

Twitter Events

Last Modified on 03/19/2020 9:32 pm EDT

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, please see [Events Overview](#).

Things to be aware of when enabling events for Twitter:

1. There are no updated dates so all `event_types` are `CREATED`
2. For `statuses` and `users`, there are some things you have to specify the username or screenname
3. And you can't query by `created_time` so we are just querying the full thing and then parsing through the response, (potentially super messy)

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:

Twitter Follows the OAuth1 Authorization protocol.

For more information about the Twitter API, please view their [API documentation](#).

Step 1. Get Connectors OAuth Token

- HTTP Header: None
- HTTP Verb: GET
- Request URL: /elements/{keyOrId}/oauth/token
- Request Body: None

- Query Parameters:
 - **key** - twitter
 - **apiKey** - the key obtained from registering your app with the provider
 - **apiSecret** - the secret obtained from registering your app with the provider
 - **callbackUrl** - the URL that you supplied to the provider when registering your app

Description: The result of this API invocation returns a requestToken and Secret from the endpoint, which are used to retrieve the redirect URL. The requestToken is used in the GET /elements/{keyOrId}/oauth/url call.

Each of the OAuth API calls will be shown below.

Example cURL Command:

```
curl -X GET
-H 'Content-Type: application/json'
'https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/elements/twitter/oauth/token?apiKey=insert_fake_api_key&apiSecret=insert_fake_api_secret&callbackUrl=https://www.mycoolapp.com/oauth&state=twitter'
```

Response:

```
{
  "token": "qyprd1Twij60MH061KGUZTJwx7tbzpPQx6aZnvKe0xI7",
  "secret": "KKSbL0J2wLGMqhMXTEh1ERkSx9tsIbAHUevF"
}
```

Twitter expects a token and secret. These are contained in the response to the initial GET request. Please make note of the token and secret. The token is needed in the GET /elements/{keyOrId}/oauth/url call which is shown below.

Step 2. Get Connectors OAuth URL

- HTTP Header: None
- HTTP Verb: GET
- Request URL: /elements/{keyOrId}/oauth/url
- Request Body: None

- Query Parameters:
 - **key** - twitter
 - **apiKey** - the key obtained from registering your app with the provider
 - **apiSecret** - the secret obtained from registering your app with the provider
 - **callbackUrl** - the URL that you supplied to the provider when registering your app,
 - **requestToken** - the token obtained from the GET /elements/{keyOrId}/oauth/token call (previous step).

Description: The result of this API invocation is an OAuth redirect URL from the endpoint. Your application should now redirect to this URL, which in turn will present the OAuth authentication and authorization page to the user. When the provided callback URL is executed, a code value will be returned, which is required for the Create Instance API.

Example cURL Command:

```
curl -X GET
-H 'Content-Type: application/json'
'https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/elements/twitter/oauth/url?apiKey=insert_fake_api_key&apiSecret=insert_fake_api_secret&callbackUrl=https://www.mycoolapp.com/oauth&requestToken=insert_fake_request_token&state=twitter'
```

Response:

```
{
  "element": "twitter",
  "oauthUrl": "https://appcenter.intuit.com/Connect/Begin?oauth_token=qyp
rdJHtIbwm3sGOoOCvXuv2Cs8fsQrZFjJWe4HEZAYb0&oauth_callback=http%3A%2F%2Fwww.
cloud-elements.com%3Fstate%3Dtwitter"
}
```

Step 3. Create an Instance

Your application should now redirect to the `oauthUrl` returned in step 2, which in turn will present the OAuth authentication and authorization page to the user.

HANDLE CALLBACK FROM THE ENDPOINT

After the user successfully authenticates, the provided callback URL is executed. The callback URL will contain several parameters, listed below. These additional parameters, along with the original API key and API secret are required for the Create Instance API.

The parameters that you will need to parse from the callback URL are listed below, along with an example of what the callback URL should look like. **oauth_token** **oauth_verifier**

```
https://www.mycoolapp.com/oauth?state=twitter&oauth_token=qyprdlGChTClXwBpAw1vm1fJSC3mQqS3dGX0PPphEzNEUI9s&oauth_verifier=br6qctk
```

These values will be used to create an Instance. An example of this process along with sample JSON will be shown in the next section.

To provision your Twitter connector, use the `/instances` API.

Below is an example of the provisioning API call.

- **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 Twitter is “twitter”. This will

need to be entered in the “key” field below depending on which connector you wish to instantiate.

```
{
  "element": {
    "key": "twitter"
  },
  "providerData": {
    "oauth_token": "",
    "oauth_verifier": "",
    "secret": ""
  },
  "configuration": {
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "oauth.callback.url": "",
    "event.poller.configuration": {
      "direct-messages/received-messages": {
        "url": "/hubs/social/direct-messages/received-messages?where=created_at='${date:MM/dd/yyyy'T'HH:mm:ssXXX}' ",
        "idField": "id_str",
        "datesConfiguration": {
          "updatedAtField": "created_at",
          "updatedAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
          "updatedAtTimezone": "GMT",
          "createdAtField": "created_at",
          "createdAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
          "createdAtTimezone": "GMT"
        }
      },
      "direct-messages/sent-messages": {
        "url": "/hubs/social/direct-messages/sent-messages?where=created_at='${date:MM/dd/yyyy'T'HH:mm:ssXXX}' ",
        "idField": "id_str",
        "datesConfiguration": {
          "updatedAtField": "created_at",
          "updatedAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
          "updatedAtTimezone": "GMT",
          "createdAtField": "created_at",
          "createdAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
          "createdAtTimezone": "GMT"
        }
      }
    },
    "lists": {
      "url": "/hubs/social/lists?where=created_at='${date:MM/dd/yyyy'T'HH:mm:ssXXX}' ",
      "idField": "id_str",
      "datesConfiguration": {
        "updatedAtField": "created_at",
```

```

    "updatedAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "updatedAtTimezone": "GMT",
    "createdAtField": "created_at",
    "createdAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "createdAtTimezone": "GMT"
  }
},
"statuses": {
  "url": "/hubs/social/statuses?where=q=''",
  "idField": "id_str",
  "datesConfiguration": {
    "updatedAtField": "created_at",
    "updatedAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "updatedAtTimezone": "GMT",
    "createdAtField": "created_at",
    "createdAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "createdAtTimezone": "GMT"
  }
},
"users": {
  "url": "/hubs/social/users?where=q=''",
  "idField": "id_str",
  "datesConfiguration": {
    "updatedAtField": "created_at",
    "updatedAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "updatedAtTimezone": "GMT",
    "createdAtField": "created_at",
    "createdAtFormat": "EEE, dd MMM yyyy HH:mm:ss Z",
    "createdAtTimezone": "GMT"
  }
}
}
},
"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": 1234,
  "name": "test",
  "token": "3sU/S/kZC46BaABPS7EAuhT+ukiEHkI=",
  "element": {
    "id": 1359,
    "name": "Twitter",
    "key": "twitter",
    "description": "Add a Twitter Instance to connect your existing Twitter account to the Social Hub, allowing you to manage statuses and followers across multiple Social connectors. You will need your Twitter account information to add an instance.",
    "image": "https://abs.twimg.com/a/1426096855/images/oauth_application.png",
    "active": false,
    "deleted": false,
    "typeOAuth": false,
    "trialAccount": false,
    "resources": [],
    "transformationsEnabled": true,
    "bulkDownloadEnabled": false,
    "bulkUploadEnabled": false,
    "cloneable": true,
    "authentication": {
      "type": "oauth1"
    },
    "hub": "social",
    "protocolType": "http",
    "parameters": [],
    "private": false
  },
  "provisionInteractions": [],
  "valid": true,
  "disabled": false,
  "maxCacheSize": 0,
```

```
"cacheTimeToLive": 0,
"configuration": {
  "base.url": "https://api.twitter.com/1.1",
  "oauth.api.secret": "DYLgpG9rFNzW34X0LU4Ik2ZSgvK0rkltOniDKk4GzAy6kqrsXi",
  "oauth.token.url": "https://api.twitter.com/oauth/access_token",
  "oauth.user.token.secret": "lPibXGVLtgURx9TwJcstN9dZkGBEriO2VyT7aicXbmWex",
  "pagination.max": "100",
  "event.vendor.type": "polling",
  "oauth.request.url": "https://api.twitter.com/oauth/request_token",
  "oauth.user.token": "794566698309472260-Ro9v0o4q8HXgmyS5aoOqs2mwQHK3Bga",
  "oauth.authorization.url": "https://api.twitter.com/oauth/authorize",
  "pagination.type": "page",
  "event.poller.refresh_interval": "15",
  "event.notification.callback.url": null,
  "oauth.request.authorization.type": "query",
  "oauth.callback.url": "http://localhost:8080/elements/jsp/home.jsp",
  "oauth.api.key": "kN7fdmMLD0XMU2Fqvbadww8mi",
  "event.poller.configuration": "{ \"direct-messages/received-messages\": {
  \"url\": \"/hubs/social/direct-messages/received-messages?where=created_at='
  ${date:MM/dd/yyyy'T'HH:mm:ssXXX}'\", \"idField\": \"id_str\", \"datesConfigurat
  ion\": { \"updatedAtField\": \"created_at\", \"updatedAtDateFormat\": \"EEE, dd
  MMM yyyy HH:mm:ss Z\", \"updatedAtTimezone\": \"GMT\", \"createdDateField\":
  \"created_at\", \"createdDateFormat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"crea
  tedDateTimezone\": \"GMT\" }}, \"direct-messages/sent-messages\": { \"url\": \"/h
  uhs/social/direct-messages/sent-messages?where=created_at='${date:MM/dd/yyyy
  'T'HH:mm:ssXXX}'\", \"idField\": \"id_str\", \"datesConfiguration\": { \"updated
  DateField\": \"created_at\", \"updatedAtDateFormat\": \"EEE, dd MMM yyyy HH:mm:s
  s Z\", \"updatedAtTimezone\": \"GMT\", \"createdDateField\": \"created_at\", \
  \"createdDateFormat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"createdDateTimezone\
  \": \"GMT\" }}, \"lists\": { \"url\": \"/hubs/social/lists?where=created_at='${dat
  e:MM/dd/yyyy'T'HH:mm:ssXXX}'\", \"idField\": \"id_str\", \"datesConfiguration\"
  : { \"updatedAtField\": \"created_at\", \"updatedAtDateFormat\": \"EEE, dd MMM y
  yyy HH:mm:ss Z\", \"updatedAtTimezone\": \"GMT\", \"createdDateField\": \"cre
  ated_at\", \"createdDateFormat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"createdDa
  teTimezone\": \"GMT\" }}, \"statuses\": { \"url\": \"/hubs/social/statuses?where=
  q='Example Screen Name'\", \"idField\": \"id_str\", \"datesConfiguration\": { \"
  updatedAtField\": \"created_at\", \"updatedAtDateFormat\": \"EEE, dd MMM yyyy
  HH:mm:ss Z\", \"updatedAtTimezone\": \"GMT\", \"createdDateField\": \"created
  _at\", \"createdDateFormat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"createdDateTi
  mezone\": \"GMT\" }}, \"users\": { \"url\": \"/hubs/social/users?where=q='example
  '\", \"idField\": \"id_str\", \"datesConfiguration\": { \"updatedAtField\": \"c
  reated_at\", \"updatedAtDateFormat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"updated
  DateTimezone\": \"GMT\", \"createdDateField\": \"created_at\", \"createdDateFor
  mat\": \"EEE, dd MMM yyyy HH:mm:ss Z\", \"createdDateTimezone\": \"GMT\" } } }\",
  "pagination.page.startindex": "0",
  "event.notification.enabled": "false"
},
```



```
"eventsEnabled": false,  
"traceLoggingEnabled": false,  
"cachingEnabled": false,  
"externalAuthentication": "none",  
"user": {  
  "id": 9723  
}  
}
```

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.
