

How to obtain an OAuth 2.0 token in SAP Cloud Platform

Author : Tobias Hofmann

Date : May 14, 2020

[SAP Help OAuth 2.0 Service](#)

[Sample request in my Gitlab repo](#)

For the example, I am going to use an OAuth authorization grant of type client credentials. SAP Cloud Platform environment: Neo.

Create scope

Select an OAuth protected Java app and add a scope to it. If you just want to test the token flow, the app does not need to implement OAuth. You can assign an OAuth scope to any app. The app does not even be up and running.

Select a Java app and go to OAuth Scopes.

Add a new scope

ID: read Description: test read scope

This adds a new scope to your Java app.

Create client

In the SCP admin dashboard, go to OAuth and tab clients

It's full of stars!

Where documentation meets reality

Register a new client

The form is activated and some values are already filled out. Some values must be changed when adding an OAuth client with client credentials.

Name: OAuthTest Description: Test OAuth client Subscription: select the Java app where you added the scope to Id: use the proposed value Authorization grant: Client Credentials Confidential: Yes (after selecting authorization grant, the value should be set and read only) Secret: provide a strong password Token Lifetime: 60 minutes

Obtain OAuth token

A token can be obtained by sending a request to the token endpoint URL. This URL can be found in the OAuth security section in the branding tab.

Here, the URI is <https://oauthservices-p539123trial.hanatrial.ondemand.com/oauth2/api/v1/token>

To get the token, a GET request containing the following information is send. The authorization is Basic, with the client_id and password

```
Client id: 5dc904c5-9130-36a9-b7e7-0f25a2e3c102 Password: abc123
```

The String 5dc904c5-9130-36a9-b7e7-0f25a2e3c102:abc123 must be Base64 encoded.

Header

```
Authorization:Basic NWRjOTA0YzUtOTEzMC0zNmE5LWI3ZTctMGYyNWEyZTNjMTAyOmFiYzEyMw==
```

Request Body

```
grant_type:"client_credentials" scope:"read"
```

Response

As a response, the SCP OAuth token service will send back an access token.

```
access_token: "6667a2b0ba812fce9283650b69141d" token_type: "Bearer" expires_in: 3600 scope: "read"
```

Example Postman

[Sample request in my Gitlab repo](#)

To get an OAuth token from SCP using Postman, create a new request and open the Authorization tab. Select as type OAuth 2.0.

Click on Get new access token.

This will open a form that helps you to insert the information needed to obtain a token.

Token Name: scpoauthtoken Grant Type: Client Credentials Access Token URL: See
SCP Client ID: See SCP OAuth client configuration Client Secret: See SCP OAuth
client configuration Scope: See SCP Java app

Request

Response

Token is issued and can be used by Postman

As you can see, the token is not an OpenID Connect token. Therefore, this token can only be used by SCP apps. You cannot use this token to forward it to an OIDC enabled app.

Troubleshooting

Error while trying to receive token.

Error: Cound not complete OAuth 2.0 token request: "Scopes exceed the scope registered for the client."

Solution A

Check that the scope assigned to the app in SCP and send by the client are the same.

Client Scope:

App Scope:

Solution B

In case you just requested a token: wait for a few minutes. Sometimes a new token is issued after waiting for a few minutes. This should not happen, but it is a workaround.