Changes while working with Cisco API's

Note this document is for those who are already using Cisco API's and have created applications on Cisco API console prior to Mar 1, 2023. If you have created applications prior to Mar 1, 2023, your client id and secret will expire by Sep 30, 2023. In order to continue using Cisco API's you will have make certain changes to your existing code.

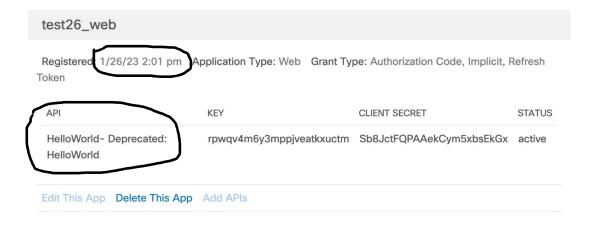
These changes are described primarily in 3 steps.

- a. Register new application for using Cisco API's at https://apiconsole.cisco.com
- b. Use new token URL along with new application details from step 1 to generate token.
- c. Use newly generated token to make API call using modified domain name in the API endpoint URL.

a) Register new Application.

- 1. Login to developer portal at https://apiconsole.cisco.com
- Click on "My apps and keys" and you will see all your existing applications with the API lists for the applications. You will see deprecated word in the API name for your existing applications. Make a note of all API's listed here. You will need this list while registering new applications.

Example:



3.	You can right click on "Register a New App" and select "open link in new tab". This way you can have list of existing applications open in one tab and "register new app" on another tab.

4.	"Register a New App'	' page has cha	nged. Primaril	y related to "	'Application ⁻	Гуре" ar	ıd
	"Grant Type". Your se	lection of "App	olication Type	' will change	the "Grant T	ype" off	fered.

Application Type				
Your choice of "Application Type" will determine the "Grant Type" available:				
⊕ Browser				
○ Web				
○ Native				
○ Service				
Grant Type				
Choose at least one Grant Type:				
Resource Owner Password Credentials				
☐ Authorization Code [with PKCE only for Browser and Native app type]				
☐ Client Credentials				
□ Implicit				
☐ Refresh Token				
Redirect URI (You can enter multiple URI with comma separated.)				
You can enter multiple redirect URIs.				

5. Take a note of "Grant Type" listed on the existing application on the "My Apps and Keys" page under "Applications" section. You may be required to register multiple applications if you are using same application for different grant types today.

Follow following table for selecting right "Application Type" to get exact "Grant Type" in your existing application.

Application Type Valid Grant Type

browser	authorization_code , implicit
native	authorization_code , implicit , password , refresh_token
service	client_credentials
web	authorization_code , implicit , refresh_token
4	

- 6. Once you have selected the right "Application type" and "Grant type", you can take a note of "API's" listed on existing application. Select the same list of "API's" while registering new application. Here you will select same API's without the deprecated word in it. Note that you won't see deprecated API's list while registering.
- 7. Agree to terms of service and click on "Register".
- 8. You will get the new application details in email as well as you can view it by clicking "My apps and keys". Note down client id and secret for the application which you will use to generate the token.

Example:

This example shows Application creation for HelloWorld API with Application Type as "Service" and Grant Type = "Client Credentials"

Register an Application

Get a key and register your application using the form below to start working with our APIs.

Application Details

Name of your application (you can change it later) *

Client credential Flow example Hello World

Please describe what your application will do

this application will be used for client credentials flow

Application Type : Service Grant Type: Client Credentials

Application Type

Your choice of "Application Type" will determine the "Grant Type" available:

Browser

○ Web

Native

Service

Grant Type

Choose at least one Grant Type:

Resource Owner Password Credentials

Authorization Code [with PKCE only for Browser and Native app type]

✓ Client Credentials

Implicit

Refresh Token

Redirect URI (You can enter multiple URI with comma separated.)

You can enter multiple redirect URIs.



HelloWorld

RATE LIMITS

100 Calls per second

100,000 Calls per day

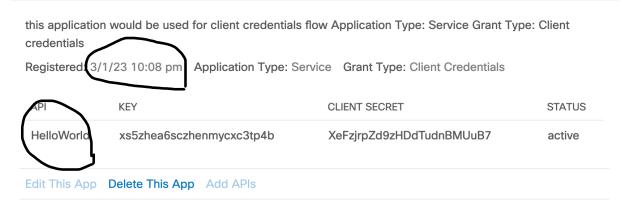
Terms of Service

Please review the services you have selected above and agree to the terms of service $\ensuremath{\mathbb{Z}}$



✓ I agree to the terms of service

Client credentials example Flow Hello World



b) Use new token URL along with new application details from step 1 to generate token

Your code is using the following end point to generate a token on the platform.

https://cloudsso.cisco.com/as/token.oauth2

Going forward you will use the following token generation endpoint.

https://id.cisco.com/oauth2/default/v1/token

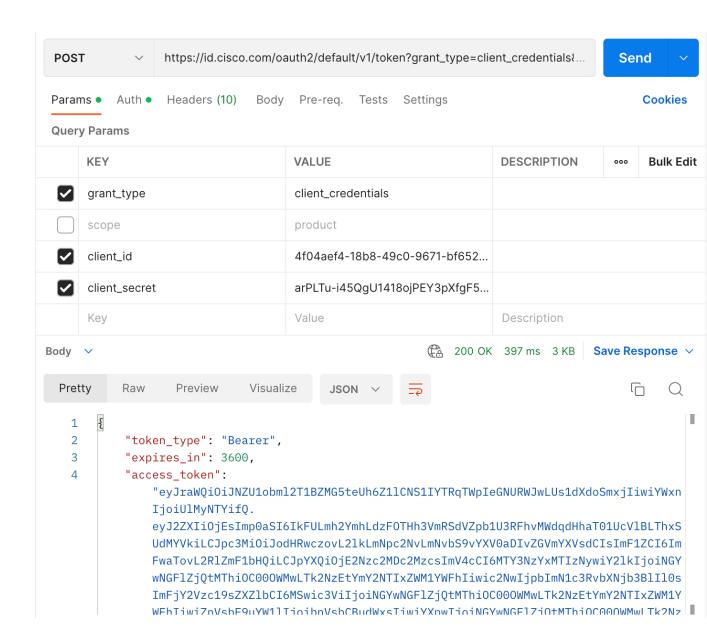
- 1. Old token generation URL: https://cloudsso.cisco.com/as/token.oauth2
- 2. You can use your newly created "Client id" and "Secret" with the new token generation URL for generating a token.

Example for "Application Type" = "Service" and "Grant Type" = "Client Credentials"

Important: you need to pass additional header with the token generation call without it you won't get the token.

Content-Type: application/x-www-form-urlencoded

3. Token generation call using Postman:



Token generation using curl # curl -X POST

'https://id.cisco.com/oauth2/default/v1/token?grant_type=client_credentials&client_id =pahfekfcjhk8w2hwpjffydgt&client_secret=Qv59wJ6nDAK7mQTPrJxCZ5zZ' --header 'Content-Type: application/x-www-form-urlencoded' --header 'Content-Length: 0'

5. Now it's time to make an API call since you have generated the "token" with the newly created application details.

Note that your "API endpoint URL" is changed for compared to your earlier endpoint URL.

Example: Take a note of domain name change from api.cisco.com to apix.cisco.com.

Please note that your API endpoint URL remains same except domain name.

Old API endpoint URL: https://api.cisco.com/hello

Old domain name: api.cisco.com

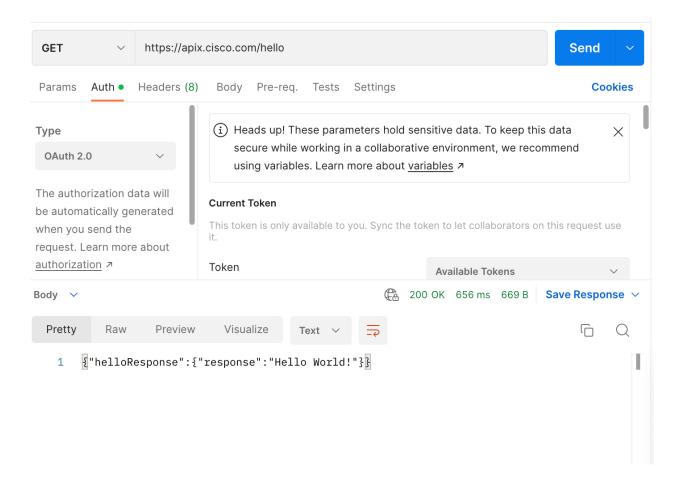
New API endpoint URL: https://apix.cisco.com/hello

New domain name: apix.cisco.com

Test your API call with the newly created token. You are ready to use this in your code / application.

Example of API call with client credentials.

Postman:



Curl:

curl --location 'https://apix.cisco.com/hello' --header 'Authorization: Bearer

eyJraWQiOiJNZU1obml2T1BZMG5teUh6Z1lCNS1lYTRqTWpleGNUR WJwLUs1dXdoSmxjliwiYWxnljoiUlMyNTYifQ.eyJ2ZXliOjEsImp0aSl6lk FULm5fUWlieVFRNEJ6cnJOOVAyUmtUaE9uaERBdEpsbE1HQTFDcj VvS0t0Qm8iLCJpc3MiOiJodHRwczovL2lkLmNpc2NvLmNvbS9vYXV0 aDlvZGVmYXVsdClsImF1ZCl6lmFwaTovL2RlZmF1bHQiLCJpYXQiOj E2Nzc2OTg4MjYsImV4cCl6MTY3NzcwMjQyNiwiY2lkljoibnlnMzh4ZW 05Zmoydjg3YndhOTk1ODZ1liwic2NwljpbImN1c3RvbXNjb3Blll0sImFj Y2Vzc19sZXZlbCl6MSwic3ViljoibnlnMzh4ZW05Zmoydjg3YndhOTk1O DZ1liwiZnVsbF9uYW1lljoibnVsbCBudWxsliwiYXpwljoibnlnMzh4ZW05 Zmoydjg3YndhOTk1ODZ1In0.H16tUd2KWyBYW_jmAU-b7gvhPe_Kt05zH_7GSpAAYB5u1bkeqjf4TlqKSP7EVr58wkF2K59xEm

2MBS4ZEDq9cecsBxdn406G9s1fJqK_95C_r6vlQHY5k-gx__lQnkoeD3gvhAgCl5eCuB4DPW-nv0xBshRNyGBk11ZDyx83RAv6Ai_3TkleeEalM5OTr7oMc8MxrPLmzl0c6txnqcX7QSHS5S3LRoQyFrCiQBPTsGcoL-vhL351XzJ0Dkfm2Hz94AVYym_alb69yE3QWE_gb0nuoA0GPx3GHi81xJN8bau-kwyVXQXzmjDew3BW4PSDCujwQRAgQX5-FYiKqtmlwA'