

Zuora v2 Authenticate a Connector

Last Modified on 03/16/2020 10:25 pm EDT

On this page

You can authenticate with Zuora to create your own instance of the Zuora v2 connector through the UI or through APIs. Once authenticated, you can use the connector instance to access the different functionality offered by the Zuora platform.

You can authenticate a Zuora v2 connector instance in one of two ways:

- Basic authentication where you supply the user name and password of the Zuora user.
- Custom where you provide the **Client ID** and **Client Secret** of the OAuth client associated with the user.

The authentication through the UI instructions are the same below, but if you authenticate via API, make sure that you follow the steps in the correct section: [Custom Authenticate Through API](#) and [Basic Authenticate Through API](#).

Authenticate Through the UI

Use the UI to authenticate with Zuora and create a connector instance. If you are authenticating with Zuora via custom OAuth 2.0, all you need to do is add a name for the instance. After you create the instance, you'll log in to Zuora to authorize SAP Cloud Platform Open Connectors access to your account. For more information about authenticating a connector instance, see [Authenticate a Connector Instance \(UI\)](#).

After successfully authenticating, we give you several options for next steps. [Make requests using the API docs](#) associated with the instance, [map the instance to a common resource](#), or [use it in a formula template](#).

Custom Authenticate Through API

Authenticating through API is similar to authenticating via the UI. Instead of clicking and typing through a series of buttons, text boxes, and menus, you will instead send a request to our `/instances` endpoint. The end result is the same, though: an authenticated connector

instance with a **token** and **id**.

To authenticate a connector instance:

1. Construct a JSON body as shown below (see [Parameters](#)):

```
{
  "element": {
    "key": "zuorav2"
  },
  "configuration": {
    "zuorav2.sandbox": "",
    "authentication.type": "custom",
    "clientId": "Zuora Client ID",
    "clientSecret": "Zuora Client Secret"
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

2. 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](#).

3. 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.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instances \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '{
    "element": {
      "key": "zuorav2"
    },
    "configuration": {
      "zuorav2.sandbox": "false",
      "authentication.type": "custom",
      "clientId": "a344xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "clientSecret": "dUwxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
    },
    "tags": [
      "Docs"
    ],
    "name": "API Instance"
  }'
```

Basic Authenticate Through API

Authenticating through API is similar to authenticating via the UI. Instead of clicking and typing through a series of buttons, text boxes, and menus, you will instead send a request to our `/instances` endpoint. The end result is the same, though: an authenticated connector instance with a **token** and **id**.

To authenticate a connector instance:

1. Construct a JSON body as shown below (see [Parameters](#)):

```
{
  "element": {
    "key": "zuorav2"
  },
  "configuration": {
    "zuorav2.sandbox": "",
    "authentication.type": "basic",
    "username": "",
    "password": ""
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

2. 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](#).

3. 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.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instances \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '{
  "element": {
    "key": "zuorav2"
  },
  "configuration": {
    "zuorav2.sandbox": "false",
    "authentication.type": "basic",
    "username": "xxxxxxxxxxxxxxxxxxxx",
    "password": "xxxxxxxxxxxxxxxxxxxx"
  },
  "tags": [
    "Docs"
  ],
  "name": "API Instance"
}'

```

Parameters

API parameters not shown in SAP Cloud Platform Open Connectors are in

`code formatting` .

Note: Event related parameters are described in [Events](#).

Parameter	Description	Data Type
<code>key</code>	The connector key. zuorav2	string
Name <code>name</code>	The name of the connector instance created during authentication.	string
Connect to Sandbox <code>zuorav2.sandbox</code>	Indicates whether to connect to a sandbox account (<code>true</code>) or a production account (<code>false</code>).	boolean
Authentication Type <code>authentication.type</code>	Determines whether authentication requires a user name and password (Basic) or Client ID and Client Secret.	string
Username <code>username</code>	The Zuora v2 User Name that you noted in API Provider Setup .	string

Parameter <code>password</code>	Description The Zuora v2 Password that you noted in API Provider Setup .	Data Type string
Client ID <code>clientId</code>	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
Client Secret <code>clientSecret</code>	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
tags	<i>Optional.</i> User-defined tags to further identify the instance.	string

Example Response for an Authenticated Connector Instance

In this example, the instance ID is `12345` and the instance token starts with "ABC/D...". The actual values returned to you will be unique: make sure you save them for future requests to this new instance.

```
{
  "id": 12345,
  "name": "API Instance",
  "createdDate": "2017-08-07T18:46:38Z",
  "token": "ABC/Dxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
  "element": {
    "id": 2245,
    "name": "Zuora v2",
    "hookName": "ZuoraV2",
    "key": "zuorav2",
    "description": "Add an Instance of Zuora to get started with Zuora to the Payment Hub, allowing you to manage customers, invoices, products, payments, etc. across multiple Payment connectors. You will need your Zuora account to create an instance.",
    "image": "elements/provider_zuorav2.png",
    "active": true,
    "deleted": false,
    "typeOauth": false,
    "trialAccount": false,
    "resources": [ ],
    "transformationsEnabled": true,
    "bulkDownloadEnabled": true,
    "bulkUploadEnabled": true,
    "cloneable": true,
    "extendable": true,
    "beta": false,
    "authentication": {
```

```
    "type": "basic"
  },
  "extended": false,
  "hub": "payment",
  "protocolType": "http",
  "parameters": [],
  "private": false
},
"elementId": 2245,
"tags": [
  "Docs"
],
"provisionInteractions": [],
"valid": true,
"disabled": false,
"maxCacheSize": 0,
"cacheTimeToLive": 0,
"configuration": {  },
"eventsEnabled": false,
"traceLoggingEnabled": false,
"cachingEnabled": false,
"externalAuthentication": "none",
"user": {
  "id": 12345
}
}
```