

Knowledgebase > Les outils NetExplorer > Platform's APIs access > Connecting your APIs with OAuth2

Connecting your APIs with OAuth2 Nicolas ARBOUIN - 2025-05-22 - Platform's APIs access

When connecting to the NetExplorer APIs, you now have to use OAuth2 authentication. This article shows you how to configure it.

The way in which your applications can connect to the NetExplorer APIs by retrieving a user's credentials is changing. Indeed, if you have subscribed to the possibility, your platform wil give you the way to create an application enabling OAuth2 connection.

On your plaform

As an administrator, when you connect on your browser, go to the menu \checkmark at the bottom left of the screen Once you are on this menu, choose the tab \checkmark at the end of the central page, then simply click on \checkmark

	Management Configuration Archives Applications		
88 Dashboard	Show NetExplorer's applications		Edit cors exceptions Create an app
		You don't have any applications at the moment.	
			T
🔟 Recycle bin			
Administration			

Creating your OAuth2 application

When clickiig on creation button, the window below will appear

	LOGO 1	
Application name	° 2	
Description	3	
Editor name	4	
Publisher's webs	site 5	
Callback URIs	6	
Select scopes	7	
Application type Client	8	
Client		
Cuent		
Server		S
Server		elieking og 🗙
Server 1. You can	assign a logo to your authentication application by	clicking on 본

- 5. The publisher's or creator's site
- 6. Any callback urls to be implemented during a client connection
- 7. By clicking on this button, you will select the rights scopes assigned to your application
- 8. The type of application you are going to use
 - Client: use an existing profile account on the platform (for example, to access items to which the user has rights)
 - Server: create an application allowing administrator access (for example, for global access scripts, etc.).

Selecting scopes

When clicking on Ebutton (point 7 in the above list) you'll get a window allowing you to

give the necessary rights for your application

Create an app	×
Select all	
account	~
read	
accountsec	~
read	
annotations	~
read	
archives	~
read	
alerts	~
read	
files	~
read	

Depending on your requirements, you can grant the various calls listed in the scopes (account, annotations, files...) the rights the applications must have, as soon as an API call is made:

5

- 1. Read
- 2. Write
- 3. All rights (read and write)

The \blacksquare button allow you to give global rights to all calls.

Once you have filled in the required fields, click save.

Your application has been created, a window showing 2 parameters required for your connection will appear:

TestSupport

warning: write down and save your client_secret now. It will no longer be accessible after this step. The client_id will remain available later. client_id

h5Cl	Ws7	ſ
client_secret		
×Y	Is4	Ū

- client_id: you'll be able to find this parameter within your application, if necessary.
- client_secret: remember to copy this parameter, as you will not be able to access it afterwards.

Connecting your application, example with Postman

Once the elements have been configured, you can call the authentication API within your script or code elements.

Here we're going to take a simple look at how to call your authentication via Postman. As shown in the API documentation, I'm going to go here:

https://api.netexplorer.fr/v3/#authorize the 3 necessary parameters:

- client_id
- grant_type
- client_secret

The purpose there is to get the connexion token of OAuth2 application, in server mode.

C Overview POST https://formation1.nete • +	
https://formation1.netexplorer.pro/oauth2/token	🖺 Save 🗸 Share
POST v https:/ netexplorer.pro/oauth2/token	Send ~
Params Authorization Headers (9) Body Scripts Settings	
⊖ none ⊖ form-data ⊖ x-www-form-urfenceded ● raw ⊖ binary ⊖ GraphQL JSON ∨	
	Beautify
1 [8] 2 "client_id": "h50 %s7", 3 "grant_type": "client_credentials",	Beautify
1 [8] 2 "client_id": "h5c Ws7", 3 "grant_type": "client_credentials", 4 "client_secret": "XY \$4"	Beautify

The answer is automatically generated in JSON, including the acces_token

Body	Cookies (1) Headers (37) Test Results 4	200 Ok 🔹 1.54 s 🕬		
{} JS	DN v D Preview 🖏 Visualize V		,≞ Q	i Gi é
	"access_token": "dIB8CD 7PT5",			
	"scope": "account.readwrite account.read accountsec.readwrite accountsec.read annotations.readwrite annotations.read alerts.readwrite alerts.read files.readwrite files.read groups.readwrite groups.read sharelinks.readwrite sharel uploadlinks.read users.readwrite users.read workflow.readwrite workflow.read trashes.readwrite trashes.read admi readwrite templates.read tags.readwrite tags.read",	archives.readwrit inks.read uploadli n.readwrite admin.	e archive nks.readw read temp	s.read rite lates.
	"refresh_token": null,			
	"token_type": "Bearer",			
	"expires_in": 86400			