# GETTING AUTHENTICATION ACCESS TOKENS FOR MICROSOFT APIS



In order to use Microsoft Power BI or other Microsoft APIs, you have to obtain an access token, also known as a bearer token. This is because Microsoft uses oAuth2, an industry standard protocol, for authentication. (In other words, a simple API key or username with a password is not enough.)

In this tutorial, we explain how to do that.

(This article is part of our *Power BI Guide*. Use the right-hand menu to navigate.)

Note: We use curl to post data to Microsoft endpoints. That's like the command line version of Postman. On Mac and Ubuntu, curl is already there. You might have to install on Windows.

### **Registering Power BI**

If you're doing all this for the very first time, in order to perform both steps of oAuth2 authentication, there's a Step 0.

You first have to register your application as a means of getting credentials. You do that one time. This generates an application ID and secret key. For Microsoft Power BI, you do it like this:

First, log into the embedding tool at <u>https://app.powerbi.com/embedsetup/UserOwnsData</u>

This is not the same as logging into Azure and creating an application in Active Directory there. You are creating an application on Power BI's Azure account (if you want to think of it that way).

Next, fill out the screens below. Note that:

- For the URL, you can use any web page. You will look at the parameters passed to this web page as we show below.
- Skip the screen that says import content.
- For API access, click **select all**.
- At the end, copy and save the **Application ID** and **Application Secret**.

Microsoft Power BI Developer Embedding setup Then, when you're ready to build a production-ready solution, we recommend setting up service princi Need help? Check out our troubleshooting guide STEP 1 Sign in to Power BI STEP 2 **Register your application** Register your application with Azure AD to allow your application to access the Power BI REST APIs and to set resource permissions for your application. You can change this later in the Microsoft Azure portal. Learn more Application Name Enter a display name to identify your application in Azure Expenses Application Type Server-side web application (for web apps or web APIs) Home Page URI



### **Using oAuth2 for rest APIs**

Once you've registered, you can move to this step.

Basic authentication is when you need only a user ID and password for access to something.

But Microsoft uses oAuth2 authentication. Microsoft APIs require that you present an Authorization header in order to use the API. Basically, oAuth2 is a two-step process:

1. Do a POST to login.microsoftonline.com

2. Take the access/bearer token from Step 1 and pass that to the API in a header called Authorization for whatever API you are calling.

# Getting a token (code)

To get the authorization code, click on this URL to open a browser:

https://login.microsoftonline.com/common/oauth2/authorize?client\_id=(appid)&response\_type=co de&response\_mode=query&redirect\_uri=(url you put when you registered app)&scope=openid&state=foo

Basically, it will take you to the URL you put when you registered the application. But a screen will pop up asking you to grant certain permissions:

- response\_type: code
- response\_mode: query
- state: foo (Sny value will work here, it's just a place for free form-data.)
- scope: openid (You could also add offline\_access.)
- **url**: We use the same URL throughout but change the URI to **authorize** and then **token** later to call different Microsoft endpoints.

Note: Here, the **tenant ID** is common, not a multi-tenant ID. Common means to retrieve the tenant ID associated with your Azure account.

Microsoft							
werowe@walkerrowe.com							
Permissions requested Review for your organization							
expenses unverified							
This application is not published by Microsoft.							
This app would like to:							
View all datasets							
✓ View all dashboards							
✓ View all reports	✓ Sign in and read u	user profile		1			
View all workspaces	Consent on behalf of your organization						
✓ View all capacities	If you accent this app will get access to the specified resources						
$\checkmark$ View all storage accounts	for all users in your organization. No one else will be prompted to review these permissions.						
View all dataflows	Accepting these permissions means that you allow this app to						
✓ View all gateways	use your data as specified in their terms of service and privacy statement. The publisher has not provided links to their terms						
✓ View all Power BI apps	for you to review. You can change these permissions at https://myapps.microsoft.com. Show details						
✓ Read and write all datasets	Does this app look suspicious? Report it here						
✓ Read and write all dashboards		<b>a</b>					
✓ Read and write all reports		Cancel	Accept	Now,			

you certainly could have written some kind of web listener to retrieve the code that Microsoft created. But we will just use the debugger in a Chrome browser to see the query parameter that Microsoft passed to our web page.

When Microsoft redirects you to the web page you indicated, go to the **network** tab in the browser and click the refresh button on the browser.

Then click on the **code** field and press **Copy as cURL**. The code (token) appears as the query parameter code as shown below.

### https://walkercodetutorials.com/?code=0.ASsARY...

If you are wondering at this point why the URL is not some URL in Power BI, that's because you registered the application in Power BI. So, Microsoft knows that Power BI is what you want to use. The redirect URL serves merely as a place to retrieve this code.

Going forward, you would not want to click on the browser every time—this is not how a batch program would work. So look at the **prompt** setting in the <u>Microsoft Identity Platform reference</u> <u>guide</u> to see how to change that.

$\leftrightarrow$ $\rightarrow$ $G$ $\oplus$ $w$	valkercodetutorials.com/	code=0.ASsARYN0k	wAbXkOtW	3fxQAoDm	Z0-8QQVj1RKtoGDl9h	eKYkrAOk.AQABAAIA	AADDL	A3VO7QrddgJg	7Wevr29qcLru-xV	@ ☆	
II	🕞 Elements Console Sources Network Performance Memory Application Security Lighthouse										
<b>VValk</b>	🔴 🛇   🝸 🔍   (	Preserve log Dis	able cache	Online	* ± ±						
	Filter	🗌 Hide data UR	Ls 📶 🛛 XHF	JS CSS	Img Media Font Do	c WS Manifest Other	r 🗌 Has	blocked cookies	Blocked Reque	sts	
Code	50 ms	100 ms		150 ms	200 ms	250 ms	8	300 ms	350 r	ns	
Tutor											
i atoi	Name			Status	Туре	Initiator		Size	Time	Waterfall	
	?code=0.ASsARYN0k	wAbXkOtW3fxQAoDmZ	0-8QQVj1	304	document	Other		191 B	250 ms	-	
Walker Rowe is a freelance tech writer and data	https://walkercodetutorials.com/?code=0.ASsARYN0kwAbXkofW3fxQAoDmZ0-8QQVj1RKtoGDI9heKYkrAOk.AQABAAIAADDLA3VO7QrddgJg7Wevr29qcLru- xVOFIKszxuzC35Zqvqufu1KFwHAo1Ob2WwAlmbS8Koxalis71PG3Jcit5HNErYeLqHuhiv1hRAPgMq4dxn6hhfQ3-unIV57Fc8- tyNrZowpJsKmmjUrhWdonsemTw0cDXzFespQDDsc_6ZFcTCRkFFSmv1Ws0Gstiu/Wr8_oDGNEPE7MbJYKuqtyaHz7Bw3x3qmIWyERdOc0F7Uk3DVoCUtUMQN4FMMNetvrHrVevgA3 X69KG4SKjeacs92TqwTy2CpjRePuRhoM1lhlq4osoCjqznOhHeo0fkDWmrAca7x7HQuzxJuCNGEOo_dNJ1xikP9CHWPQIQFHP4_8JQZpBuZ67F0LFPaqECGmMkrKxJYshiMRYrKeRs6 6s_uWPENmzBvTAcv_mcJamGSm0C1xjT6_tNjWW5SISRWB_eWWWp7r8LE8rTdQ492u5FN6e3dQ_L1Z38L- iboerrBRNpJCGL3ofA0G6qUdd3HIDN5iwdh0Vw60j1tB_L0mGdxkAz0hR1ahBtNNAnf10ezkDuGmqppL37484MXPEqYbDsh3zv4Se1795IKg0Kg_pNqHcQPGSzIzqATaD5PwS70ACjHF 6LzsHVXNi23Q6wl8cEMyVJhQ7hVsGWTM3RsTNIYAXHVu7F8u47J2VF6enRitDSYtMPYLj8k5G2K7W7vMRIMz_z4_3- g01NcQT0gnjz4N3U3I374UDF12VKkeUELEmrf1cOzhTSh011M_YbJLniSfApAVvZZPECLIyTm0oefrcsQD- DNCi8JooTOY084gJuDMhvDkokH44UJvonoprlDi00JKYDSXd8gAA&state=foo&session_state=828b86e1-a897-4d6a-9fcd-e848f660e31f										
science and big data programmer	New CSS angle vis Better visualize and Emulate unsuppor	sualization tools edit CSS angle in the ted image types	Styles pane.				$\cap$				

# Getting an access token

We use curl to illustrate the next steps. Get the access token (bearer token) this way.

The values are:

- grant\_type: Put "authorization\_code"
- client\_id: Application ID from above (The dots above hide my actual ID.)
- client\_secret: Application Secret from above
- redirect\_uri: Same as above
- scope: Same as above
- url: Note that the endpoint has changed to token

curl -X POST --form 'grant\_type=authorization\_code' --form 'client\_id=7...5'
--form 'client\_secret=21dVzEgtjUhfyZS3AJDaH0eMYB0q0ovYeH4YUoa//FM' --form
'scope=openid%20offline\_access'--form 'response\_type=code' --form
'redirect\_uri=https://walkercodetutorials.com/' --form 'code=0.AS...AA'
https://login.microsoftonline.com/common/oauth2/token

#### Returns:

{"token\_type":"Bearer","expires\_in":"3599","ext\_expires\_in":"3599","expires\_o
n":"1614591204","not\_before":"1614587304","resource":"https://analysis.window
s.net/powerbi/api","access\_token":"ey....G8CYZQT6t2p5IC1r3E7D\_koNqc6h\_f3918o\_BP2N0YOweCKKZ7WCw"}

### **Testing your Microsoft API access**

Take the **access\_token** value from the previous step and add it as an **Authorization** header value as shown below. (You have one hour before it expires.)

This, for example, is how you return a list of datasets in Power BI in My workspace. (That's the default workspace for free Power BI accounts, meaning for one individual's use only, as opposed to, for example, an enterprise account.)

Note: myorg does not mean your org. It's just a placeholder required by Microsoft.

curl -X GET -H "Authorization: Bearer ey....W\_A" -H "Content-Type: application/json" https://api.powerbi.com/v1.0/myorg/datasets

That concludes this tutorial.

# **Related reading**

- BMC Machine Learning & Big Data Blog
- Power BI Basics: Creating a Pie Chart with Power Query Editor
- Data Visualization Guide, a series of tutorials
- Data Storage Explained: Data Lake vs Warehouse vs Database
- Enabling the Citizen Data Scientists
- MySQL vs MongoDB: Comparing Databases