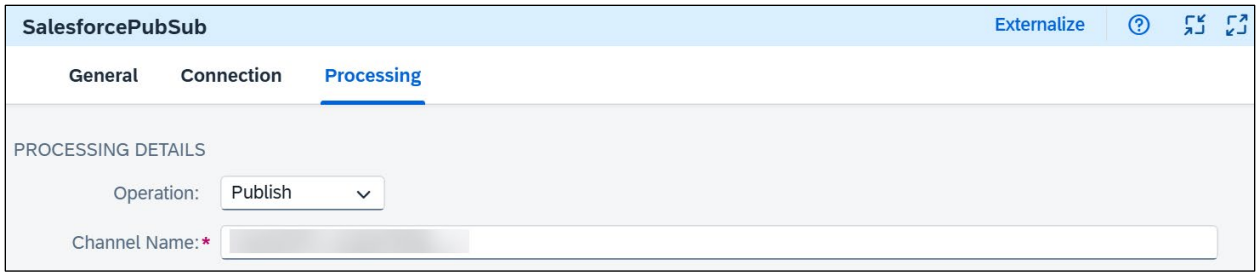
Sample response:


```
{
    "fields": [
        {
            "doc": "CreatedDate:DateTime",
            "name": "CreatedDate",
            "type": "long"
        },
        {
            "doc": "CreatedBy:EntityId",
            "name": "CreatedById",
            "type": "string"
        },
        {
            "default": null,
            "doc": "Data:Text:00N5E000006pRW5",
            "name": "CPI_CustomField__c",
            "type": [
                "null",
                "string"
            ]
        }
    ],
    "name": "CPI_Testing__e",
    "namespace": "com.sforce.eventbus",
    "type": "record"
}
```


## 5.2.2 Publish

The **Publish** operation allows you to send events to a topic/channel. When performing a publish operation, you need to specify the **Channel Name** to ensure the event is delivered to the correct subscribers.

For more information about the operation, see Salesforce Pub/Sub documentation.




| Parameter | Description |
|---|---|
| **Operation** | Select the operation as Publish. |
| **Channel Name** | Specify the Salesforce channel name to perform the Publish operation. |

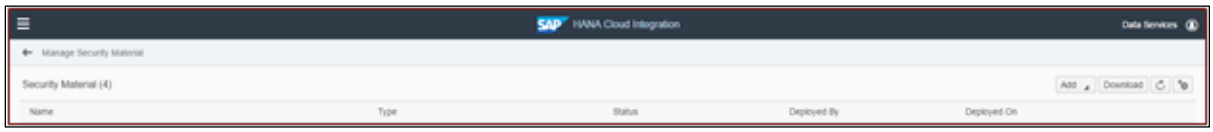
**Sample Request:**

```
{
    "CPI_CustomField__c": {
        "string": "DemoTest"
    },
    "CreatedById": "JohnDoe",
    "CreatedDate": 1732691760000,
    "CustomerName__c": {
        "string": "Demo"
    },
    "OrderID__c": {
        "string": "012345"
    }
}
```

# 6 Reference

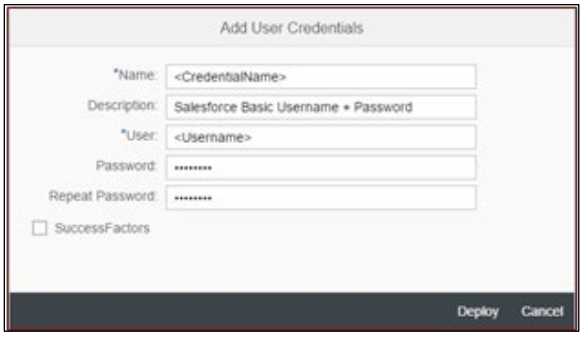
## 6.1 Initialize OAuth Username-Password in Salesforce

1. The Login URL is the login URL that is used by the authorization server of Salesforce. In the documentation of Salesforce, this URL is specified as https://login.salesforce.com. This link directs the Salesforce adapter to the login page of Salesforce to authorize.
2. The Basic Credential Name refers to an alias of a Username and Password combination stored in the SAP Cloud Integration Secure Store. This is the exact Username and Password used to log in to the Salesforce console.
3. Open the **Monitor** tab in SAP Cloud Integration.
4. Select **Security Material** in the section **Manage Security Material**.




**SAP Cloud Integration - Security Material**

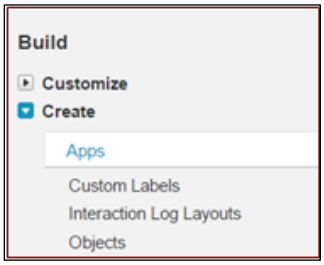
5. Select **Add**, select **User Credentials**, and enter a **Name** that should be the same as the Basic Credential Name configured in the Connection Tab of the Salesforce Adapter. In the **User** and **Password** fields, fill in the username and password used to log in to the Salesforce console. Click on **Deploy**.




**Add User Credentials for Basic Authentication**

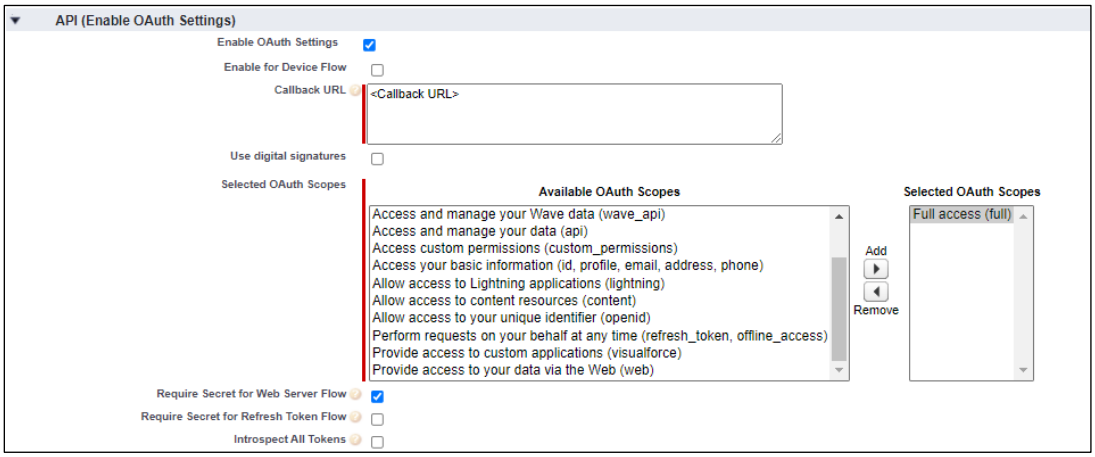
6. For the Security Token and OAuth Credentials, an app is required to be created in the Salesforce tenant. Login to the Salesforce console and select Setup.

7. On the left panel of the **Build** overview, select **Create**. Click on **Apps** and select New in the Connected Apps section.



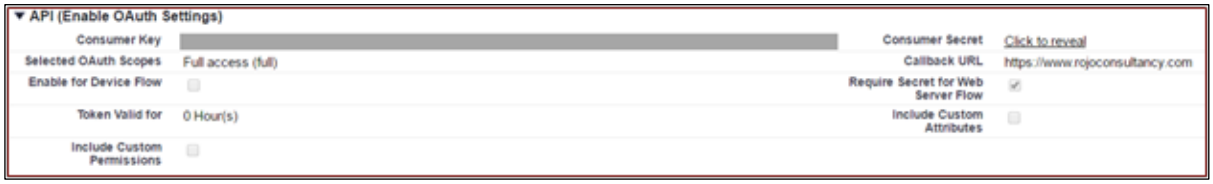

**Create Salesforce App**

8. In the next screen, fill in basic details such as **App Name**, **API Name**, and **Contact Email**.
9. In the **API (Enable OAuth Settings)**, select **Enable OAuth Settings**. Figure below displays a sample configuration of the OAuth Settings.
10. Click **Save** to complete the creation of the app.




**Salesforce API Enable OAuth Settings**

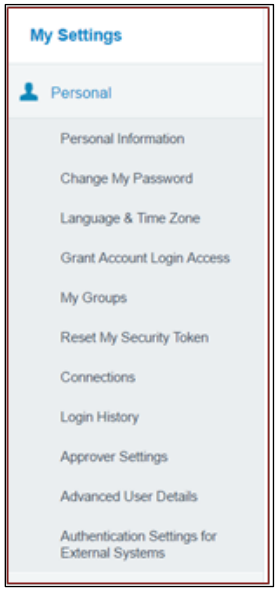
11. The next overview displays what can be seen when a specific Connected App is selected. The Consumer Key and Consumer Secret can be seen in the respective Consumer Key and Consumer Secret fields.




**OAuth Setting of the app**

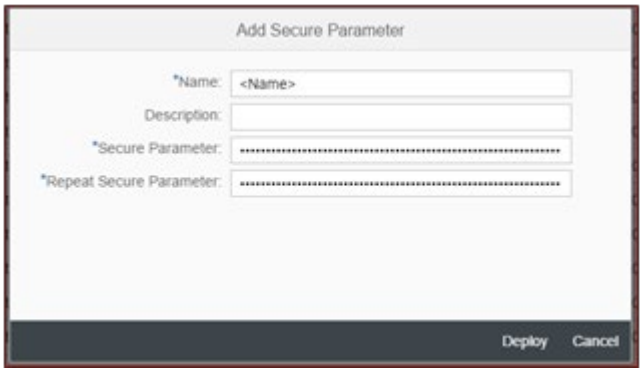
12. Deploy the **Consumer Key** and **Consumer Secret** as User Credentials similarly as done in **Step 2**. Create a new Credential Name and fill in Consumer Key as User and Consumer Secret as Password.

13. In case the IP address of SAP Cloud Integration is not whitelisted in the settings of Salesforce, a Security Token is needed. If the Security Token has not been received via email yet, navigate to My Settings and select Personal. Select Reset My Security Token to receive a new security token.




**Obtaining the Security Token**

14. To deploy the Security Token, select **Security Material** in the section **Manage Security Material**. Select **Add** and then select **Secure Parameter**.
15. Fill in the Secure Parameter Name. Note that this should match the Security Token Name used in the Salesforce Adapter. Fill in the Security Token in the **Secure Parameter** field and select **Deploy**.




**Deploy Security Token as a Secure Parameter**

16. Security Details successfully deployed in SAP Cloud Integration ensure a secure OAuth 2.0 Autonomous Client connection for the Salesforce Adapter. The names configured for Basic Credential Name, Security Token, and OAuth Credential Name must match the names of the deployed artifacts in the SAP Secure Stores. With the authentication mechanisms in place, the Salesforce Adapter can be configured for specific Salesforce integration scenarios.

# 6.2 Initialize OAuth JWT Bearer in Salesforce

When using the OAuth JWT Bearer, the following properties need to be maintained in the Connection Tab:

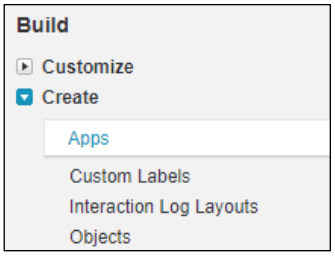
1. The Audience is the login URL used by the authorization server of Salesforce. In the documentation of Salesforce, this URL is specified as https://login.salesforce.com. This link directs the Salesforce Adapter to the login page of Salesforce.
2. The Subject Alias refers to an alias of a Secure Parameter stored in the SAP Cloud Integration Security Material. It specifies the username of the Salesforce user.
3. The Issuer Alias refers to an alias of a Secure Parameter stored in the SAP Cloud Integration Security Material. It specifies the OAuth Consumer Key of the connected app for which the certificate was registered.
4. The Keystore Alias refers to the added JKS file in Keystore as a Key Pair. It consists of a key and certificate to sign the JWT.

Refer to Salesforce documentation for more information.

## 6.2.1 Create a Connected App in Salesforce

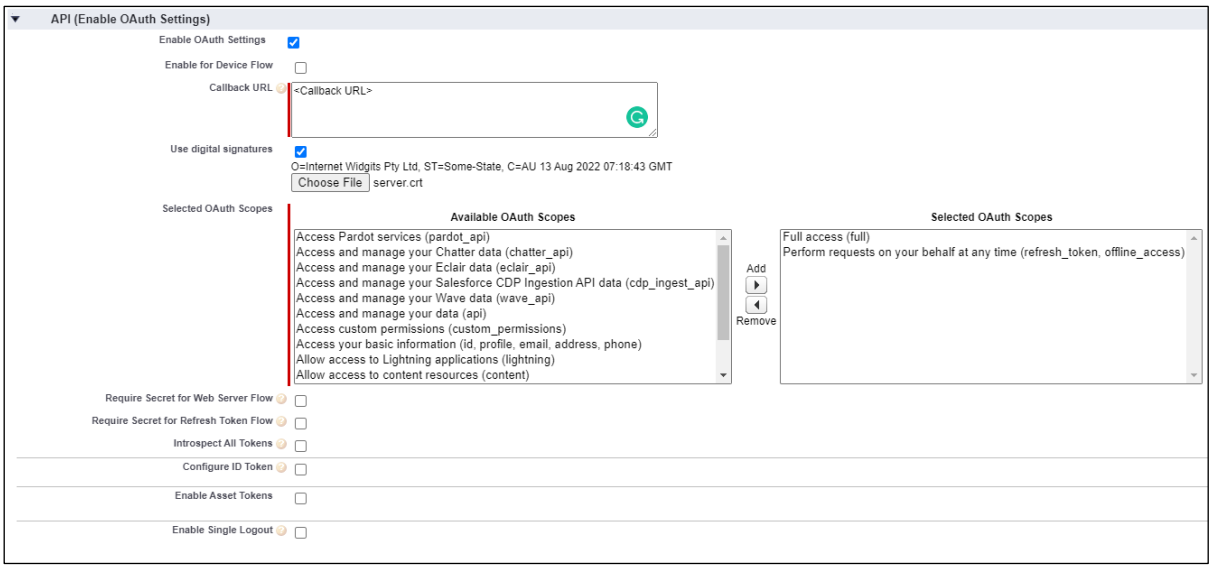
To create a **Connected App**, follow the steps below:

1. Login to the Salesforce console and select **Setup**.
2. On the left panel in the Build overview, select **Create**. Click on **Apps** and select **New** for the **Connected Apps** section.



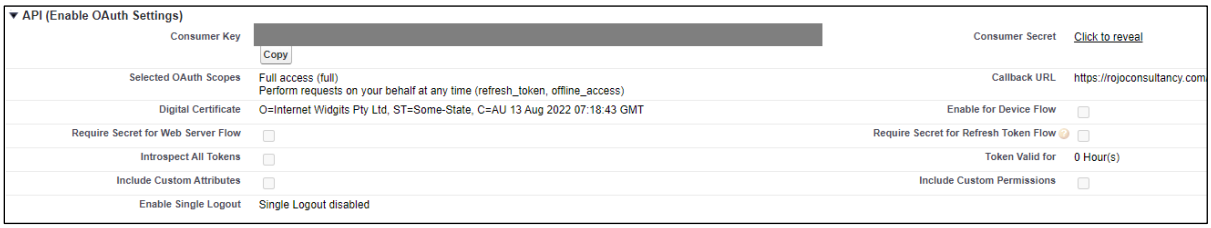

**Create Salesforce App**

3. In the next screen, fill in basic details such as **Connected App Name**, **API Name**, and **Contact Email**.
4. In the **API (Enable OAuth Settings),** select **Enable OAuth Settings**.
5. Select **Use digital signatures** and upload Security Certificate(*.crt). The figure below shows a sample configuration of the OAuth Settings section.



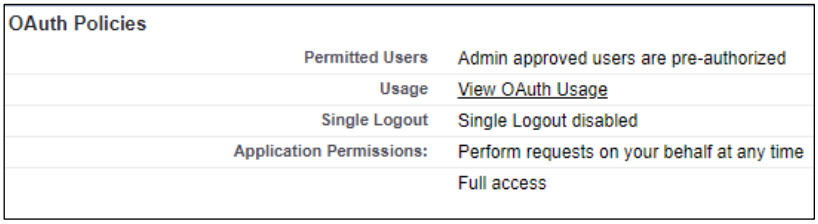

**API (Enable OAuth Settings)**

6. Click on **Save** to complete the creation of the App.
7. In the next view, the **Consumer Key** can be seen in the respective field.



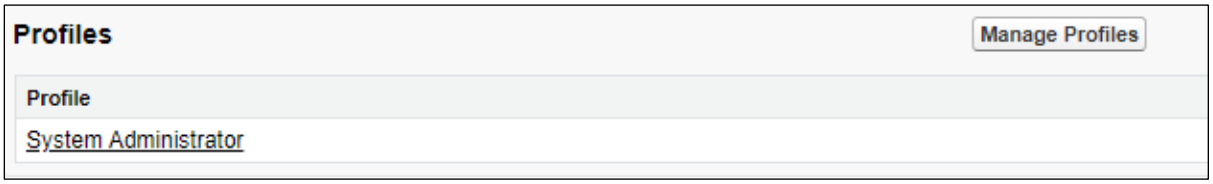

**Salesforce OAuth Setting indicating Consumer Key of the App**

8. In the created **App** overview, select **Manage** and choose **Edit** Policies.
9. Under **OAuth Policies**, change the **Permitted Users** to **Admin approved users are pre-authorized** from. See Figure below for a sample configuration.




**Salesforce App OAuth Policies**

10. From the **Edit Policies** overview, select **Manage Profiles** then select System Administrator as shown in Figure below.




**Salesforce App Manages Profiles**

## 6.2.2 Create Security Certificate and JKS file using OpenSSL

To create Security Certificate and JKS file using OpenSSL, follow the steps below:

1. Generate a private key and store it in a file called server.key

   **OpenSSL Command:**
   - `openssl genrsa -out server.pass.key 2048`
   - `openssl rsa -in server.pass.key -out server.key`

2. Generate a certificate signing request by using the server.key file created in the previous step and store it in a file called server.csr. Enter the information about the organization when prompted.

   **OpenSSL Command:**
   - `openssl req -new -key server.key -out server.csr`

3. Generate a self-signed digital certificate from the server.key and server.csr files and store the certificate in a file called server.crt. This server.crt is uploaded when creating the Connected App in Salesforce.

   **OpenSSL Command:**
   - `openssl x509 -req -sha256 -days 365 -in server.csr -signkey server.key -out server.crt`

4. Generate Java Key Store (JKS) file as salesforcejwt.jks and name/alias as salesforcejwt by providing the files server.key and server.crt created in previous steps.

**OpenSSL Command:**

- openssl pkcs12 -export -in server.crt -inkey server.key -out salesforcejwt.jks -name salesforcejwt
- Password:<PASSWORD>

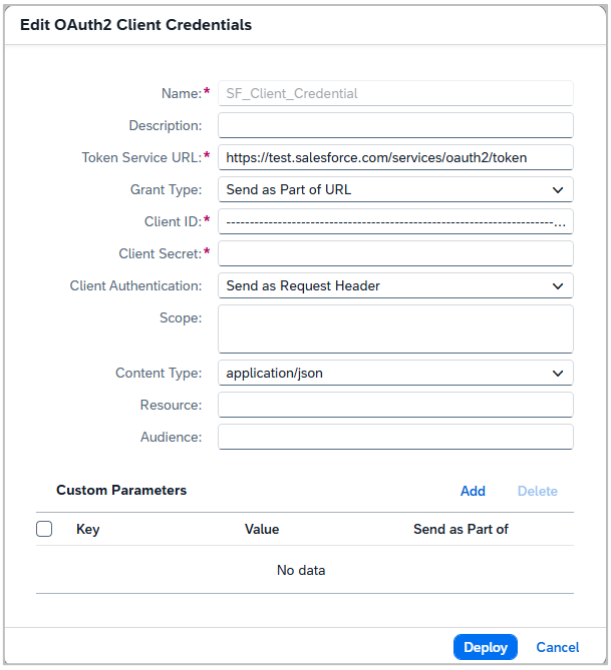
For more information, see Salesforce documentation.

## 6.3 Creating OAuth Client Credentials

- For more information on how to create OAuth Client Credentials, see OAuth 2.0 Client Credentials Flow.
- As per the instructions in the above link, you must create an external an external client app. For more information, see Create an External Client App.
- After completing the above setup, fetch **Consumer Key** and **Consumer Secret** for your app using **External Client App Manager**. You will require those details in the next step
- For creating OAuth2 Client Credentials in SAP Cloud Integration, see below procedure.

### 6.3.1 Creating OAuth2 Client Credentials in Security Material

1. Open the **Monitor** Tab in SAP Cloud Integration.
2. Select **Security Material** in the **Manage Security** section.
3. On the right side of your screen, click **Create** and select **OAuth2 Client Credentials.**
4. Populate the following details
    a. https://yourDomain.my.salesforce.com/services/oauth2/token **as Token Service URL**.
    b. Consumer Key as **Client ID.**
    c. Consumer Secret as **Client Secret.**
    d. Retain Client Authentication as **Send as Request Header**.
    e. Retain Content Type as **application/json.**




5. Click **Deploy** to finish.