



# Google Pub/Sub Adapter for SAP Integration Suite

Version 1.0.0 – February 2025

# Contents

1	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	Adapter Installation on Cloud Foundry.....	6
2.1.1	Prerequisite .....	6
2.1.2	Procedure .....	6
2.1.2.1	Adapter Installation by Creating a New Integration Flow .....	6
2.1.2.2	Adapter Installation without Creating a New Integration Flow.....	7
2.1.3	Monitor the Deployment Status .....	8
3	Getting Started: Google Pub/Sub Adapter .....	10
3.1	Architecture Overview .....	10
3.2	Application Configuration.....	11
3.3	Authentication.....	11
3.3.1	Creating Keys for OAuth2 Service Account.....	11
3.3.1.1	Creating Keystore for JSON key .....	11
3.3.1.2	Creating Key-Pair for P12 key.....	14
4	Google Pub/Sub Adapter Configuration .....	16
4.1	Sender Adapter .....	16
4.1.1	Sender (gRPC) .....	16
4.1.1.1	Connection Tab.....	16
4.1.1.2	Processing Tab.....	17
4.1.2	Sender (Rest).....	19
4.1.2.1	Connection Tab .....	19

4.1.2.2	Processing Tab .....	20
4.2	Receiver Adapter .....	22
4.2.1	Connection Tab .....	22
4.2.2	Processing Tab .....	23
5	Google Pub/Sub Adapter Operations .....	26
5.1	Sender Adapter .....	26
5.1.1	gRPC .....	26
5.1.2	Rest .....	27
5.2	Receiver Adapter .....	28
5.2.1	Topics: Publish Message .....	28
5.2.1.1	Basic .....	28
5.2.1.2	Advanced .....	29
5.2.2	Subscriptions: Modify Acknowledgment Deadline .....	30
6	Troubleshooting .....	32
7	References .....	33
7.1	Keystore Generation (JKS File Creation) .....	33

# 1 Introduction

## 1.1 Objective

This is the official guide for the Google Pub/Sub Adapter for SAP Integration Suite. This guide covers all relevant information for integration developers to start working with the Google 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

In many integration scenarios, messages must be exchanged between applications or systems. The Google Pub/Sub adapter is a Cloud Integration capability available on SAP Integration Suite that facilitates message publishing and subscription service. Create your message-based integration solutions with ease using the Google Pub/Sub Adapter.

## 1.5 Features

Google Pub/Sub Adapter has the following features:

- Receiver adapter allows reliable **publishing** of messages to Google Pub/Sub. It offers single message (**Basic**) and multiple message publishing (**Advanced**) capabilities.
- Support for listing Topics with **pagination** using **Page Token**.
- Allows you to modify the acknowledgement deadline of a message using the **Modify Acknowledgement Deadline** operation.
- Sender adapter allows **consuming** messages from Google Pub/Sub. You can read messages in real time (**gRPC**) or using a specified polling interval (**REST**).
- Provides **multiple post-processing** options: *Acknowledge after Successful Processing*, *Acknowledge Immediately*, and *Do not Acknowledge*.
- Offers secure authentication via **OAuth2 Service Account** option which uses Key Pair authentication.

## 2 Installation and Configuration

This section details the prerequisites to install and configure the Google Pub/Sub adapter.



The Google Pub/Sub adapter is available as part of your SAP Integration Suite license.

### 2.1 Adapter Installation on Cloud Foundry

Before the Google Pub/Sub adapter can be used in the Cloud Foundry environment, it must be deployed to the SAP Integration Suite tenant.

#### 2.1.1 Prerequisite

To deploy the Google PubSub adapter, you must have access to “*GooglePubSub Adapter for SAP Integration Suite*” as part of your SAP Integration Suite license.

#### 2.1.2 Procedure



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

You can deploy the adapter using the following methods:

##### 2.1.2.1 Adapter Installation by Creating a New Integration Flow

The Google Pub/Sub adapter is available for selection in the Sender and 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

Go to **Design** workspace and select the integration package where you want to create a new Integration flow.


1. Click **Edit** to make the package editable.

2. Go to the **Artifacts** tab. Click **Add** and select **Integration Flow**.
3. Enter the **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.
4. 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**.
5. A drop-down with the available adapters appears. The **GooglePubSub** adapter should show up in the list.
6. Select the **GooglePubSub** adapter from the list.

The adapter is now imported which *triggers* an adapter deployment. Once the Google Pub/Sub Adapter is deployed, a success message is *displayed*.

After the above steps are done, the Google Pub/Sub Adapter is successfully deployed in your **Design** workspace of the SAP Integration Suite tenant.

### 2.1.2.2 Adapter Installation without Creating a New Integration Flow

 The following procedure describes how the Google 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 Google 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 Google Pub/Sub adapter to a higher environment. Alternatively, if the TMS is not available in the landscape, the adapter package can be imported into the Design workspace by copying it from the Discover workspace.

## Purpose

To import the GooglePubSub adapter to **Design** workspace by copying the integration package from **Discover** workspace.

## Procedure

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

This completes the adapter deployment to **Design** workspace.

## 2.1.3 Monitor the Deployment Status

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.

The screenshot shows a web interface for managing integration content. On the left, a sidebar titled "Integration Content (562)" contains a search bar and a table with the following data:

Name	Status
GooglePubSub Integration Adapter	Started

The main content area is titled "GooglePubSub" and includes "Undeploy" and "Download" buttons. It displays deployment details:

- Deployed On: Oct 29, 2024, 13:41:00
- Deployed By: [Redacted]
- ID: [Redacted]
- Version: 1.0.0
- Package: [Redacted]

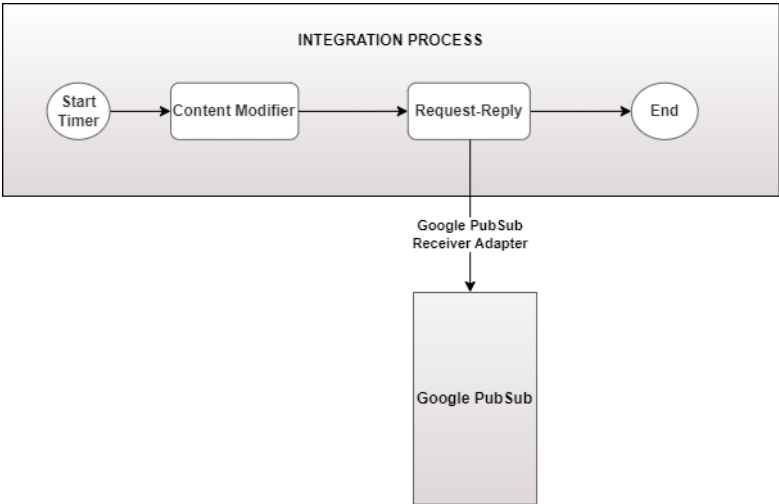
Below this information, a "Status Details" section features a green notification box stating: "The Integration Adapter is deployed successfully."

# 3 Getting Started: Google Pub/Sub Adapter

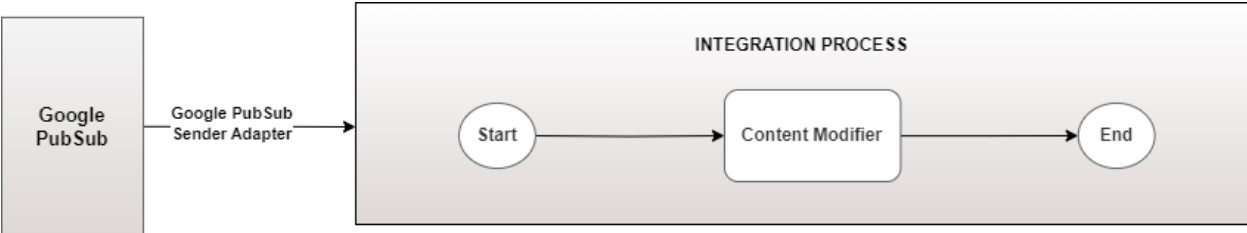
Before you start using the adapter, you can get to know underlying [Architecture](#) for the adapter and [Application Configuration](#) for Google Pub/Sub.

## 3.1 Architecture Overview

**How Google Pub/Sub Receiver Adapter Works:** The Google Pub/Sub Receiver adapter (as the name suggests) is designed to function as a receiver adapter. In such a scenario where the adapter is used as a receiver adapter, SAP Integration Suite acts as the initiator of the calls.



**How Google Pub/Sub Sender Adapter Works:** The Google Pub/Sub Sender adapter (as the name suggests) is designed to function as a sender adapter. In such a scenario where the adapter is used as a sender adapter, Google Pub/Sub acts as the initiator of the calls.



## 3.2 Application Configuration

- To get an overview of Pub/Sub, see [Google Pub/Sub Overview](#).
- For more information about connecting to Pub/Sub, see [OAuth for Service Accounts](#).
- For more information about roles for your service account, see [Access Control](#).
- To learn more about Google Pub/Sub publish and subscribe, see [Publish & receive messages in Pub/Sub](#).

## 3.3 Authentication

The Google Pub/Sub adapter supports authentication using OAuth Service Account. Find more information about how to create service account, see [Create Keys](#).

### 3.3.1 Creating Keys for OAuth2 Service Account

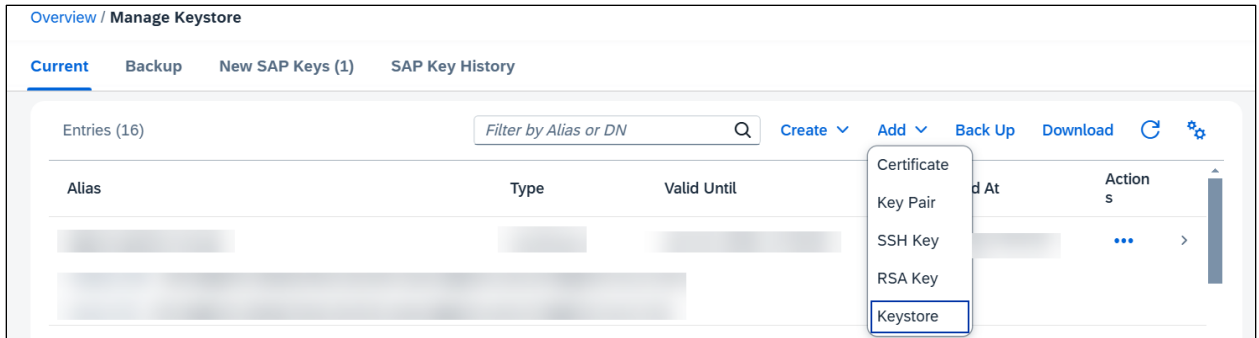
The Security artifact created below is used to connect to the Google Pub/Sub Application by configuring the **Connection tab** of the Adapter.

Login to Google Pub/Sub and generate a key. For more information, see [Create Keys](#). The **Key type** for private key can be either **JSON** or **P12**. Refer the following sections for each Key type usage.

#### 3.3.1.1 Creating Keystore for JSON key

1. After downloading the newly created key, the content in the key needs to be extracted and aligned in a proper format. Follow the below steps:
  - i. Open the key as a text file (preferably notepad++) and delete content so that you only retain the private key content between and inclusive of `BEGIN PRIVATE KEY` and `END PRIVATE KEY`.
  - ii. Remove all occurrences of `\n` from your private key content.





6. In the **Add Keystore** popup, provide the below details.

**Add Keystore**

Keystore: \*  Browse...

Passphrase: \*

Action: Add ▼

Overwrite existing entries

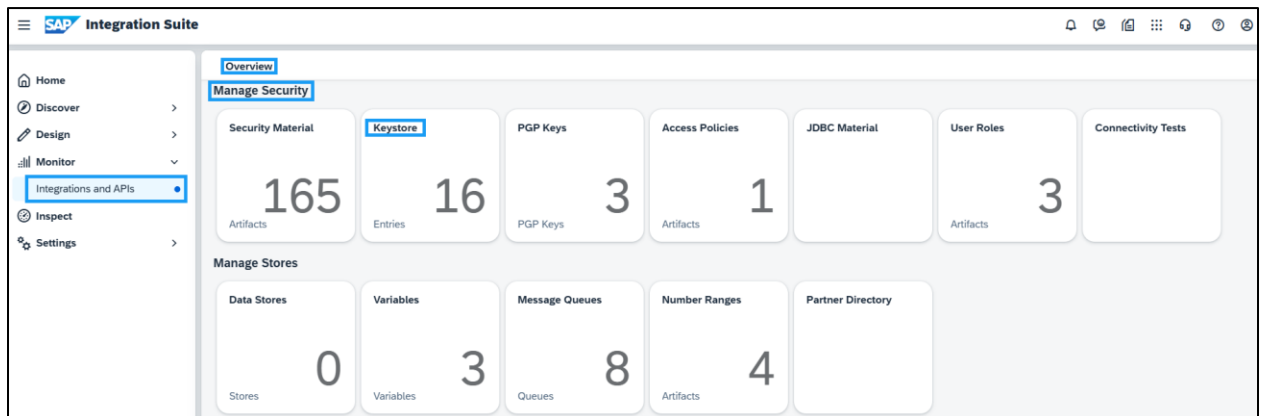
Deploy
Cancel

Parameter	Description
<b>Keystore</b>	Choose a keystore for upload. Select the .jks file. For detailed information, see <a href="#">Keystore Generation (JKS file creation)</a> .
<b>Passphrase</b>	Specify the password used while creating the .jks file.
<b>Action</b>	Specify the action to be performed as <b>Add</b> .
<b>Overwrite Existing Entries</b>	Enable or Disable depending on your need to overwrite existing entries.

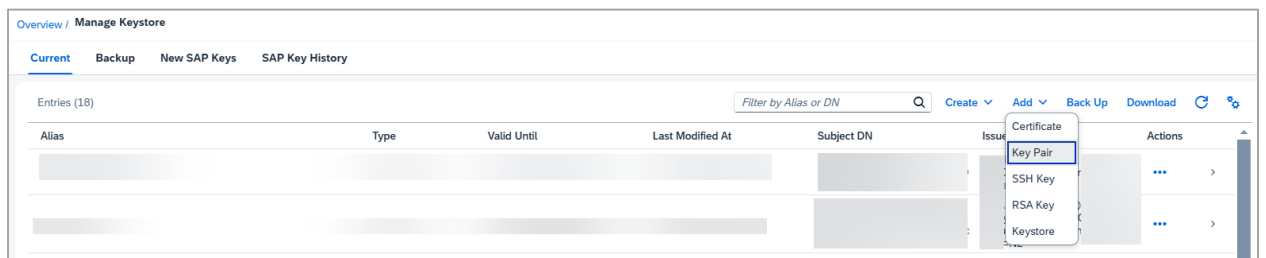
7. Click **Deploy** to complete the process.

### 3.3.1.2 Creating Key-Pair for P12 key

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 **Keystore**.



3. On **Manage Keystore** page, click **Add** to select **Key Pair** from the dropdown.



- In the **Key Pair** popup, provide the below details.

**Add Key Pair**

Alias: \*

File: \*

Password: \*

Parameter	Description
<b>Alias</b>	Specify the alias for the key pair.
<b>File</b>	Select the P12 key for upload.
<b>Password</b>	Specify the password used while creating the .jks file.

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

# 4 Google Pub/Sub Adapter Configuration

This section describes the parameters to be configured for your Google Pub/Sub sender and receiver adapters. You need to configure the **Connection** and **Processing** tabs. A description and example usage for each field has been added.

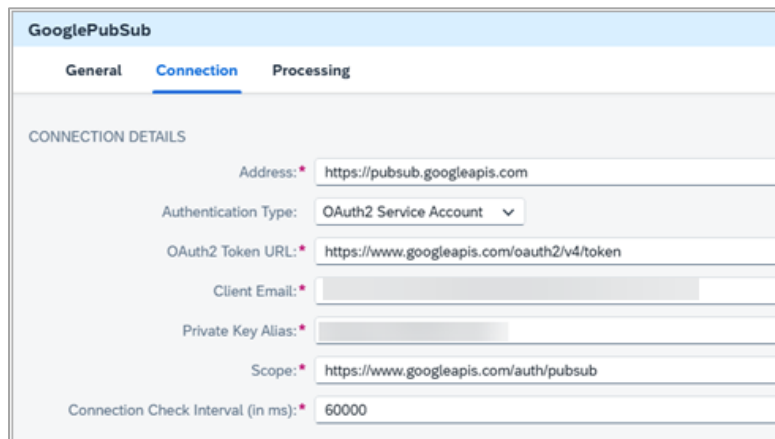
## 4.1 Sender Adapter

The Sender Adapter is available as **gRPC** and **REST** variants.

### 4.1.1 Sender (gRPC)

#### 4.1.1.1 Connection Tab

The **Connection** tab contains connection and authentication parameters for Google Pub/Sub. Before you set the connection details, see [Creating a Keystore](#).



The screenshot shows the 'GooglePubSub' configuration window with the 'Connection' tab selected. The 'CONNECTION DETAILS' section includes the following fields:

- Address:
- Authentication Type:
- OAuth2 Token URL:
- Client Email:
- Private Key Alias:
- Scope:
- Connection Check Interval (in ms):

The connection tab contains the following fields:

Parameter	Description
<b>Address</b>	Specify the hostname pointing to the Google Pub/Sub service.  Default: <a href="https://pubsub.googleapis.com">https://pubsub.googleapis.com</a>



Parameter	Description
<b>Authentication Type</b>	Select the method for authentication to Google Pub/Sub. Currently <b>OAuth2 Service Account</b> is supported.
<b>OAuth2 Token URL</b>	Specify the OAuth2 Token URL which identifies as the authorization server for producing a JWT token internally.  Default: <code>https://www.googleapis.com/oauth2/v4/token</code>
<b>Client Email</b>	Specify the Google Service Account Client Email.  Example: <a href="mailto:limited-svc-account@192843.iam.gserviceaccount.com">limited-svc-account@192843.iam.gserviceaccount.com</a>
<b>Private Key Alias</b>	Specify the Google Service Account Private Key for authentication.
<b>Scope</b>	Specify the scope of the connection to Google Pub/Sub service.  Default: <code>https://www.googleapis.com/auth/pubsub</code>
<b>Connection Check Interval (in ms)</b>	Specify the interval in milliseconds to verify the connection validity of gRPC.

#### 4.1.1.2 Processing Tab

The **Processing** tab contains the operational configurations.

**GooglePubSub**

General
Connection
Processing

PROCESSING DETAILS

Project ID:\*

Subscription ID:\*

Post-Processing: Acknowledge after Successful Processing v

Duplicate Check Expiration (in ms):\*

Parameter	Description
<b>Project ID</b>	Specify the Google Pub/Sub project ID.
<b>Subscription ID</b>	Specify the Google Pub/Sub Subscription ID from which the messages will be read.
<b>Post-Processing</b>	Select the action to be performed after polling a message: <ul style="list-style-type: none"> <li>• <b>Acknowledge after Successful Processing</b></li> <li>• <b>Acknowledge Immediately</b></li> <li>• <b>Do not Acknowledge</b></li> </ul>
<b>Duplicate Check Expiration (in ms)</b>	Specify the expiry time in milliseconds while handling the Duplicate check.  The default value is 300000.

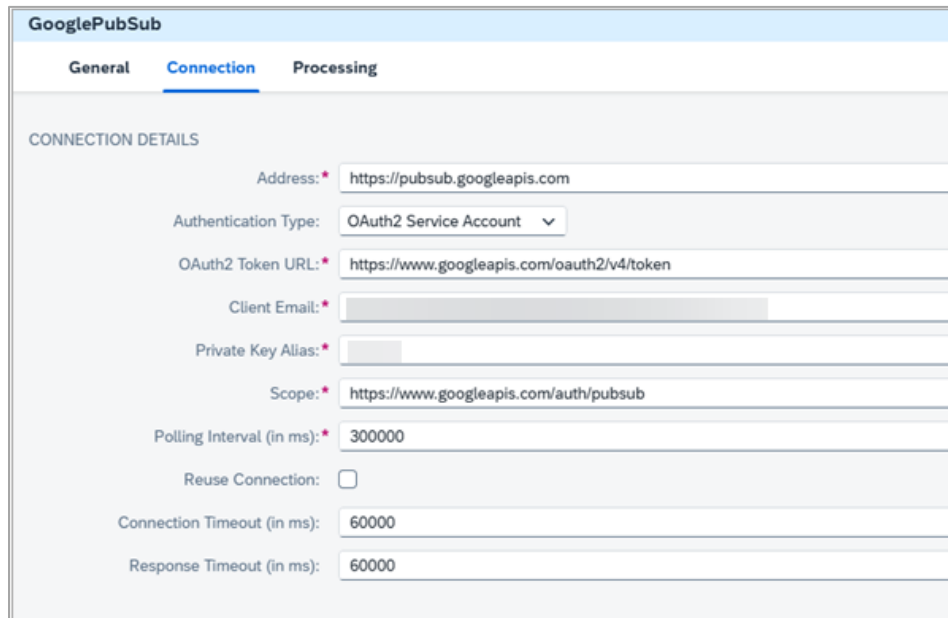
The message body for Sender Adapter adheres to a specific format. For more information regarding format, see below links:



- <https://cloud.google.com/pubsub/docs/reference/rest/v1/projects.subscriptions/pull#ReceivedMessage> (For gRPC variant, `ackId` and `deliveryAttempt` are not applicable)
- <https://cloud.google.com/pubsub/docs/reference/rest/v1/PubsubMessage>

## 4.1.2 Sender (Rest)

### 4.1.2.1 Connection Tab



The screenshot shows the 'GooglePubSub' configuration window with the 'Connection' tab selected. The 'CONNECTION DETAILS' section includes the following fields:

- Address:
- Authentication Type:
- OAuth2 Token URL:
- Client Email:
- Private Key Alias:
- Scope:
- Polling Interval (in ms):
- Reuse Connection:
- Connection Timeout (in ms):
- Response Timeout (in ms):

The **Connection** tab contains connection and authentication parameters for Google Pub/Sub. Before you set the connection details, see [Creating a Keystore](#).

The connection tab contains the following fields:

Parameter	Description
<b>Address</b>	Specify the hostname pointing to the Google Pub/Sub service.  Default: <a href="https://pubsub.googleapis.com">https://pubsub.googleapis.com</a>
<b>Authentication Type</b>	Select the method for authentication to Google Pub/Sub. Currently <b>OAuth2 Service Account</b> is supported.
<b>OAuth2 Token URL</b>	Specify the OAuth2 Token URL which identifies as the authorization server for producing a JWT token internally.  Default: <a href="https://www.googleapis.com/oauth2/v4/token">https://www.googleapis.com/oauth2/v4/token</a>

Parameter	Description
<b>Client Email</b>	Specify the Google Service Account Client Email.  Example: <a href="mailto:limited-svc-account@192843.iam.gserviceaccount.com">limited-svc-account@192843.iam.gserviceaccount.com</a>
<b>Private Key Alias</b>	Specify the Google Service Account Private Key for authentication.
<b>Scope</b>	Specify the scope of the connection to Google Pub/Sub service.  Default: <code>https://www.googleapis.com/auth/pubsub</code>
<b>Reuse Connection</b>	Enable to reuse this connection.
<b>Polling Interval (in ms)</b>	Specify the Polling Interval in milliseconds.
<b>Connection Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for the connection to be established with Google Pub/Sub service.
<b>Response Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for a response message to be received with Google Pub/Sub service.

### 4.1.2.2 Processing Tab

**GooglePubSub**

General   Connection   **Processing**

---

PROCESSING DETAILS

Project ID: \*

Subscription ID: \*

Max Messages to Read: \*


Post-Processing:  ▾

Duplicate Check Expiration (in ms): \*

HEADER DETAILS

Response Headers: \*

The **Processing** tab contains the operational configurations for the Google Pub/Sub Sender adapter.

Parameter	Description
<b>Project ID</b>	Specify the Google Pub/Sub project ID.
<b>Subscription ID</b>	Specify the Google Pub/Sub Subscription ID from which messages will be read.
<b>Max Messages to Read</b>	<p>Specify the maximum number of messages to be returned for this request. This must be a positive integer. The Google Pub/Sub system may return fewer than the number specified.</p> <p> Maximum number of messages allowed in a Pull response is 1000.</p>
<b>Post-Processing</b>	<p>Select the action to be performed after polling a message:</p> <ul style="list-style-type: none"><li>• <b>Acknowledge after Successful Processing</b></li><li>• <b>Acknowledge Immediately</b></li><li>• <b>Do not Acknowledge</b></li></ul>
<b>Duplicate Check Expiration (in ms)</b>	Specify the expiry time in milliseconds while handling the Duplicate check. The default value is 300000ms.
<b>Response Headers</b>	Enter a list of headers coming from the target system's response, separated by a pipe ( ), to be received in the message. Use an * to receive all the headers from the target system, which is also the default value.

The message body for Sender Adapter adheres to a specific format. For more information regarding format, see below links:



- <https://cloud.google.com/pubsub/docs/reference/rest/v1/projects.subscriptions/pull#ReceivedMessage>
- <https://cloud.google.com/pubsub/docs/reference/rest/v1/PubsubMessage>

## 4.2 Receiver Adapter

### 4.2.1 Connection Tab

The **Connection** tab contains connection and authentication parameters for Google Pub/Sub. Before you set the connection details, see [Creating a Keystore](#).

The screenshot shows the 'GooglePubSub' configuration window with the 'Connection' tab selected. The 'CONNECTION DETAILS' section contains the following fields:

- Address:
- Authentication Type:
- OAuth2 Token URL:
- Client Email:
- Private Key Alias:
- Scope:
- Reuse Connection:
- Connection Timeout (in ms):
- Response Timeout (in ms):

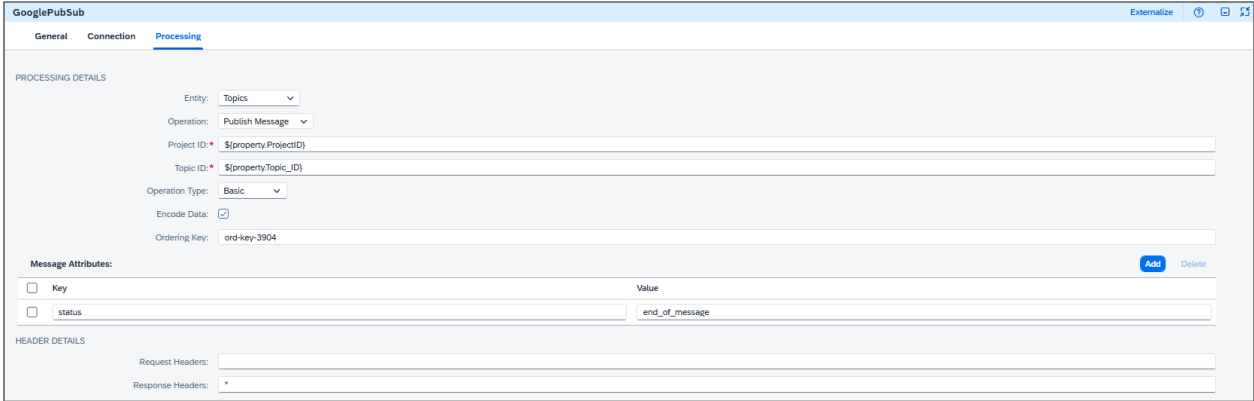
The connection tab contains the following fields:



Parameter	Description
<b>Address</b>	Specify the name for the Google Pub/Sub service.  Default: <a href="https://pubsub.googleapis.com">https://pubsub.googleapis.com</a>
<b>Authentication Type</b>	Select the method for authentication to Google Pub/Sub. Currently <b>OAuth2 Service Account</b> is supported.
<b>OAuth2 Token URL</b>	Specify the OAuth2 Token URL which identifies as the authorization server for producing a JWT token internally.  Default: <a href="https://www.googleapis.com/oauth2/v4/token">https://www.googleapis.com/oauth2/v4/token</a>
<b>Client Email</b>	Specify the Google Service Account Client Email.  Example: <a href="mailto:limited-svc-account@192843.iam.gserviceaccount.com">limited-svc-account@192843.iam.gserviceaccount.com</a>

Parameter	Description
<b>Private Key Alias</b>	Specify the Google Service Account Private Key for authentication.
<b>Scope</b>	Specify the scope of the connection to Google Pub/Sub service.  Default: <a href="https://www.googleapis.com/auth/pubsub">https://www.googleapis.com/auth/pubsub</a>
<b>Reuse Connection</b>	Enable this property to reuse the connection.
<b>Connection Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for the connection to be established with Google Pub/Sub service.
<b>Response Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for a response message to be received with Google Pub/Sub service.


### 4.2.2 Processing Tab

The **Processing** tab contains the operational configurations for the Google Pub/Sub receiver adapter.



Parameter	Description
<b>Entity</b>	Select the entity for message processing: <ul style="list-style-type: none"> <li>• <b>Topics</b></li> <li>• <b>Subscriptions</b></li> </ul>
<b>Operation</b>	Select the operation to be performed.
<b>Project ID</b>	Specify the Google Pub/Sub project ID.
<b>Topic ID</b>	Specify the Topic ID to be used.
<b>Operation Type</b>	Select the type of operation when <b>Operation</b> is set to <b>Publish Message</b> .
<b>Encode Data</b>	Enable to encode data to Base64. <div style="background-color: #e6f2ff; padding: 5px; margin-top: 10px;">  Google Pub/Sub can only accept Base64 encoded data. If not enabled data must be converted to Base64 before using the adapter. </div>
<b>Ordering Key</b>	Specify the ordering key for this message. <div style="background-color: #e6f2ff; padding: 5px; margin-top: 10px;">  For subscriptions with message ordering enabled, messages with the same ordering key are sent in the order in which they were published. </div>
<b>Subscription ID</b>	Specify the Subscription ID.
<b>Acknowledgement ID</b>	Specify the Acknowledgement ID for the messages being used.
<b>Acknowledgement Deadline (in sec)</b>	Specify the new acknowledgement deadline with respect to the time when this request is sent to the Google Pub/Sub system.
<b>Page Size</b>	Specify the maximum number of topics to be returned.  Example: 10



Parameter	Description
<b>Page Token</b>	<p>Specify the value returned by the last ListTopicsResponse; indicates that this is a continuation of a prior topics.list call, and that the system should return the next page of data.</p> <div data-bbox="508 436 1398 554" style="background-color: #e6f2ff; padding: 5px;">  A value of null or an empty string retrieves the first page. </div>
<b>Message Attributes</b>	<p>Specify multiple message attributes as key value pairs.</p> <p>Example: Set <b>Key</b> to <code>status</code> and <b>Value</b> to <code>end_of_message</code></p>
<b>Request Headers</b>	<p>Enter a list of custom headers, separated by a pipe ( ), that you want to send to the target system. By default, no custom headers are sent.</p> <p>Alternatively, use an * to send all custom headers to the target system.</p>
<b>Response Headers</b>	<p>Enter a list of headers coming from the target system's response, separated by a pipe ( ), to be received in the message.</p> <p>Use an * to receive all the headers from the target system, which is also the default value.</p>

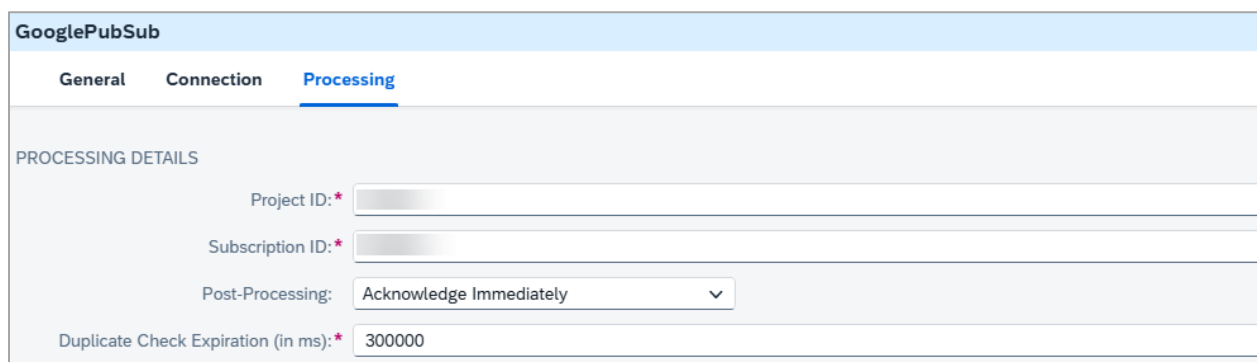
# 5 Google Pub/Sub Adapter Operations

This section describes some of the operations supported by the Google Pub/Sub adapter.

## 5.1 Sender Adapter

You can read messages from a subscribed topic using the Google Pub/Sub adapter.

### 5.1.1 gRPC



The screenshot shows the 'GooglePubSub' configuration interface with the 'Processing' tab selected. Under 'PROCESSING DETAILS', the following fields are visible:

- Project ID: \* [text input field]
- Subscription ID: \* [text input field]
- Post-Processing: [dropdown menu showing 'Acknowledge Immediately']
- Duplicate Check Expiration (in ms): \* [text input field with value '300000']

Parameter	Values
<b>Project ID</b>	Set as <code>project-4321</code>
<b>Subscription ID</b>	Set as <code>proj431-sub</code>
<b>Post-Processing</b>	Select action as <b>Acknowledge Immediately.</b>
<b>Duplicate Check Expiration (in ms)</b>	Set value as 300000

## 5.1.2 Rest

GooglePubSub

[General](#) [Connection](#) [Processing](#)

PROCESSING DETAILS

Project ID: \*

Subscription ID: \*

Max Messages to Read: \*

Post-Processing:

Duplicate Check Expiration (in ms): \*

HEADER DETAILS

Response Headers:

Parameter	Values
<b>Project ID</b>	Set as <code>project-4321</code>
<b>Subscription ID</b>	Set as <code>firstsub-324</code>
<b>Max Messages to Read</b>	Set value to 10
<b>Post-Processing</b>	Set to <b>Do not Acknowledge.</b>
<b>Duplicate Check Expiration (in ms)</b>	Set value is 300000

## 5.2 Receiver Adapter

### 5.2.1 Topics: Publish Message

#### 5.2.1.1 Basic

The screenshot shows the 'Processing' tab of the GooglePubSub configuration interface. The 'PROCESSING DETAILS' section includes the following settings:

- Entity: Topics
- Operation: Publish Message
- Project ID: \${property.ProjectID}
- Topic ID: \${property.Topic\_ID}
- Operation Type: Basic
- Encode Data:
- Ordering Key: ord-key-3904

The 'Message Attributes' section shows a table with one entry:

Key	Value
status	end_of_message

The 'HEADER DETAILS' section is currently empty.

Parameter	Values
<b>Entity</b>	Select <b>Topics</b> from the available dropdown.
<b>Operation</b>	Select Operation as <b>Publish Message</b> .
<b>Project ID</b>	Set as <code>\${property.ProjectID}</code>
<b>Topic ID</b>	Set as <code>\${property.Topic_ID}</code>
<b>Operation Type</b>	Set as <b>Basic</b>
<b>Encode Data</b>	Enable this field.
<b>Ordering key</b>	Specify ordering key as <code>ord-key-3904</code>
<b>Message Attributes</b>	Set Key to <code>status</code> and Value to <code>end_of_message</code>

## 5.2.1.2 Advanced

GooglePubSub

General Connection **Processing**

PROCESSING DETAILS

Entity: Topics

Operation: Publish Message

Project ID: \* \${property.ProjectID}

Topic ID: \* \${property.Topic\_ID}

Operation Type: Advanced


HEADER DETAILS

Request Headers:

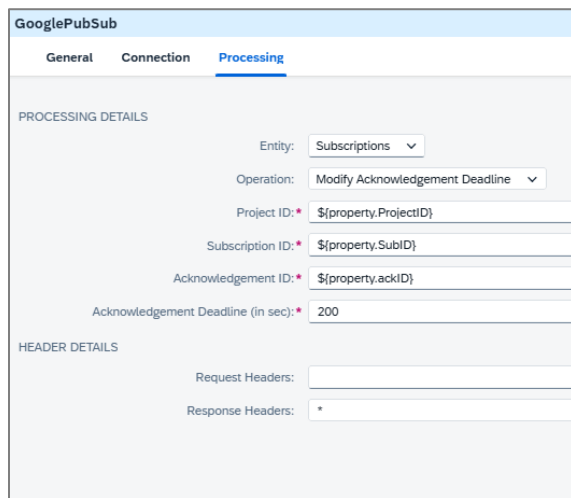
Response Headers: \*

Sample Payload:

```
{
  "messages": [
    {
      "data": "dFeWzdC1kYXRh",
      "attributes" : {},
      "orderingKey": "1"
    },
    {
      "attributes" : {
        "data1" : "ERfsdS",
        "data2" : "eqwWRfs"
      }, "data" : "dTVzWCR1kYXRh",
      "orderingKey": "1"
    },
    {
      "data": "",
      "attributes" : {
        "data3" : "value"
      },
      "orderingKey": "1"
    }
  ]
}
```


Parameter	Values
Entity	Select <b>Topics</b> from the available dropdown.
Operation	Select Operation as <b>Publish Message</b> .
Project ID	Set as <code>\${property.ProjectID}</code>
Topic ID	Set as <code>\${property.Topic_ID}</code>
Operation Type	Set as <b>Advanced</b> <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin-top: 10px;">  In case of multiple (Advanced) mode, body must be set in the expected JSON format before using this adapter. </div>

## 5.2.2 Subscriptions: Modify Acknowledgement Deadline



The screenshot shows the 'Processing' configuration page for GooglePubSub. The 'Entity' is set to 'Subscriptions' and the 'Operation' is set to 'Modify Acknowledgement Deadline'. The 'Project ID' is set to `${property.ProjectID}`, 'Subscription ID' is `${property.SubID}`, and 'Acknowledgement ID' is `${property.ackID}`. The 'Acknowledgement Deadline (in sec)' is set to '200'. The 'Request Headers' and 'Response Headers' fields are empty.

Parameter	Values
Entity	Select <b>Subscriptions</b> from the available dropdown.
Operation	Select Operation as <b>Modify Acknowledgement Deadline</b> .
Project ID	Set as <code>\${property.ProjectID}</code>
Subscription ID	Set as <code>\${property.SubID}</code>
Acknowledgement ID	Set as <code>\${property.ackID}</code>

Parameter	Values
<b>Acknowledgement Deadline (in sec)</b>	Set as 200   This value ranges from 0 to 600. For more information, see <a href="#">modifyAckDeadline</a> .

## 6 Troubleshooting

A few issues that you might encounter and possible solutions for them have been documented below.

Error	Error Message	Possible Solution
Publish Message Failure if <b>Encode Data</b> is unchecked and exchange body has invalid characters.	<pre>"message": "Invalid value at 'messages[0].data' (TYPE_BYTES), Base64 decoding failed for \"bkZpZWxkIjogdHJ1ZX0=\r\n\r\n\""</pre>	You must ensure encoded data being sent is properly encoded, check and remove invalid characters.



# 7 References

## 7.1 Keystore Generation (JKS File Creation)

1. Generate a certificate signing request by using the `gps_rsa_key.p8` file and store it in a file called `gps_server.csr`.

OpenSSL Command:

```
openssl req -new -key <source_path>/gps_rsa_key.p8 -out  
<target_path>/gps_server.csr
```

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:<enter\_country\_code>

State or Province Name (full name) [Some-State]:<enter\_city\_name>

Locality Name (eg, city) []:<enter\_locality\_name>

Organization Name (eg, company) [Internet Widgits Pty Ltd]:<enter\_org\_name>

Organizational Unit Name (eg, section) []:<enter\_org\_unit\_name>

Common Name (e.g. server FQDN or YOUR name) []:<enter\_common\_name>

Email Address []:<enter\_email\_address>

Please enter the following 'extra' attributes to be sent with your certificate request

A challenge password []:<password>

An optional company name []:<company\_name>

2. It is possible to sign the certificate using a self-signing mechanism or via CA.



Please refer to your organisation policy to choose a suitable signing mechanism for the certificate.

You can generate a self-signed digital certificate using our private key

(`gps_rsa_key.p8`) and certificate signing request (`gps_server.csr`) file and store the certificate in a file called (`gps_server.crt`).

This `gps_server.crt` is uploaded when creating the jks file.

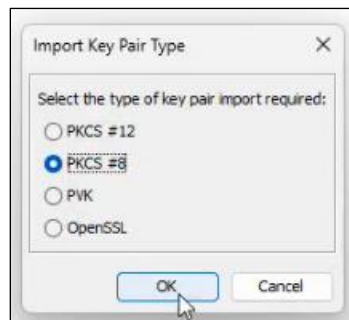
OpenSSL Command:

```
openssl x509 -req -sha256 -days 365 -in  
<source_path>/gps_server.csr -signkey  
<source_path>/gps_rsa_key.p8 -out <target/path>/gps_server.crt
```

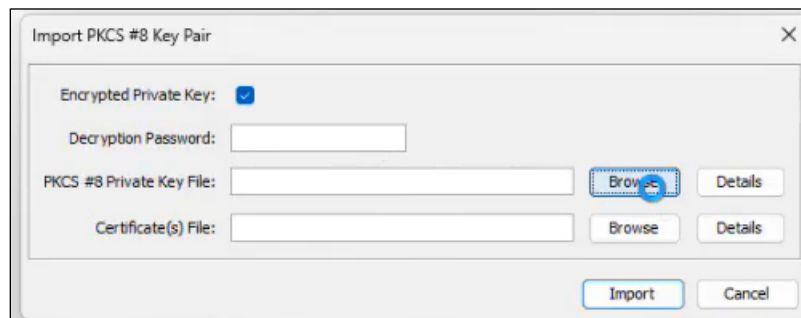
3. To create a JKS file using Keystore in Windows follow the below steps:
  - i. Go to Keystore and **Import key Pair**.

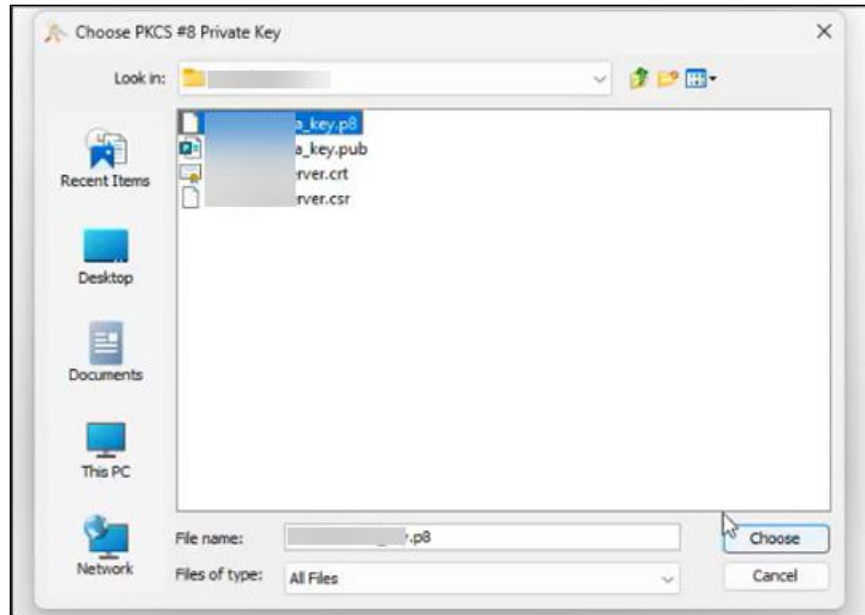


- ii. Select the type of key pair as **PKCS #8**.

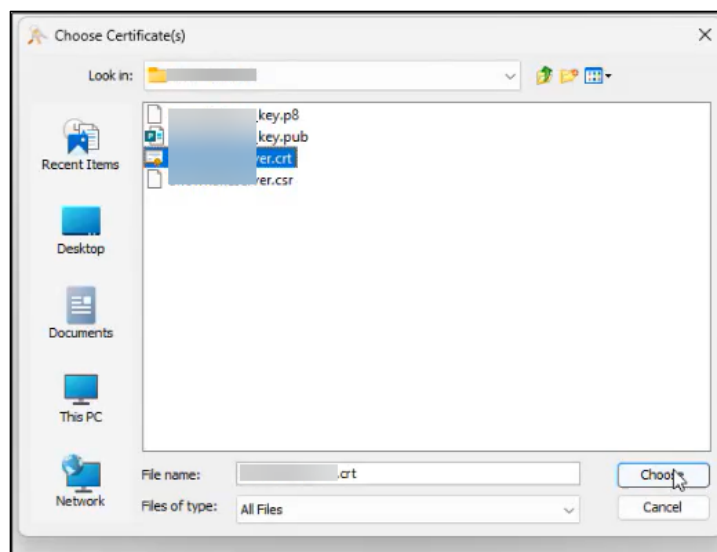
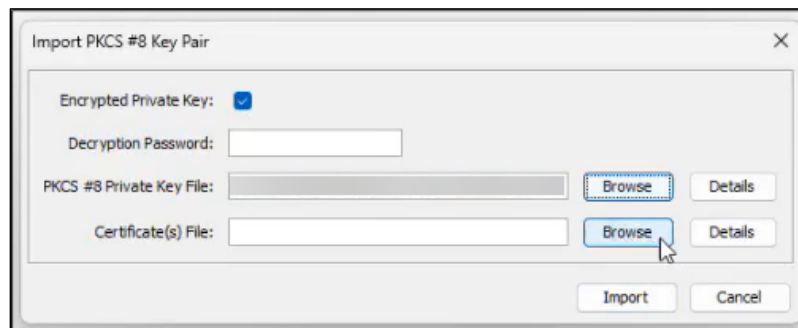


- iii. Add the private key file by uploading the PKCS **Private Key** file.

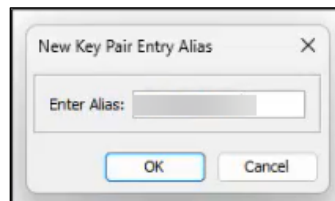
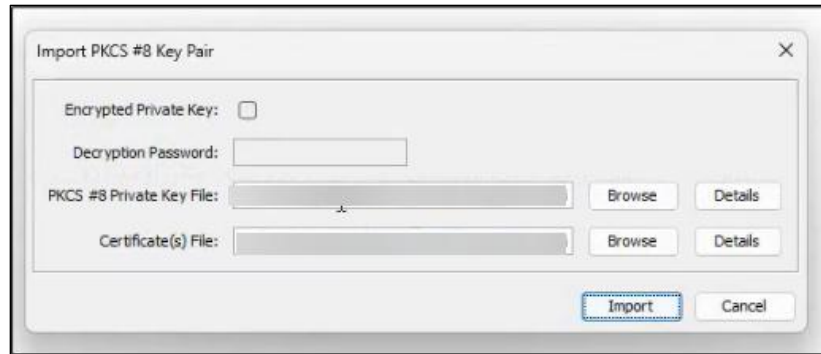




iv. Add the Google PubSub **Certificate(s) File** created above.



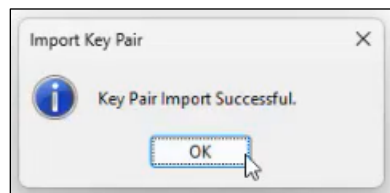
- v. Click **Import** and provide the **Alias**.



- vi. Provide a **New Key Pair Entry Password**.



- vii. You will get a message that indicates key pair import was successful.



viii. Provide the **KeyStore Password**.



ix. Finally, save the .jks file.

