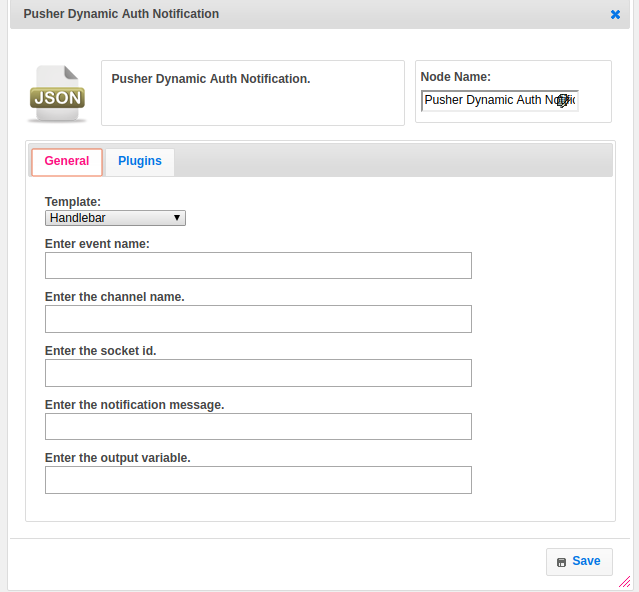
Pusher Dynamic Oauth Step

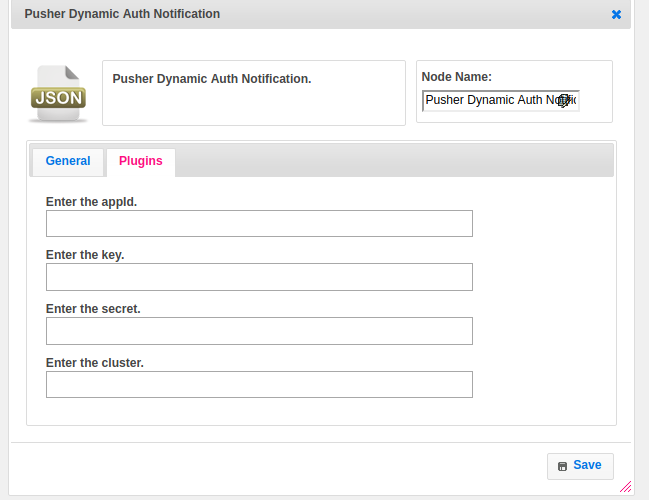
Objective

Pusher Channels provides real-time communication between servers, apps, and devices. Pusher Channels is used for notifications, chat, gaming, web-page updates, IoT, and many other systems requiring real-time communication.

Application security is very important so Channels provides a mechanism for authenticating a user’s access to a channel at the point of subscription.

To do so you must expose an authentication endpoint from your own servers that are called by the Channels WebSocket libraries. Your server can then respond to either allow or deny the subscription.

UI



## **Attributes**

|  |  |
| --- | --- |
| **Attributes** | **Description** |
| **1. Template** | Value type used to set the value type. Value types are two types.  1.hb-HandleBars  2.ftl-Freemarker Template Language. |
| **2. Enter event name** | The name of the event to bind to.  Events are the primary method of packaging messages in the Channels system. they form the basis of all communication. |
| **3. Enter the channel name** | Provide the name of the channel.  Each application can have one channel or many, and each client can choose which channels it subscribes to. |
| **4. Enter the socket id** | A unique identifier for the specific client connection to Channels.  Once the connection has been established a universally unique socket id is returned to the Channels |
| **5. Enter the notification message** | A message having information to provide the notification when the event trigger. |
| **6. Enter the output variable** | The output variable is used to hold the response. |
| **7.App Id** | The application key is a string which is globally unique to your application. It can be found in the API Access section of your application within the Channels user dashboard. |
| **8.Key** | Your application key, to provide access. |
| **9.Cluster** | The identifier of the cluster your application was created in. When not supplied, will connect to the mt1 (us-east-1) cluster. |
| **10.Secret** | All interactions with the Channels HTTP API must contain an authentication signature that is generated with your secret key. |
| **10.Output variable** | To store the response |

## **Executor Description**

Initialize the Pusher Oauth plugin. Authenticate the user authenticate method of Pusher API Pusher class. Send the message to the channel and event with message data by calling the pushMessage method and pass channel, event\_name, and message as arguments.Call trigger method to notification on specific channel and event name with notification object or message.

## **Resource**

Please refer the following link for more information.

<https://pusher.com/docs/authenticating_users>