HubSpot CRM Events

Last Modified on 03/19/2020 6:41 pm EDT

SAP Cloud Platform Open Connectors supports events via polling or webhooks depending on the API provider. For more information about our Events framework, see Events Overview.

Supported Events and Resources

SAP Cloud Platform Open Connectors supports both webhooks and polling events for HubSpot CRM.

You can set up events for the following resources:

- Accounts
- Contacts
- Other objects that include `created`, `updated`, and `deleted` data.

> ℹ **Note:** Even if you enable events to be polled for just one object, Hubspot sends event notifications for all objects. For example, Hubspot creates an **account** whenever a **contact** is created. This is why you always get more than one event for an action on a contact.

Configure Polling Through the UI

> ℹ **Note:** Unless configured for a specific time zone, polling occurs in UTC.

To configure polling through the UI, follow the same steps to authenticate a connector instance, and then turn on events. Select the resources to poll, and then click **Create Instance**. For more information, see Authenticate an Connector Instance with Events (UI) or the connector-specific authentication topic.

Configure Polling Through API

To add polling when authenticating through the `/instances` API call, add the following to the `configuration` object in the JSON body. For more information about each parameter described here, see Parameters.

```
{
"event.notification.enabled": true,
"event.vendor.type": "polling",
"event.notification.callback.url": "",
"event.notification.signature.key": "",
"event.objects": "",
"event.poller.refresh_interval": ""
}
```

Note: `event.notification.signature.key` is optional.

Example JSON with Polling

instance JSON with polling events enabled:

```
{
  "element": {
    "key": "hubspotcrm"
  },
  "providerData": {
    "code": ""
  },
  "configuration": {
    "authentication.type": "oauth2 OR apiKey",
    "oauth.callback.url": "",
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "hubspot.authorization.apikey":"",
    "create.bulk.properties": "false",
    "filter.response.nulls": true,
    "event.notification.enabled": true,
    "event.vendor.type": "polling",
    "event.notification.callback.url": "https://mycoolapp.com",
    "event.notification.signature.key": "12345",
    "event.poller.refresh_interval": "5",
    "event.poller.configuration":{
      "accounts":{
        "url":"/hubs/crm/contacts?where=lastmodifieddate='${date}'",
        "idField":"vid",
        "filterByUpdatedDate":true,
        "datesConfiguration":{
          "updatedDateField":"properties.lastmodifieddate",
          "updatedDateFormat":"milliseconds",
          "createdDateField":"properties.createdate",
          "createdDateFormat":"milliseconds",
          "createdCheckTolerance": 10
        }
      }
    }
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

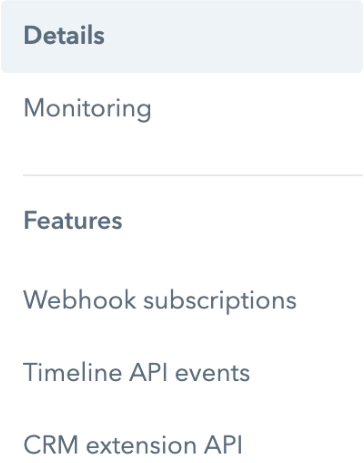
Webhooks

You can configure webhooks through the UI or through API in the JSON body of the `/instances` API call. First, you must set up webhooks in HubSpot.

Set Up Webhooks

Follow these steps to set up your Hubspot application with the endpoint.

1. Via a web browser, log in to your Hubspot developer account at https://app.hubspot.com/signup/developers.
2. In the developer app dashboard, select the app that you'd like to send webhooks for.
3. Select **Webhook Subscriptions** from the navigation items displayed on the left.

   Details

   Monitoring

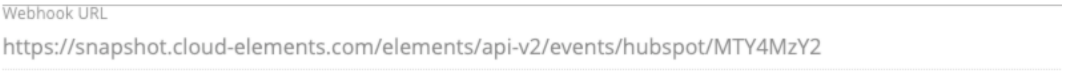
   Features

   Webhook subscriptions

   Timeline API events

   CRM extension API

   The Webhooks page appears, where you see a field to enter the target URL.
4. In order to get this URL, create an instance on HubSpot Marketing connector.
5. While creating an instance, enable **Events** under **Events Configuration** and select **Webhook** as the **Vendor Event Type**.
6. Click **Create Instance.**
7. Choose an account through which you'd like to create an instance.
8. You're now taken back to SAP Cloud Platform Open Connectors. Select **Instances** on the navigation pane on your left and edit your instance.
9. You will see a field named Webhook URL as shown below.

   Webhook URL

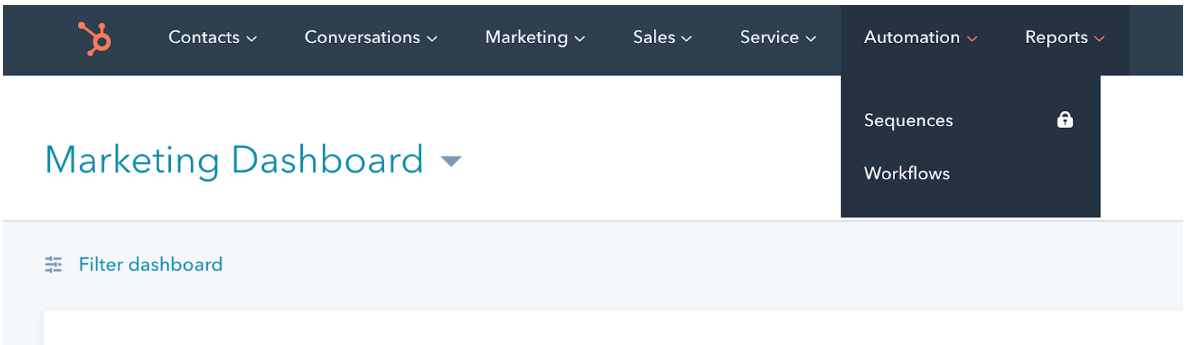
   https://snapshot.cloud-elements.com/elements/api-v2/events/hubspot/MTY4MzY2

   You can now copy the URL, complete step 4 and click **Save.**
10. Next, you'll need to set up individual subscriptions for each object and event type for which you'd like to receive a notification. Do this by clicking **Create Subscription**.
11. Now, follow the prompts to configure the subscription.

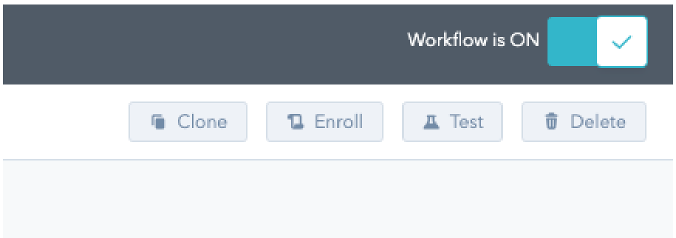
Set up Workflow Webhooks

Follow these steps to set up your Hubspot application with the endpoint.

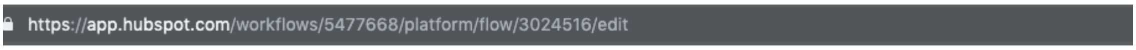
1. To create a workflow webhook, go to https://app.hubspot.com/ and select the account through which you'd like to create an instance with SAP Cloud Platform Open Connectors.
2. On the HubSpot Reports Dashboard, navigate to **Automation** > **Workflows** and click **Create Workflow**.



3. Set enrollment triggers and add an action. Make sure that workflow is **on in the top-right corner.**



4. Once you've created a workflow, you should be be to see the workflow ID in the URL.


https://app.hubspot.com/workflows/5477668/platform/flow/3024516/edit

In the above example, **3024516** is the workflow ID.

5. Create an instance and enter this workflow ID in the corresponding field.

Your workflow webhook is now created.

Configure Webhooks Through the UI

To configure webhooks through the UI, follow the same steps to authenticate a connector instance, and then turn on events. For more information, see Authenticate an Connector Instance with Events (UI) or the connector-specific authentication topic.

Configure Webhooks Through API

To add webhooks when authenticating through the `/instances` API call, add the following to the `configuration` object in the JSON body. For more information about each parameter described here, see Parameters.

```
{
"event.notification.enabled": true,
"event.vendor.type": "polling",
"event.notification.callback.url": "",
"event.notification.signature.key": "",
"event.objects": ""
}
```

> **Note:** `event.notification.signature.key` is optional.

Example JSON with Webhooks

Instance JSON with webhooks events enabled:

```
{
  "element": {
    "key": "hubspotcrm"
  },
  "providerData": {
    "code": ""
  },
  "configuration": {
    "authentication.type": "oauth2 OR apiKey",
    "oauth.callback.url": "",
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "hubspot.authorization.apikey":"",
    "create.bulk.properties": "false",
    "filter.response.nulls": true,
    "event.notification.enabled": true,
    "event.vendor.type": "webhooks",
    "event.notification.callback.url": "https://mycoolapp.com",
    "event.notification.signature.key": "12345",
    "event.objects": "Contact,Account"
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

Parameters

API parameters are in `code formatting`.

Parameter	Description	Data Type
`key`	The connector key. hubspotcrm	string
`code`	The authorization grant code returned from the API provider in an OAuth2 authentication workflow.	string

Parameter	Description	Data Type
Name `name`	The name for the connector instance created during authentication.	string
`authentication.type`	Identifies how you are authenticating with HubSpot CRM. Either `oauth2` or `apiKey` .	string
Create Bulk Properties for Migration `create.bulk.properties`	Identifies if you want to create custom properties in Hubspot for bulk uploads.	string, must be `true` (Yes) or `false` (No)
`oauth.callback.url`	OAuth 2.0 authentication only. The URL where you want to redirect users after they grant access. This is the **Callback URL** that you noted in API Provider Setup.	string
`oauth.api.key`	OAuth 2.0 authentication only. The Client ID from HubSpot CRM. This is the **Client ID** that you noted in API Provider Setup.	string
`oauth.api.secret`	OAuth 2.0 authentication only. The Client Secret from HubSpot CRM. This is the **Client Secret** that you noted in API Provider Setup.	string
Hubspot API Key `hubspot.authorization.apikey`	API Key authentication only. The Hubspot API key that you noted in API Provider Setup.	string
Events Enabled `event.notification.enabled`	*Optional*. Identifies that events are enabled for the connector instance. Default: `false`	boolean
Event Type `event.vendor.type`	*Optional*. Identifies the type of events enabled for the instance, either `webhook` or `polling` .	string
Event Notification Callback URL `event.notification.callback.url`	*For webhooks and polling.* The URL where your app can receive events.	string
	For webhooks and polling.	

Parameter	Description	Data Type
Callback Notification Signature Key `event.notification.signature.key`	*Optional* A user-defined key for added security to show that events have not been tampered with. This can be any custom value that you want passed to the callback handler listening at the provided Event Notification Callback URL.	string
Objects to Monitor for Changes `event.objects`	*For webhooks and polling.* *Optional* Comma separated list of objects to monitor for changes.	string
Event poller refresh interval (mins) `event.poller.refresh_interval`	*For polling only.* A number in minutes to identify how often the poller should check for changes.	number
Configure Polling `event.poller.configuration`	Optional*. Configuration parameters for polling.	JSON object
resource name e.g., contact, account, etc.	The configuration of an individual resource.	JSON object
URL `url`	The url to query for updates.	String
ID Field `idField`	The field that is used to uniquely identify an object.	String
Advanced Filtering `datesConfiguration`	Configuration parameters for dates in polling	JSON Object
Updated Date Field `updatedDateField`	The field that identifies an updated object.	String
Updated Date Format `updatedDateFormat`	The date format of the field that identifies an updated object.	String
Created Date Field `createdDateField`	The field that identifies a created object.	String
Created Date Format `createdDateFormat`	The date format of the field that identifies a created object.	String
tags	*Optional*. User-defined tags to further identify the instance.	string