

Salesforce Service Cloud data connector

Last updated by | Naser Mohammed | Mar 19, 2023 at 7:18 AM CDT

Contents


- [About the Salesforce Service Cloud solution](#)
- [Account Setup](#)
 - [Setup your Salesforce developer Account](#)
 - [Confirming "API Enabled" permissions.](#)
 - [Creating a Connected App.](#)
 - [Gettin a Consumer Key and a Consumer Secret.](#)
 - [Generate a Security Token](#)
 - [Additional Note](#)
- [Salesforce common Errors:](#)
 - [Token getting failed. Exiting program. 'access_token'](#)
 - [Credential Validation](#)
 - [Via postman](#)
 - [Via curl](#)
 - [NoneType INVALID_TYPE](#)
- [How to request a Salesforce Trial](#)
- [Documentation](#)

About the Salesforce Service Cloud solution

Salesforce Service Cloud is a customer relationship management (CRM) platform for customer service and support.

Account Setup

This article explains the steps needed to create a Salesforce Developer Account to enable API. The following parameters are needed to configure the Salesforce Service Cloud data connector:

- SalesforceUser
- SalesforcePass
- SalesforceConsumerKey
- SalesforceConsumerSecret
- SalesforceTokenUri (<https://login.salesforce.com/services/oauth2/token> )
- SalesforceSecurityToken

This article provides the steps to collect each of the parameters listed above.

Setup your Salesforce developer Account

1. Go to the signup page. <https://developer.salesforce.com/signup> ☑. <- User your PERSONAL email address.
2. Fill in all the details. Take note of the username (SalesforceUser). It will be in the form [\(name@yourcompany.com\)](mailto:(name@yourcompany.com))
3. Check your email to verify the account. In the email, there'll be a unique URL for login.

Click below to verify your account.

Verify Account

To easily log in later, save this URL:

<https://knowledgemd-dev-ed.my.salesforce.com>

Username:

james@knowledgemd.com

Again, welcome to Salesforce!

4. Complete the sign-up by providing the password. Take note of the password as it will be your "SalesforcePass".

Confirming "API Enabled" permissions.

1. Go to Setup. Enter Users in the Quick Find box. Select the Users tab.
2. You'll see a list of all the users. Check your Profile. Click on it.

The screenshot shows the Salesforce Setup page with the 'Users' tab selected. The left sidebar contains navigation options like 'Service Setup Assistant', 'Multi-Factor Authentication Assistant', 'Release Updates', 'Lightning Experience Transition Assistant', 'New Salesforce Mobile App QuickStart', 'Lightning Usage', 'Optimizer', and 'ADMINISTRATION'. Under 'ADMINISTRATION', 'Users' is selected. The main content area shows 'All Users' with a message: 'On this page you can create, view, and manage users. In addition, download SalesforceA to view and edit user details, reset passwords, and perform other administrative tasks from your mobile devices: iOS | Android'. Below this is a table of users.

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit	Chatter Expert	Chatter	chatty.00d5i000007dmwkea0.8a0yqbf6imgtx@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit	Miller_James	JMill	james@knowledgemd.com		✓	System Administrator
<input type="checkbox"/> Edit	User_Integration	integ	integration@00d5i000007dmwkea0.com		✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	insightssecurity@00d5i000007dmwkea0.com		✓	Analytics Cloud Security User

3. On the profile page, under "Administrative Permissions", the API Enabled check box should be marked.

SETUP Profiles	
Administrative Permissions	
Access Conversation Entries	✓
Access Experience Management	<input type="checkbox"/>
Access Libraries	✓
Add People to Direct Messages	✓
Allow Inclusion of Code Snippets from UI	<input type="checkbox"/>
Allow user to modify Private Connections	<input type="checkbox"/>
Apex REST Services	✓
API Enabled	✓
Assign Permission Sets	✓
Author Apex	✓
Bulk API Hard Delete	<input type="checkbox"/>
Can Approve Feed Post and Comment	<input type="checkbox"/>
Change Dashboard Colors	✓
Chatter Internal User	✓
Close Conversation Threads	<input type="checkbox"/>
Configure Custom Recommendations	<input type="checkbox"/>
Create and Customize Dashboards	✓
Create and Customize List Views	✓
Create and Customize Reports	✓
Create and Own New Chatter Groups	✓
Manage Next Best Action Strategies	✓
Manage Package Licenses	✓
Manage Password Policies	✓
Manage Profiles and Permission Sets	✓
Manage Promoted Search Terms	✓
Manage Prompts	<input type="checkbox"/>
Manage Public Classic Email Templates	✓
Manage Public Documents	✓
Manage Public List Views	✓
Manage Release Updates	✓
Manage Reporting Snapshots	✓
Manage Reports in Public Folders	✓
Manage Roles	✓
Manage Salesforce CRM Content	✓
Manage Salesforce Knowledge	✓
Manage Security Center	<input type="checkbox"/>
Manage Session Permission Set Activations	<input type="checkbox"/>
Manage Sharing	✓
Manage Synonyms	✓
Manage Unlisted Groups	✓

Creating a Connected App.

- Log in to Salesforce Cloud
- In the drop-down list of the account (in the upper-right corner), select **Setup**.
- In the left-hand pane, go to **App Setup > Create > Apps**.
- In the **Connected Apps** pane, click the New button.
- On the **New Connected App** page, fill in the following required fields under Basic Information:
 - Connected App Name. For example, Sentinel Integration.
 - API name. For example, UCMDB Integration.
 - Contact Email.
- Go to **API (Enable OAuth Settings)** and select **Enable OAuth Settings**.
- In the **Callback URL** field, enter <https://login.salesforce.com/> ☐ or <https://localhost> ☐ If you don't have a callback URL configured.
- In the **Selected OAuth Scopes** field, select **Access and manage your data (api)**, and then click **Add**.
- Click the **Save** button to save the new Connected App.
- In the **Connected Apps** list, find the App that you just created, and then click **Manage**.
- On the page that opens, click the **Edit** button.
- Under **OAuth policies**, select **All users may self-authorize** in the **Permitted Users** list, and then click the **Save** button.

Gettin a Consumer Key and a Consumer Secret.

1. From **Setup**, enter **App Manager** in the Quick Find box, and then select **App Manager**.
2. Click the dropdown menu for the connected app that you created and select **View**.
3. If the values for the Consumer Key and Consumer Secret fields are hidden, **Click to reveal**.
4. Copy the Consumer Key (**SalesforceConsumerKey**) and Consumer Secret (**SalesforceConsumerSecret**) values and save them for later use in this quick start.

Generate a Security Token

1. Log in to your Salesforce account.
2. Click the profile avatar and choose Settings.
3. Select My Personal Information → Reset My Security Token.
4. Check your email for the security token (**SalesforceSecurityToken**).

Additional Note

If you don't want to use security token in API request, **change "IP Restrictions" to "Relax IP restrictions"** setting for your Connected App. Although this is not recommended.

Salesforce common Errors:

Token getting failed. Exiting program. 'access_token'

This specific error can be identified from the Monitor section of the Azure Function as described in [Monitor Azure Functions](#)

Invocation Details		
Run query in Application Insights		
Timestamp	Message	Type
2021-06-22 13:59:59.997	Executing 'Functions.SalesforceSentinelConnector' (Reason='Timer fired at 2021-06-22T13:59:59.9953185+00:00', Id=11b7f057-0c3a-4f25-bff1-f6c6bb5c4be7)	Information
2021-06-22 14:00:00.022	Script started	Information
2021-06-22 14:00:00.173	Token getting failed. Exiting program. 'access_token'	Error
2021-06-22 14:29:59.990	Timeout value of 00:30:00 exceeded by function 'Functions.SalesforceSentinelConnector' (Id: '11b7f057-0c3a-4f25-bff1-f6c6bb5c4be7'). Initiating cancellation.	Error
2021-06-22 14:30:00.010	Executed '{functionName}' ({status}, Id={invocationId}, Duration={executionDuration}ms)	Error
2021-06-22 14:30:00.010	Executed 'Functions.SalesforceSentinelConnector' (Failed, Id=11b7f057-0c3a-4f25-bff1-f6c6bb5c4be7, Duration=1800004ms)	Error
2021-06-22 14:30:00.012		Error

We can see that is failing at the time to authenticate

```
def _get_token():
    params = {
        "grant_type": "password",
        "client_id": consumer_key,
        "client_secret": consumer_secret,
        "username": user,
        "password": f'{password}{security_token}'
    }
    try:
        r = requests.post(url, params=params)
        _token = json.loads(r.text)['access_token']
        _instance_url = json.loads(r.text)['instance_url']
        return _token, _instance_url
    except Exception as err:
        logging.error(f'Token getting failed. Exiting program. {err}')
        exit()
```

To troubleshoot this issue you need to make sure that the requested values (**Salesforce API Username, Salesforce API Password, Salesforce Security Token, Salesforce Consumer Key, Salesforce Consumer**

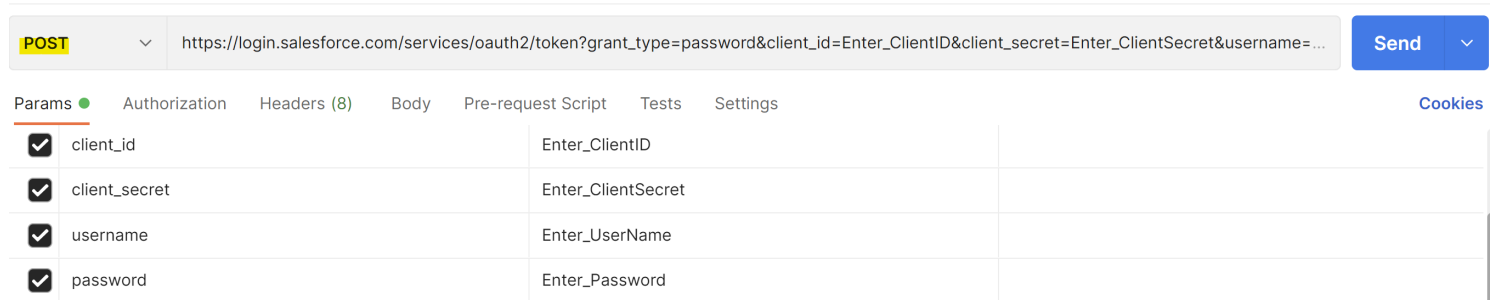
Secret) on the Azure Function configuration are set correctly.

Credential Validation

Via postman

You can validate the credentials using [POSTMAN](#)  performing a POST request to

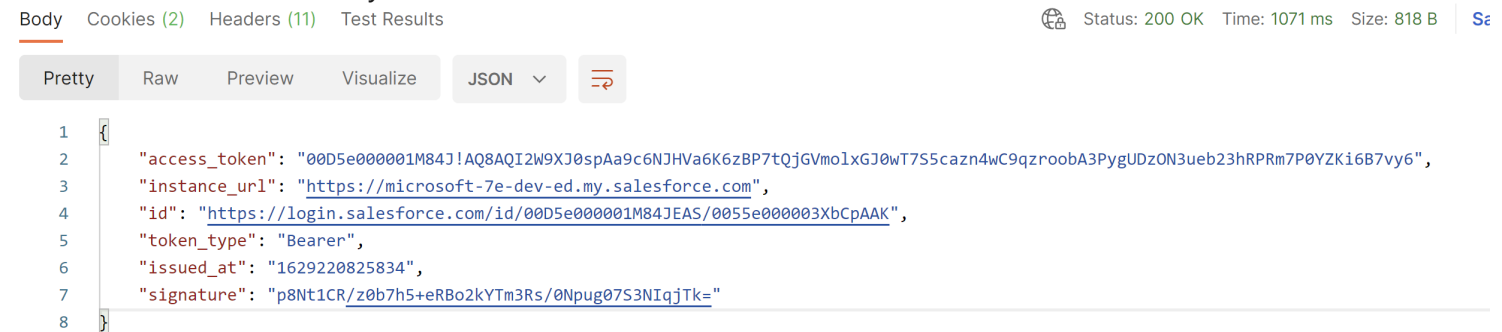
```
https://login.salesforce.com/services/oauth2/token?grant_type=password&client_id=Enter_ClientID&client_secret=Enter_ClientSecret&username=Enter_UserName&password=
<SalesforcePassword concatenated with the security token>
```



The image shows the Postman interface for a POST request. The URL bar contains the endpoint: `https://login.salesforce.com/services/oauth2/token?grant_type=password&client_id=Enter_ClientID&client_secret=Enter_ClientSecret&username=...`. The 'Params' tab is active, displaying a table of query parameters:

Key	Value
client_id	Enter_ClientID
client_secret	Enter_ClientSecret
username	Enter_UserName
password	Enter_Password

If authentication is success you would see a 200 status.



The image shows the Postman interface displaying the response body. The status is 200 OK, and the response is in JSON format. The response body is as follows:

```
{
  "access_token": "00D5e000001M84J!AQ8AQI2W9XJ0spAa9c6NJHVa6K6zBP7tQjGVmolxGJ0wT7S5cazn4wC9qzroobA3PygUDzON3ueb23hRPRm7P0YZKi6B7vy6",
  "instance_url": "https://microsoft-7e-dev-ed.my.salesforce.com",
  "id": "https://login.salesforce.com/id/00D5e000001M84JEAS/0055e000003XbCPAAK",
  "token_type": "Bearer",
  "issued_at": "1629220825834",
  "signature": "p8Nt1CR/z0b7h5+eRBo2kYTm3Rs/0Npug07S3NIqjTk="
}
```

Via curl

You can execute a curl command in azure cloud shell to test the salesforce credentials.

```
curl -i -X POST https://microsoftcss-dev-ed.my.salesforce.com/services/oauth2/token -d "grant_type=password" -
```

For example:

```
benjamin@Azure:~$ curl -i -X POST https://microsoftcss-dev-ed.my.salesforce.com/services/oauth2/token -d "grant_type=password" -d "client_id=3M9A...  
...  
-d "username=l...  
...  
-d "password=...  
...  
HTTP/1.1 200 OK  
Date: Tue, 28 Feb 2023 14:37:12 GMT  
Set-Cookie: CookieConsentPolicy=0;1; path=/; expires=Wed, 28-Feb-2024 14:37:12 GMT; Max-Age=31536000  
Set-Cookie: LSKey=c$CookieConsentPolicy=0;1; path=/; expires=Wed, 28-Feb-2024 14:37:12 GMT; Max-Age=31536000  
Strict-Transport-Security: max-age=63072000; includeSubDomains  
X-Content-Type-Options: nosniff  
X-XSS-Protection: 1; mode=block  
X-Robots-Tag: none  
Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private  
Set-Cookie: BrowserId=Y0cxdbd1Ee21eY2fnpSBA; domain=.salesforce.com; path=/; expires=Wed, 28-Feb-2024 14:37:12 GMT; Max-Age=31536000  
Expires: Thu, 01 Jan 1970 00:00:00 GMT  
X-ReadOnlyMode: false  
Content-Type: application/json;charset=UTF-8  
Vary: Accept-Encoding  
Transfer-Encoding: chunked
```

NoneType INVALID_TYPE

This specific error can be identified from the Monitor section of the Azure Function as described in [Monitor Azure Functions](#)

We can see that is failing at the time to get the Salesforce log information

```
def pull_log_files():  
    past_time = generate_date()  
  
    if interval == 'hourly':  
        query = "/services/data/v44.0/query?q=SELECT+Id+,+EventType+,+Interval+,+LogDate+,+LogFile+,+LogFileLe  
        "+FROM+EventLogFile" + \  
        f"+WHERE+Interval+=+'Hourly'+and+CreatedDate+>+{past_time}"
```

◀ ▶

Invocation Details		
Run query in Application Insights		
Timestamp	Message	Type
2021-08-05 16:00:00.008	Executing 'Functions.SalesforceSentinelConnector' (Reason=Timer fired at 2021-08-05T16:00:00.0025170+00:00; Id=457bf3be-3355-446b-bd77-c5854f630b6c)	Information
2021-08-05 16:00:00.031	Script started	Information
2021-08-05 16:00:01.300	Searching files last modified from 2021-08-05T15:00:00Z	Information
2021-08-05 16:00:01.400	File list getting failed. Exiting program. 400 [{"message": "\n LogFile , LogFileLength FROM EventLogFile WHERE Interval = 'Hourly'\n ^\nERROR at Row:1:Column:75]rsObject type 'EventLogFile' is not supported. If you are attempting to use a custom object, be sure to append the '_'c' after the entity name. Please reference your WSDL or the describe call for the appropriate names.", "errorCode": "INVALID_TYPE"]]	Error
2021-08-05 16:00:01.421	Result: Failure Exception: TypeError: 'NoneType' object is not iterable Stack: File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/dispatcher.py", line 401, in _handle__invocation_request invocation_id, fi_context, fi_func, args) File "/usr/local/lib/python3.6/concurrent/futures/thread.py", line 56, in run result = self.fn(*self.args, **self.kwargs) File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/dispatcher.py", line 603, in _run_sync_func func(params) File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/extension.py", line 215, in _raw_invocation_wrapper result = function(**args) File "/home/site/wwwroot/SalesforceSentinelConnector/_init_.py", line 215, in main for line in pull_log_files():	Error
2021-08-05 16:00:01.421	Executed 'Functions.SalesforceSentinelConnector' (Failed, Id=457bf3be-3355-446b-bd77-c5854f630b6c, Duration=1410ms)	Error
2021-08-05 16:00:01.425	Result: Failure Exception: TypeError: 'NoneType' object is not iterable Stack: File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/dispatcher.py", line 401, in _handle__invocation_request invocation_id, fi_context, fi_func, args) File "/usr/local/lib/python3.6/concurrent/futures/thread.py", line 56, in run result = self.fn(*self.args, **self.kwargs) File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/dispatcher.py", line 603, in _run_sync_func func(params) File "/azure-functions-host/workers/python/3.6/LINUX/X64/azure_functions_worker/extension.py", line 215, in _raw_invocation_wrapper result = function(**args) File "/home/site/wwwroot/SalesforceSentinelConnector/_init_.py", line 215, in main for line in pull_log_files():	Error

Customer need to validate if he's able to get the log information using [WORKBENCH](#)

```
Performing a GET request /services/data/v44.0/query?  
q=SELECT+Id+,+EventType+,+Interval+,+LogDate+,+LogFile+,+LogFileLength+FROM+EventLogFile
```

Try the [Salesforce APIs for Postman](#).

Choose an HTTP method to perform on the REST API service URI below:

☒ GET ☐ POST ☐ PUT ☐ PATCH ☐ DELETE ☐ HEAD

Headers

Reset

Up


/services/data/v44.0/query?q=SELECT+Id+,+EventType+,+]

Execute

[Expand All](#) | [Collapse All](#) | [Show Raw Response](#)

```
{
  "totalSize": 6,
  "done": true,
  "records": [
    { "Id": "0AT5e00001KNLbsGAH" },
    { "Id": "0AT5e00001JZyaDGAT" },
    { "Id": "0AT5e00001KNLbtGAH" },
    { "Id": "0AT5e00001JZyaJGAT" },
    { "Id": "0AT5e00001KNLbuGAH" },
    { "Id": "0AT5e00001JZyaKGAT" }
  ]
}
```

How to request a Salesforce Trial

You can request a developer edition following this steps: https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/quickstart_dev_org.htm 

** You need to use your personal email address.

Documentation

Salesforce data connector Public documentation:

<https://docs.microsoft.com/en-us/azure/sentinel/connect-salesforce-service-cloud> 

Troubleshoot REST API connectors:

[https://dev.azure.com/SupportabilityWork/Azure Security/_wiki/wikis/Azure Sentinel CSS wiki/3452/REST-API-Connectors-\(Poll-Azure-Functions\)](https://dev.azure.com/SupportabilityWork/Azure Security/_wiki/wikis/Azure Sentinel CSS wiki/3452/REST-API-Connectors-(Poll-Azure-Functions))

[https://dev.azure.com/SupportabilityWork/Azure Security/_wiki/wikis/Azure Sentinel CSS wiki/3659/Sentinel-Function-Type-Data-Connectors-information-\(Beginning\)](https://dev.azure.com/SupportabilityWork/Azure Security/_wiki/wikis/Azure Sentinel CSS wiki/3659/Sentinel-Function-Type-Data-Connectors-information-(Beginning))