# Getting started with our APIs via Postman

07/24/2024 2:55 pm EDT

Today we will begin connecting with DrChrono's APIs using Postman. Before we start let's make sure you download Postman here: https://www.postman.com/downloads/

## Authorization

Let's begin by opening Postman and creating a collection. Navigate to the left-hand side of the page and select "Create Collection".



Once the collection is created and named, you should see tabs from the overview screen. Select "Authorization", choose the "Type" dropdown box and select "OAUTH 2.0", then scroll down to the "Configure New Token" section.

	IO API	New Import	Overview DrChrono AP	I - User 1 • +		
Collections	+ =	000	DrChrono API - User 1		A S	
_	<ul> <li>DrChrono API - User 1</li> </ul>		Overview Authorization • Pre-request S	Script Tests Variables Runs		
invironments	This collection is empty Add a request to start working.		This authorization method will be used for even	ry request in this collection. You can overri	de this by specifying one in the request.	
History			Туре	OAuth 2.0 $\vee$		
0+			The authorization data will be automatically Learn more about <u>OAuth 2.0</u> authorization.	generated when you send the request.		
		,	Add auth data to	Request Headers ~		
		(1) Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about <u>variables</u> .				
			Current Token			
		Token Heade Auto-r Your e: Share This w	Token	Token	Available Tokens ~	
				Token		
				Header Prefix ()	Bearer	
				Auto-refresh Token Your expired token will be auto-refreshed be	fore sending a request.	
			Share Token This will allow anyone with access to this rec	quest to view and use it.		
		Configure New Token				
			Token Name	Enter a token name		
			Grant type	Authorization Code $\vee$		
			Callback URL ①	http://your-application.com/registered,		
□ ⊘ Online	Q Find and replace 🖾 Console	3		Authorizo using browsor	🖏 Posti	

#### Under the "Configure New Token" section, please fill out the following:

Token Name	This can be anything. EX: Token A
Grant Type	Leave as is
Callback URL	https://oauth.pstmn.io/v1/browser-callback
AuthURL	https://drchrono.com/o/authorize/
Access Token URL	https://drchrono.com/o/token/
Client ID	From the DrChrono API Page
Client Secret	From the DrChrono API Page
Scope	labs:read labs:write messages:read messages:write patients:read patients:write patients:summary:read patients:summary:write settings:read settings:write tasks:read tasks:write user:read user:write billing:patient-payment:read billing:patient- payment:write billing:read billing:write calendar:read calendar:write clinical:read clinical:write
State	Leave as is
Client Authentication	Leave as is

Once everything is filled, select the orange button at the bottom of the page "Get New Access Token".



You will be redirected to a prompt box to sign into DrChrono and then redirected to another screen to Authorize the connection...select "Authorize"

Congratulations! You received Access and Refresh Tokens. Select "Use Token" from Postman to set the access token to your collection.

DrChrono Collection							
Overview Authorization  Scripts Tests Variables Runs The authorization data will be automatically generated when you send the request. Learn more about <u>OAuth 2.0</u> authorization.							
Add auth data to	Request Headers		~				
Current Token		MANAGE AC	CESS TOKENS		×		
Token	Available Tokens	All Tokens	Delete 🗸	Token Details	Use Token		
	Token	wy token		Token Name	My Token 🥒		
Header Prefix (3)	Bearer			Access Token	zXiqpYWrgoJ90IPqPd8ESATkDI0VoR		
Auto-refresh Token Your expired token will be auto-refreshed be	efore sending a reque			Token Type	Bearer		
				expires_in	172800		
Share Token This will allow anyone with access to this red	quest to view and use			refresh_token	6npNQtxEzHGMmnpwu1pXZ0Chh90ZRY		
Configure New Token				scope	tasks labs:write billing:read patients:read user:write messages:writ e billing:patient-payment:write user tasks:write patients:summary: write calendar user:read billing:patient-payment tasks:read billing:		
Token Name	First Token				patient-payment:read patients:summary:read labs billing labs:read		
Grant type	Authorization Code				setungs.read bining.wine messages clinical:read patients message s.read clinical:write clinical settings:write settings patients:summar v.natients:write calenda:read calenda:write		
Callback URL ④	https://oauth.pstm	n.io/v1/browser	r-c				

### **First Request Call**

Let's create your first DrChrono API Request Call. Add a request within your collection and let's call the following endpoint: https://app.drchrono.com/api/users/current

When adding a request, **ensure that it is made inside your collection**. We want to inherit the token we generated when making API calls.



	+ =		00
Collections	✓ DrChrono Collection <sup>A</sup> / <sub>A</sub>		
Environments	This colle Add a req Share		
49	Move		
History	Run co	ollection	
	Genera	rate tests	
0+	Edit		
	Add re	equest	
	Add fo	older	
	Monito	or collection	

Here is how the request appears in Postman:

A DrChrono API	New Import	er New Regulat • +	No environn	ment v	EB
		22 DiChrono Collection / New Request	🖺 Save 👻	Share	P
DrChrono     OFT No	no Collection New Request	GET v Https://gop.dictoron.com/apil/users/current	Send	4 ~ L	
Environments		varias Authorization Headers (8) Body Scripts Tests Settings		Cookies	$\langle \rangle$
History		Auth Type			
00		Inhert auch from parent v			
0+		The authorization header will be automatically generated when you send the request. Learn more about <u>authorization</u> .			(j)
		This request is using Okuth 2.8 from collection <u>protocols of the collection</u>			
		Body Cookes (1) Headers (18) Test Results	[A] Save as example.	mple ***	
		Peetry Rev Preview Visualize JSON V 📅		1 Q	
		'secure': 'subd',           's_scrit': lab,           's_scrit': lab,           's_scrit': lab,           's_scrit': lab,           'stric': la			

#### **Alternative Configuration:**

Instead of creating a collection, you can configure a connection from the request level. To start, select the "New" or "+" button. If you select "New" you will be prompted to choose a specific method, select HTTP. If you select "+" icon, Postman will automatically create a new HTTP request.



From the new request, navigate to the Authorization tab and configure the new token. You can use the same details from above.

CET Untitled Request • + ~ ~						
11 Untitled Request						
GET v Enter URL or paste text						
Params Authorization ● Headers (7) Body Scripts Tests Settings						
Auth Type	Auth Type Current Token					
OAuth 2.0	Token	Available Tokens 🗸				
The authorization data will be automatically generated when you send the request. Learn more about OAuth 2.0		Token				
Add authorization data to Request Headers	Header Prefix (3)	Bearer				
	Auto-refresh Token Your expired token will be auto-refreshed before	e sending a request.				
	Share Token This will allow anyone with access to this reques	st to view and use it.				
	Configure New Token					
	Token Name	Enter a token name				
	Grant type	Authorization Code				
	Callback URL ④	http://your-application.com/registered/callbac				
		Authorize using browser				
	Auth URL <sup>①</sup>	https://example.com/login/oauth/authorize				
	Access Token URL ④	https://example.com/login/oauth/access_toke				
	Client ID (3)	Client ID				
	Client Secret ④	Client Secret				
	Scope ④	e.g. read:org				
	State (1)	State				
	Client Authentication (3)	Send as Basic Auth header				
	> Advanced					
	(3) Clear cookies					
	Get New Access Token					

Once you are finished select "Get New Access Token" and you should be able to receive a new set of tokens to use. In that same instance, you can run any API resource from that request tab.

#### **Refresh Method**

Lastly, I would like to introduce the refresh token method. This will help automate the process of grabbing new access tokens. Our access token has an expiration date of 48 hours. Our refresh token does not have an expiration, the refresh token only invalidates if you go through the OAuth process again or utilize the /o/revoke\_token endpoint.

Here is a Python snippet depicting the refresh method.

import requests
url = "https://drchrono.com/o/token/"
payload = 'client_id=1234&client_secret=1234&grant_type=refresh_token&refresh_token=1234' headers = { 'Content-Type': 'application/x-www-form-urlencoded' }
response = requests.request("POST", url, headers=headers, data=payload)
print(response.text)

Let's also take a look at this in Postman.

POST	POST v https://drchrono.com/o/token/						
Params	Params Authorization Headers (9) Body • Pre-request Script Tests Settings						
● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL							
	Key	Value	Description	••• Bulk Edit			
	client_id	1234					
	client_secret	1234					
	grant_type	refresh_token					
	refresh_token	1234					
	Key	Value	Description				

I hope this information was useful! If you have any questions please reach out to api@drchrono.com

#### References

https://app.drchrono.com/api-docs/#section/Introduction

https://app.drchrono.com/api-docs-old/v4/documentation#orevoke\_token