Citrix Sharefile Events

Last Modified on 03/26/2020 2:53 am 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 polling events and webhooks for Citrix Sharefile. After receiving an event, SAP Cloud Platform Open Connectors standardizes the payload and sends an event to the configured callback URL of your authenticated connector instance.

Polling

You can set up polling for the `events` resource. You can also copy the `events` configuration to poll other resources. See Configure Polling Through API for more information.

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

Configure Polling Through the UI

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

Use the `/instances` endpoint to authenticate with Citrix ShareFile and create a connector instance with polling enabled.

> **Note:** The endpoint returns a connector instance token and id upon successful completion. Retain the token and id for all subsequent requests involving this connector instance.

To authenticate a connector instance with polling:

1. Get an authorization grant code by completing the steps in **Getting a redirect URL** and **Authenticating users and receiving the authorization grant code in** Citrix Sharefile Authenticate a Connector Instance.

2. Construct a JSON body as shown below (see Polling Parameters):

```
{
  "element":{
    "key":"sharefile"
  },
  "providerData":{
    "code": ""
  },
  "configuration":{
    "oauth.callback.url": "",
    "oauth.api.key": "",
    "oauth.api.secret": "",
    "document.root.folder.name": "",
    "sharefile.subdomain": "",
    "event.notification.enabled": true,
    "event.vendor.type": "polling",
    "event.notification.callback.url": "http://mycoolapp.com",
    "event.notification.signature.key": "123456",
    "event.poller.refresh_interval": "",
    "event.poller.configuration":{
      "documents":{
        "url":"/hubs/documents/events/poll/documents?where=lastmodifieddate='${gmtDate:yyyy-MM-dd'T'HH:mm:ss'Z'}'",
        "idField":"id",
        "datesConfiguration":{
          "updatedDateField":"CreationDate",
          "updatedDateFormat": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
          "updatedDateTimezone": "GMT",
          "createdDateField": "ProgenyEditDate",
          "createdDateFormat": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
          "createdDateTimezone": "GMT"             }
      }
    }
  },
  "tags":[
    ""
  ],
  "name":""
}
```

3. Call the following, including the JSON body you constructed in the previous step:

```
POST /instances
```

> **Note:** Make sure that you include the User and Organization keys in the header. For more information, see Authorization Headers, Organization Secret, and User Secret.

4. Locate the `token` and `id` in the response and save them for all future requests using the connector instance.

Example cURL with Polling

```
curl -X POST \
https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instan
ces \
-H 'authorization: User , Organization ' \
-H 'content-type: application/json' \
-d '{
"element": {
  "key": "sharefile"
},
"providerData":{
  "code": ""
},
"configuration": {
    "oauth.callback.url": "https;//mycoolapp.com",
    "oauth.api.key": "xxxxxxxxxxxxxxxxxx",
    "oauth.api.secret": "xxxxxxxxxxxxxxxxxxxxxxxx",
    "document.root.folder.name": "/top",
    "sharefile.subdomain": "cloud-elements",
    "event.notification.enabled": true,
    "event.vendor.type": "polling",
    "event.notification.callback.url": "",
    "event.poller.refresh_interval": "15",
    "event.poller.configuration":{
        "documents": {
        "url": "/hubs/documents/events/poll/documents?where=lastmodifiedda
te='${gmtDate:yyyy-MM-dd'T'HH:mm:ss'Z'}'",
        "idField": "Id",
        "datesConfiguration": {
          "updatedDateField": "CreationDate",
          "updatedDateFormat": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
          "updatedDateTimezone": "GMT",
          "createdDateField": "ProgenyEditDate",
          "createdDateFormat": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
          "createdDateTimezone": "GMT"
            }
        }
    }
  },
  "tags": [
    "Docs"
  ],
  "name": "API Instance with Polling"
}'
```

Polling Parameters

API parameters not shown in SAP Cloud Platform Open Connectors are in `code formatting` .

Parameter	Description	Data Type
`key`	The connector key. sharefile	string
`code`	The authorization grant code returned from the API provider in an OAuth 2.0 authentication workflow. SAP Cloud Platform Open Connectors uses the code to retrieve the OAuth access and refresh tokens from the endpoint.	string
Name `name`	The name of the connector instance created during authentication.	Body
`oauth.api.key`	The API key or client ID obtained from registering your app with the provider. This is the **Client Id** that you noted in API Provider Setup.	string
`oauth.api.secret`	The client secret obtained from registering your app with the API provider. This is the **Client Secret** that you noted in API Provider Setup.	string
`oauth.callback.url`	The API key or client ID obtained from registering your app with the provider. This is the **Redirect URI** that you noted in API Provider Setup.	
Sharefile Root Folder `document.root.folder.name`	The root folder. The /top folder is the root level folder.	string
Subdomain `sharefile.subdomain`	The subdomain part of your Sharefile url. For example, if your url is `https://cloud-elements.sharefile.com` enter `cloud-elements` .	string
Events Enabled `event.notification.enabled`	*Optional*. Identifies that events are enabled for the connector instance. Default: `false` .	boolean
Event Notification Callback URL	The URL where you want SAP Cloud	

Parameter	Description	Data Type
`event.notification.callback.url`	Platform Open Connectors to send the events.	string
Event poller refresh interval (mins) `event.poller.refresh_interval`	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 to Poll	The polling event configuration of the resource that you will monitor.	JSON object
URL `url`	The url to query for updates to the resource.	String
ID Field `idField`	The field in the resource that is used to uniquely identify it.	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

Webhooks

Webhooks utilize existing functionality at the API provider and typically require additional setup. See your API provider documentation for details.

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

Use the `/instances` endpoint to authenticate with Citrix ShareFile and create a connector instance with webhooks enabled.

> ℹ **Note:** The endpoint returns a connector instance token and id upon successful completion. Retain the token and id for all subsequent requests involving this connector instance.

To authenticate a connector instance with webhooks:

1. Get an authorization grant code by completing the steps in **Getting a redirect URL** and **Authenticating users and receiving the authorization grant code in** Citrix Sharefile Authenticate a Connector Instance.

2. Construct a JSON body as shown below (see Webhook Parameters):

```
{
  "element": {
    "key": "sharefile"
  },
  "providerData": {
    "code": ""
  },
  "configuration": {
    "oauth.callback.url": "https;//mycoolapp.com",
    "oauth.api.key": "xxxxxxxxxxxxxxxxxx",
    "oauth.api.secret": "xxxxxxxxxxxxxxxxxxxxxxxx",
    "document.root.folder.name": "/top",
    "sharefile.subdomain": "cloud-elements",
    "event.notification.enabled": true,
    "event.vendor.type": "webhook",
    "event.notification.callback.url": "",
    "event.notification.signature.key": ""
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

3. Call the following, including the JSON body you constructed in the previous step:

```
POST /instances
```

> ℹ **Note:** Make sure that you include the User and Organization keys in the header. For more information, see Authorization Headers, Organization Secret, and User Secret.

4. Locate the `token` and `id` in the response and save them for all future requests using the connector instance.

Example cURL

```
curl -X POST \
  https://api.cloud-elements.com/elements/api-v2/instances \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '{
  "element": {
    "key": "sharefile"
  },
  "providerData": {
    "code": "xoz8AFqScK2ngM04kSSM"
  },
  "configuration": {
    "oauth.callback.url": "https;//mycoolapp.com",
    "oauth.api.key": "xxxxxxxxxxxxxxxxxx",
    "oauth.api.secret": "xxxxxxxxxxxxxxxxxxxxxxxx",
    "document.root.folder.name": "/top",
    "sharefile.subdomain": "cloud-elements",
    "event.notification.enabled": true,
    "event.vendor.type": "webhook",
    "event.notification.callback.url": "https://mycoolapp.com/events",
    "event.notification.signature.key": "xxxxxxxxxxxxxxxxxxxxxxxxx"
  },
  "tags": [
    "Docs"
  ],
  "name": "API Instance"
}'
```

Webhook Parameters

API parameters not shown in the SAP Cloud Platform Open Connectors are in `code formatting`.

Parameter	Description	Data Type
`key`	The connector key. sharefile	string
`code`	The authorization grant code returned from the API provider in an OAuth 2.0 authentication workflow. SAP Cloud Platform Open Connectors uses the code to retrieve the OAuth access and refresh tokens from the endpoint.	string
Name `name`	The name of the connector instance created during authentication.	Body
`oauth.api.key`	The API key or client ID obtained from registering your app with the provider. This is the Client Id that you recorded in API Provider Setup.	string
`oauth.api.secret`	The client secret obtained from registering your app with the API provider. This is the Client Secret that you recorded in API Provider Setup.	string
`oauth.callback.url`	The URL that the API provider returns a user to after they authorize access. This is the Redirect URI that you recorded in API Provider Setup.	
Sharefile Root Folder `document.root.folder.name`	The root folder. The /top folder is the root level folder.	string
Subdomain `sharefile.subdomain`	The subdomain part of your Sharefile url. For example, if your url is `https://cloud-elements.sharefile.com` enter `cloud-elements` .	string
Events Enabled `event.notification.enabled`	Optional. Identifies that events are enabled for the connector instance. Default: `false` .	boolean
Event Notification Callback URL `event.notification.callback.url`	The URL where you want SAP Cloud Platform Open Connectors to send the events.	string
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.	string
tags	Optional. User-defined tags to further identify the instance.	string