NetSuite CRM Authenticate a Connector

On this page

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

Due to changes to upcoming NetSuite requirements, we recommend using custom authentication to ensure proper functionality.

Authenticate Through the UI

Use the UI to authenticate with Oracle and create a connector instance as described in Authenticate a Connector Instance (UI). In addition to the base authentication parameters, you will need the credentials that you identified in API Provider Setup.

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 object, or use it in a formula template.

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 API calls to our `/instances` endpoint. The end result is the same, though: an authenticated connector instance with a **token** and **id** for future API calls.

If you are configuring events, see the Events section.

To create a connector instance:

1. Construct a JSON body based on your authentication type, either Basic or Custom. See Parameters for detailed descriptions.

   Basic Authentication

```
{
  "element": {
    "key": "netsuitecrmv2"
  },
  "configuration": {
    "netsuite.sandbox": false,
    "netsuite.accountId": "my_account_id",
    "netsuite.single.session": true,
    "netsuite.single.session.key": "my_unique_key",
    "authentication.type": "Basic",
    "user.username": "my@somewhere.com",
    "user.password": "my_secret_password",
    "netsuite.appId": "my_app_id",
    "event.notification.enabled": false
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

Custom Authentication

```
{
  "element": {
    "key": "netsuitecrmv2"
  },
  "configuration": {
    "netsuite.sandbox": false,
    "netsuite.accountId": "my_account_id",
    "netsuite.single.session": true,
    "netsuite.single.session.key": "my_unique_key",
    "authentication.type": "custom",
    "consumer_key": "consumer_key_1234567",
    "consumer_secret": "secret_1234567",
    "token_id": "token_1234",
    "token_secret": "token_secret_1234",
    "filter.response.nulls": true,
    "event.notification.enabled": false
  },
  "tags": [
    ""
  ],
  "name": ""
}
```

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

> ⓘ **Note:** Make sure that you include your 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

Basic Authentication

```
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": "netsuitecrmv2"
  },
  "configuration": {
    "netsuite.sandbox": false,
    "netsuite.accountId": "my_account_id",
    "netsuite.single.session": true,
    "netsuite.single.session.key": "my_unique_key",
    "authentication.type": "Basic",
    "user.username": "my@somewhere.com",
    "user.password": "my_secret_password",
    "netsuite.appId": "my_app_id",
    "event.notification.enabled": false
  },
  "tags": [
    "Accounting"
  ],
  "name": "NetSuite Instance 1"
}'
```

Custom Authentication

```
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": "netsuitecrmv2"
  },
  "configuration": {
    "netsuite.sandbox": false,
    "netsuite.accountId": "my_account_id",
    "netsuite.single.session": true,
    "netsuite.single.session.key": "my_unique_key",
    "authentication.type": "custom",
    "consumer_key": "consumer_key_1234567",
    "consumer_secret": "secret_1234567",
    "token_id": "token_1234",
    "token_secret": "token_secret_1234",
    "filter.response.nulls": true,
    "event.notification.enabled": false
  },
  "tags": [
    "Accounting"
  ],
  "name": "NetSuite Instance 1"
}'
```

Parameters

See NetSuite CRM API Provider Setup for directions on how to obtain authentication parameters.

> ℹ Note: Event related parameters are described in Events.

Parameter	Description	Data Type	Required
'key'	The connector key. netsuitecrmv2	string	Y
Name `name`	The name for the connector instance created during authentication.	Body	Y
Connect TO Sandbox `netsuite.sandbox`	Whether to use a NetSuite sandbox	boolean	Y
Account ID `netsuite.accountId`	The NetSuite account ID. See NetSuite CRM API Provider Setup for directions on how to retrieve.	string	Y
Enforce Single Session `netsuite.single.session`	whether to use single session to limit API requests to this account	boolean	N
Specify Single Session Lock Key `netsuite.single.session.key`	the unique key for this single session (only applicable if single session = true)	Y if enforcing	
Authentication Type `authentication.type`	the authentication type, which must be either `basic` or `custom`	string	Y
Email `user.username`	the email of a NetSuite authenticated user (only applicable for `Basic` authentication)	string	Y if Basic Authentication
User Password `user.password`	the password of a NetSuite authenticated user (only applicable for `Basic` authentication)	string	Y if Basic Authentication
App ID `netsuite.appId`	The App ID of a NetSuite integration (only applicable for `Basic` authentication). Complete these steps to find the App ID for your integration: 1. Log into Netsuite. 2. Navigate to Setup. 3. Click Integrations. 4. Click Manage Integrations, and then click New. 5. On the New Integration page, name your application instance. Ensure that "Token Based Authentication" is unchecked and that "User Authentication" is checked. 6. Click Save. Your Application ID is then generated.	string	Y if Basic Authentication
Consumer Key `consumer_key`	For Custom authentication, the consumer key of a token-based NetSuite integration.	string	Y if Token Based Authentication
Consumer Secret `consumer_secret`	For Custom authentication, the consumer secret of a token-based NetSuite integration.	string	Y if Token Based Authentication
Access Token ID `token_id`	For Custom authentication, the token ID of a token-based NetSuite integration.	string	Y if Token Based Authentication
Access Token Secret	For Custom authentication, the token secret of a	string	Y if Token Based

Parameter	Description	Data Type	Required
`token_secret`	token-based NetSuite integration.		Authentication
Filter null values from the response `filter.response.nulls`	whether or not to filter out *null* values from the response	boolean	N

Example Response

In this example, the authenticated connector instance ID is `12345` and token starts with `ABC/D...`. Your values will be different. Make sure that you save the `id` and `token` for future requests.

```json
{
    "id": 12345,
    "name": "NetSuite Instance 1",
    "token": "ABC/D/efgHIJK1234lmnopQRS+1tuVWx+yz98765",
    "element": {
        "id": 988,
        "name": "NetSuite Instance 1",
        "key": "netsuitecrmv2",
        "description": "NetSuite CRM.",
        "active": true,
        "deleted": false,
        "typeOauth": true,
        "trialAccount": false,
        "configDescription": "NetSuite CRM",
        "signupURL": ""
    },
    "provisionInteractions": [],
    "valid": true,
    "eventsEnabled": true,
    "disabled": false,
    "maxCacheSize": 0,
    "cacheTimeToLive": 0,
    "cachingEnabled": false
}
```