aruba
a Hewlett Packard Enterprise company

Central OAuth Token Generation

API Guide

Copyright

Open Source Code

Certain Aruba products include Open Source software code developed by third parties, including software code subject to the GNU General Public License (GPL), GNU Lesser General Public License (LGPL), or other Open Source Licenses.

Contents

Introduction

The API Gateway feature in Central supports the REST API for all Central services. This feature allows Central users to write custom applications and embed or integrate the APIs with their own applications. The Central API Framework plug-in supports the OAuth protocol for authentication and authorization, to provide secure access to the APIs. The access tokens created by this OAuth token generation procedure provide temporary and secure access to the APIs. These access tokens have a limited lifetime for enhanced security, and the applications should use the refresh API to obtain new tokens periodically (every 2 hours). This document also describes the procedures that enable applications to automatically get new tokens, or refresh an old token.

OAuth Token Generation for the Central App

Step 1: Authenticate a User and Create a User Session

The following API authenticates a user and returns a user session value that can be used to create future requests for a client with the specified username and password. It is assumed that you already have a client ID for your application. For more information on how to create an application and obtain tokens, see Creating an Application. The domain URLs in Table 1 allow you to log in to the API gateway server and to establish the user session. This endpoint is accessible over SSL, and HTTP (non-SSL) connections are redirected to SSL port. The following table lists the region specific domain URLs for accessing the API gateway.

Table 1: Domain URLs for API Gateway Access

Region	Domain Name
US-1	app1-apigw.central.arubanetworks.com
US-2	apigw-prod2.central.arubanetworks.com
EU-1	eu-apigw.central.arubanetworks.com
Canada-1	apigw-ca.central.arubanetworks.com
China-1	apigw.central.arubanetworks.com.cn
APAC-1	api-ap.central.arubanetworks.com
APAC-EAST1	apigw-apaceast.central.arubanetworks.com
APAC-SOUTH1	apigw-apacsouth.central.arubanetworks.com

The procedures described in this article use app1-apigw.central.arubanetworks.com/oauth2/authorize/central/api/login as an example.
Ensure that you use the appropriate domain URL when accessing API Gateway or generating tokens.

If user authentication is successful, the request will return HTTP code 200 and the response header will include the following attributes.

Table 2: Authentication and User session Response Codes

Header Key	Values	Description
Set Cookie	csrftoken=xxxx; session=xxxx	The server returns a CSRF token and identifies the user session, which must be used for all subsequent HTTP requests.

Example:

Request Method: POST

URL: https://app1-apigw.central.arubanetworks.com/oauth2/authorize/central/api/login?client_id=<client_id> HTTP/1.1
Host: app1-apigw.central.arubanetworks.com
Accept: application/json
Content-Length: ??

Content-Type: application/json

POST Request Body (JSON):

```
{
  "username": "xxxxx",
  "password": "xxxxx"
}
```

Error Response:

400: Bad Request

- Response Body (JSON):

```
{
  "extra": {},
  "message": "<error string>"
}
```

401: Auth failure

- Response Body (JSON):

```
{
  "message": "Auth failure",
  "status": false
}
```

Success Response:

200: OK

- Response Body (JSON):

```
{
 "status": true
}
```

- Response Header: Set-Cookie:

```
Set-Cookie: csrftoken=xxxx;session=xxxx;
```

The **csrf token** value received in the successful response message must be used as a parameter for all subsequent POST/PUT requests. The **session** value must also be used for all subsequent requests to maintain the user session context.

Step 2: [Optional] Generate Client Credentials

The following API can be used to generate client credentials for a specific tenant using your Managed Service Provider (MSP) Client ID.

Table 3: URL to Generate Client Credentials

URL	Description
https://app1-apigw.central.arubanetworks.com/oauth2/authorize/central/api/client_credentials?client_id=<msp_client_id>	The **<msp_client_id>** variable is the client ID given from Central to that a Managed Service Provider that user registered the application.

Example:

Request Method: POST
URL:
https://app1apigw.central.arubanetworks.com/oauth2/authorize/central/api/client_credentials
? client_id=<msp_client_id>

POST Request Body (JSON):

```
{
  "customer_id": "<tenant_id>"
}
```

Request Header: (Values from login API request)

```
Set-Cookie: csrftoken=xxxx;session=xxxx;
```

Response Body (JSON):

```
{
"client_id": "<new-client-id>",
"client_secret": <new-client-secret>"
}
```

Step 3: Generate Authorization Code

After the user is authenticated and you have a valid session for that user, use this API to get authorization code. The authorization code is valid only for 5 minutes and must be exchanged for a token within that time.

Table 4: URL for to Generate an Authorization Code

URL	Description
https://app1-apigw.central.arubanetworks.com/oauth2/authorize/central/api	The endpoint is a POST call to get an authorization code.

Query parameters for this API are as follows:

Table 5: Query Parameters for the Auth Code API

Parameter	Values	Description
client_id	**client_id** is a unique hexadecimal string	The **client_id** is a unique identifier that identifies the caller. Application developers obtain a client ID and a client secret when they register with the API gateway admin.
response_type	**code**	Use **code** as the response type to get the authorization code that can be exchanged for token
scope	**all** or **read**	Requested API permissions may be either **all** (for both read and write access) or **read** for read-only access.

Example:

Request Method: **POST**

URL: https://app1 - apigw.central.arubanetworks.com/oauth2/authorize/central/api/?client_id=<client_id>&response_type=code&scope=all HTTP/1.1
Host: app1-apigw.central.arubanetworks.com
Accept: application/json Cookie: “session=xxxx” X-CSRF-Token: xxxx Content-Length: ??
Content-Type: application/json

POST Request Body (JSON):

```
{
  "customer_id": "xxxxx"
}
```

Error Response:

400: Bad Request

- Response Body (JSON):

```
{
  "extra": {},
  "message": "<error string>"
}
```

Success Response:

200: OK

- Response Body (JSON):

```
{
" auth_code ": “xxxx”
}
 session=xxxx;
```

Pass the **csrf-token** value you obtained in step one in the request header, otherwise the request will be rejected. Note the **auth_code** value in the response, as you will use this code to obtain an OAuth token.

Step 4: Exchange Auth Code for a Token

Once you have an authorization code, you just use that code to request an access from the server. The exchanges should be done within 300 seconds of obtaining the auth code from the previous step, or the API will return an error.

Table 6: URL to Generate an Auth Token

URL	Description
https:// app1-apigw.central.arubanetworks.com/oauth2/token	The endpoint is a POST call to get an access token using the authorization code obtained from the server.

Query parameters for this API are as follows:

Table 7: Query Parameters for the Auth Code API

Parameter	Values	Description
client_id	**client_id** is a unique hexadecimal string	The **client_id** is a unique identifier that identifies the caller. Application developers can obtain a client ID and a client secret when they register with the API gateway admin.
client_secret	**client_secret** is a unique hexadecimal string	The **client_secret** is a unique identifier provided to each developer at the time of registration. Application developers can obtain a client ID and client secret when they register with the API gateway admin.
grant_type	**authorization_code**	Use **code** to get the authorization code that can be exchanged for the token.
code	**auth_code** received from step 1	The authorization code received from the authorization server.
redirect_uri	String	The redirect URI must be the same as the one given at the time of registration. This is an optional parameter.

The response to this API query is a JSON dictionary with following values:

Table 8: Auth Token Values

Parameter	Values	Description
token_type	bearer	Identifies the token type. Central supports only the **bearer** token type (See https://tools.ietf.org/html/rfc6750)

Parameter	Values	Description
refresh_token	string	Refresh tokens are credentials used to renew or refresh the access_token when it expires without repeating the complete authentication flow. A refresh token is a string representing the authorization granted to the client by the resource owner.
expires_in	seconds	The lifetime, in seconds, of the access token.
access_token	string	Access tokens are credentials used to access protected resources. An access token is a string representing an authorization issued to the client.

Example:

Method: POST

https: //apigw-prod2.central.arubanetworks.com/oauth2/token?client_id=<Ccentral-API-app-client-id>&client_secret=xxxx&grant_type=authorization_code&code=xxxx \
Content-Type: application/json

Response:

```
{
"refresh_token": "xxxx",
"token_type": "bearer",
"access_token": "xxxx",
"expires_in": 7200
}
```

Step 5: Refreshing a Token

You can use the refresh token obtained in the previous step to update the access token without repeating the entire authentication process.

Table 9: URL to Generate an Auth Token

URL	Description
https:// app1-apigw.central.arubanetworks.com/oauth2/token	The endpoint is a POST call to refresh the access token using the refresh token obtained from the server

Query parameters for this API are as follows:

Table 10: Query Parameters for the Auth Token API

Parameter	Values	Description
client_id	**client_id** is a unique hexadecimal string	The **client_id** is a unique identifier that identifies the caller. Application developers obtain a client ID and a client secret when they register with the API gateway admin.

Parameter	Values	Description
client_secret	**client_secret** is a unique hexadecimal string	The **client_secret** is a unique identifier provided to each developer at the time of registration. Application developers obtain a client ID and a client secret when they register with the API gateway admin.
grant_type	**refresh_token**	Specify **refresh_token** as the grant type to request that an authorization code be exchanged for a token.
refresh_token	string	The string must be the refresh token received in Step 4. A refresh token is a string representing the authorization granted to the client by the resource owner.

The response to this API query is a JSON dictionary with following values:

Table 11: Auth Token Values

Parameter	Values	Description
token_type	bearer	This value identifies the token type. Central supports only the **bearer** token type (See https://tools.ietf.org/html/rfc6750)
refresh_token	string	Refresh tokens are credentials used to renew or refresh the access token when it expires without repeating the complete authentication flow. A refresh token is a string representing the authorization granted to the client by the resource owner.
expires_in	seconds	The lifetime, in seconds, of the access token.
access_token	string	Access tokens are credentials used to access protected resources. An access token is a string representing an authorization issued to the client.

Example:

Method: POST

https: //apigw-prod2.central.arubanetworks.com/oauth2/token?client_id=<Ccentral-API-app-client-id>&client_secret=xxxx&grant_type=authorization_code&code=xxxx \
Content-Type: application/json

Response:

```
{
"refresh_token": "xxxx",
"token_type": "bearer",
"access_token": "xxxx",
"expires_in": 7200
}
```