

Close.io Events

Last Modified on 02/12/2019 4:15 pm EST

On this page

SAP Cloud Platform Open Connectors currently supports events via **polling** or **webhooks** depending on the endpoint.

- **Polling** is a mechanism where SAP Cloud Platform Open Connectors executes the configured query every `n` minutes and captures the changed information.
- **Webhooks** are when the provider lets SAP Cloud Platform Open Connectors know what information has changed. Note that additional endpoint setup may be required prior to creating your Connector Instance.

Note: SAP Cloud Platform Open Connectors normalizes only the `objectId`, `objectType`, and `eventType` event data. Event data also contains raw data, which the provider returns. The raw data varies based on the endpoint.

If you would like to see more information on our Events framework, see [Events Overview](#).

SAP Cloud Platform Open Connectors supports both polling and webhook events for Close.io.

Polling

In order to enable polling, add these extra configurations to your instance JSON:

```
"event.notification.enabled": "true",  
"event.notification.callback.url": "",  
"event.poller.configuration": ""
```

instance JSON with polling events enabled:

```

{
  "connector": {
    "key": "closeio"
  },
  "configuration": {
    "username": "",
    "password": "",
    "event.notification.enabled": "true",
    "event.notification.callback.url": "",
    "event.vendor.type": "polling",
    "event.poller.configuration": {
      "contacts": {
        "url": "/hubs/crm/contacts?where=date_updated>'${gmtDate:yyyy-MM-dd'T'HH:mm:ss.SSSSSSXXX}'",
        "idField": "id",
        "datesConfiguration": {
          "updatedAtField": "date_updated",
          "updatedAtFormat": "yyyy-MM-dd'T'HH:mm:ss.SSSSSSXXX",
          "createdAtField": "date_created",
          "createdAtFormat": "yyyy-MM-dd'T'HH:mm:ss.SSSSSSXXX"
        }
      }
    }
  },
  "tags": [
    ""
  ],
  "name": ""
}

```

Webhooks

Close.io requires an extra specification to be entered once instance has been created. The Close.io Instance ID must be 64 base encoded, then included in the webhook callback URL. This document will walk you through the entire workflow:

- create an instance
- retrieve the instance ID
- 64base Encode Instance ID
- webhook setup

Create Instance

To provision your Close.io Connector, use the /instances API.

Step 1. Call the /instances API

- **HTTP Headers:** Authorization- User , Organization
- **HTTP Verb:** POST
- **Request URL:** /instances
- **Request Body:** Required – see below
- **Query Parameters:** none

Description: token is returned upon successful execution of this API. This token needs to be retained by the application for all subsequent requests involving this connector instance.

A sample request illustrating the /instances API is shown below.

HTTP Headers:

```
Authorization: User , Organization
```

This instance.json file must be included with your instance request. Please fill your information to provision. The “key” into SAP Cloud Platform Open Connectors Close.io is “closeio”. This will need to be entered in the “key” field below depending on which Connector you wish to instantiate.

```
{
  "connector": {
    "key": "closeio"
  },
  "configuration": {
    "username": "",
    "password": "",
    "event.notification.enabled": "true",
    "event.notification.callback.url": "",
    "event.vendor.type": "webhook",
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

Here is an example cURL command to create an instance using /instances API.

Example Request:

```
curl -X POST
-H 'Authorization: User , Organization '
-H 'Content-Type: application/json'
-d @instance.json
'https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instances'
```

If the user does not specify a required config entry, an error will result notifying her of which entries she is missing.

Below is a successful JSON response:

```

{
  "id": 12,
  "name": "Test",
  "token": "cuED0/DezalhhzK2OtO6kMP7NvPnMyNcRDZc=",
  "element": {
    "id": 146,
    "name": "Close.io Beta",
    "key": "closeio",
    "description": "Add a Close.io CRM Instance to connect your existing Close.io account to the CRM Hub, allowing you to manage contacts, accounts, opportunities, etc. across multiple CRM Connectors. You will need your Close.io CRM account information to add an instance.",
    "image": "https://pbs.twimg.com/profile_images/497905627948015616/vNCOkr1Z.png",
    "active": true,
    "deleted": false,
    "typeOauth": false,
    "trialAccount": false,
    "transformationsEnabled": true,
    "authentication": {
      "type": "basic"
    },
    "hub": "crm",
    "parameters": []
  },
  "provisionInteractions": [],
  "valid": true,
  "disabled": false,
  "maxCacheSize": 0,
  "cacheTimeToLive": 0,
  "configuration": {
    "base.url": "https://app.close.io/api/v1",
    "pagination.offset": "true",
    "password": "CLOSE_IO_PASSWRD",
    "pagination.max": "100",
    "username": "CLOSE_IO_API_KEY"
  },
  "eventsEnabled": false,
  "cachingEnabled": false
}

```

Note: Make sure you have straight quotes in your JSON files and cURL commands. Please use plain text formatting in your code. Make sure you do not have spaces after the in the cURL command.

Retrieve Instance ID from the create instance response.

Copy the Instance ID and go to a 64Base Encoding website.

Here is a sample site that will 64 base encode your instance ID:

<https://www.base64encode.org/>.

Copy the ID, encode it, then copy the encoded ID.

Place the ID in the following URL:

```
https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/events/closeio/{INSERT_64BASE_ENCODED_INSTANCE_ID}
```

An example of the URL once the Instance ID has been encoded:

```
https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/events/closeio/MjA5MzE=
```

Currently, webhooks cannot be setup via the Close.io UI. Close.io requests an email be sent with the URL of webhook and they will POST data to that endpoint.

Excerpt from the [Close.io Developer Docs](#) :

Webhooks

Send us an email at support@close.io with the URL of your webhook and we will POST the data to that endpoint for each created/updated lead. Here's an example payload:

```
{
  'model': 'lead',
  'event': 'create', # create|update|delete|merge
  'data': { ... } # serialized object, same as the response of `/lead/{id}/`. In the case of a delete it will just be the 'id' of the object. A merge event will have the source_id and destination_id in the payload.
}
```

A webhook request is attempted up to 3 times with retries taking place every 5 minutes.

End Excerpt

Include the following URL with 64Base encoded Instance ID in your email message to Close.io:

```
https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/events/closeio/{INSERT_64BASE_ENCODED_INSTANCE_ID}
```

