

Kippy API Integration with Power BI using JWT for Authentication

Nauman Khan 29-Apr-2024



Intro

This paper provides details of a proof of concept that integrates a kippy data endpoint using basic authentication and a JWT token from PowerBI desktop. The outcome successfully showed how kippy users could dynamically pull data from kippy to PowerBI.

The example uses the /api/v4/query/ data endpoint, which when called with the parameter "table=user" returns all users in the kippy instance. See <u>https://www.kippy.cloud/api</u> for more information on this and other data endpoints.

Background

What is Basic Authentication?

Basic Authentication is a simple and widely used method for authenticating users on the web. It is a part of the HTTP protocol and involves sending a username and password in the request headers. The basic idea is to include a "Authorization" header in the HTTP request, which contains the word "Basic" followed by a space and a base64-encoded string of "username:password".

Here's a basic example of how it works:

- 1. The client (such as a web browser or a software application) sends an HTTP request to a server.
- 2. The server responds with a 401 Unauthorized status code, indicating that authentication is required.
- 3. The client includes an "Authorization" header in the request with the credentials encoded in base64.

For example:

Authorization: Basic dXNlcm5hbWU6cGFzc3dvcmQ=

In this example, "dXNlcm5hbWU6cGFzc3dvcmQ=" is the base64-encoded form of "username:password".

What is a JWT token?

JWT stands for JSON Web Token, and it is a compact, URL-safe means of representing claims to be transferred between two parties. JWTs are often used for authentication and authorization in web applications.

JWTs are commonly used for authentication by generating a token on the server side, sending it to the client upon successful login, and having the client include the token in the headers of subsequent requests. The server can then verify the integrity and authenticity of the token using the stored secret or public key.



What is PowerBI?

Power BI is a business analytics service developed by Microsoft that allows users to visualize and analyze data from a variety of sources, helping businesses make data-driven decisions. It's a suite of tools designed to transform raw data into interactive dashboards, reports, and visualizations, providing insights for better decision-making.

Here's an overview of some key features and components of Power BI:

Data Connectivity: Power BI connects to a wide range of data sources, including Excel spreadsheets, SQL databases, cloud services, and online platforms. This flexibility allows users to import and combine data from multiple sources.

Data Transformation: The Power Query editor in Power BI allows users to clean, transform, and prepare data for analysis. Users can reshape, filter, and manipulate data as needed before creating visualizations.

Visualizations and Reports: Power BI offers a variety of visualization options, such as charts, graphs, maps, and tables, to represent data. Users can create interactive reports with multiple visualizations, allowing them to explore data and discover insights.

Dashboards: Power BI dashboards are collections of visualizations that provide a high-level overview of key metrics and insights. Dashboards can be customized to focus on specific areas of interest and shared with others.

Power BI Service: This is the cloud-based component where users can publish and share their reports and dashboards. It supports collaboration and sharing within an organization and includes features like scheduled data refreshes, security settings, and mobile access.

Power BI Desktop: A Windows application used to create reports and visualizations. It's the primary tool for data transformation and report design before publishing to the Power BI Service.

Collaboration and Sharing: Power BI enables users to share reports and dashboards with colleagues or teams. It includes features like data-driven alerts, collaboration, and integration with Microsoft Teams.

Data Analysis Expressions (DAX): Power BI supports DAX, a formula language used to create custom calculations, measures, and aggregations, providing advanced analytical capabilities.

Overall, Power BI is designed to make data analysis accessible to a wide range of users, from business analysts to data scientists, and is commonly used in business intelligence and data analytics contexts to support decision-making and strategic planning.



Approach

A kippy instance was integrated with PowerBI Desktop, so that that the information in the kippy instance was pulled via the existing kippy data endpoint APIs – and then used within PowerBI Desktop to create bespoke visualisations on the returned data.

The integration between kippy and PowerBI was done using JWT authentication. Therefore, the JWT token would first be retrieved by PowerBI from the kippy token provider URL – and then passed through on all subsequent calls by PowerBI to the kippy data endpoint APIs.

To do this, PowerBI Desktop needed to be configured in a very particular way, with some bespoke "query" code provided to PowerBI Desktop.

One thing to note is that in a recent version of PowerBI Desktop, Microsoft put in an undocumented restriction, such that the "Authentication" header was no longer sent on calls made from PowerBI Desktop. This approach includes a work around for for that issue.

Prerequisites

The POC was done using the following version of Power BI Desktop. Screens and functionality may differ for other versions.

Microsoft Power BI Desktop

Microsoft Power BI Desktop is a companion product to app.powerbi.com

Version: 2.127.1235.0 64-bit (March 2024)

Also, the POC assumes you have access to a kippy instance with some existing data. If you need to, create a new kippy instance at <u>www.kippy.cloud</u> and use that to test your connectivity. The following uses kippy instance with username owner@test.kippy.jwt.powerbi.com and password jBkQdi08QB3gPB2d as an example.

To ensure the connectivity is working as expected, it is highly recommend to use a tool like postman.com to first ensure API access is working as expected outside of Power BI.

First, we call <u>https://kippy-tsc.appspot.com/jwt/token</u> with the username and password using Basic Authentication to get a token.





Home Workspaces \checkmark API Network \checkmark			🐥 Invite 🎼 🏠 🏠 Upgrade 🗸
ິ My Workspace New Import		GET Get data • +	
□ + = ∞∞ Collections ~ REST API basics: CRUD, test & variable		est & variable / Get data	🖺 Save 🗸 🍠 🖹
ି ତିତ୍ର ଜୁଣ data	GET ~ https://kip	py-tsc.appspot.com/jwt/token	Send 🗸 🛡
APIs Post Post data	Params Authorization		
DEL Delete data	Type Basic A	 Heads up! These parameters hold s collaborative environment, we recorr 	rensitive data. To keep this data secure while working in a $ imes$ 2^* mmend using variables. Learn more about <u>variables</u> .
	The authorization header will be automatically generated when yo send the request. Learn more abo <u>Basic Auth</u> authorization.	out Username Password	owner@test.kippy.jwt.powerbi.com j8kQdi08Q83gPB2d
	Body Cookies (2) Headers (1) Pretty Raw Preview 1 [*access_token*: 2 2 *eyJhc5(01520114) eyJpc3Mi01JraXE JpYx010jE3MTM5C Dyxx0V0nGb_3HE DX2-PAY/WY3107 Z9vyDKsod3duzdf sseUeEXJ5PYbvbJ SseUeEXJ5PYbvbJ	 Test Results (1/1) Visualize HTML ✓ ⇒ HTML ✓ ⇒ Bwe55jbG91ZCIsInN1YiI6Im93bmVyQHRlc3Qua2 DTU1NDcsInVzZXJurW11joib3duZXJAdGv2d65 DTUSMbcsInVzZXJurW11joib3duZXJAdGv2d65 Zvz7Fh9H21t2Bfwz6a2mo6v41k030IrtxeNzP15 EXMuOKn53dm1ssHBfbKIZ_26IHXY8wxamJroky Ixx005Hb10⁻} 	200 OK 1492 ms 1.55 KB Save as example *** C Q 21wcHkuand@LnBvd2VyYmkuY29tIiwiZXhwIjoxNzE0MDgxOTQ3LC 21wcHkuand@LnBvd2VyYmkuY29tIiwiZXhwIjoxNzE0MDgxOTQ3LC 21wcHkuand@LnBvd2VyYmkuY29tIiwiZXhwIjoxNzE0MDgxOTQ3LC 12f57mb40c0910000000000000000000000000000000000
1 O Online 🖸 Console			③ Postbot I Runner ⊙ Auto-select agent ⑤ Cookies 前 Trash 昭 ③

Next, we call the data endpoint API using Bearer token authorization.

Home Workspaces \checkmark API Network \checkmark	۵		🐥 Invite 🔯 🗘 🏠 Up	grade \vee
		ET Get data • GET https://kippy-tsc.appsp • +		
Collections + =	https://kippy-tsc.appspot.com/ GET https://kippy-tsc Params • Authorization • Head Type Bearer ` `	api/v4/query/?username=owner@test.kippy.jwt.pow e.appspot.com/api/v4/query/?username=owner@test.k ers (7) Body Pre-request Script Tests S ① Heads up! These parameters hold sensitive da collaborative environment, we recommend usir	erbi.com&organi 😭 Save 🗸 🖉 🛛 ippy.jwt.powerbi.com&organisation: Send iettings Cook ret. To keep this data secure while working in a rg variables. Learn more about <u>variables</u> .	
₽₽	The authorization header will be automatically generated when you send the request. Learn more about <u>Bearer Token</u> authorization.	Token	eyJhbGciOiJSUz11NiJ9.eyJpc3MiOiJraXBwe	
	Body Cookies (1) Headers (9) Te		🔁 200 OK 781 ms 946 B 🖺 Save as example	
	 [{ "key": { "kind": "User", "id": 4590340738646016 , "propertyMap": { "orlent": false, "role": "Admin", "unlicensed": false, "created": 1799528401 "organisation": "test.k "employeeDD": "", "isFree": false, "encrvoted": true. "encrvoted": true. 	4, ippy.jwt.powerbi.com",		
🗉 🕑 Online 🖾 Console		🕲 Postbot	🕒 Runner 🕑 Auto-select agent 🛞 Cookies 🏢 T	ash 🖭 (?)

You can see the response is some JSON showing the users in the kippy instance.



By the way, make sure POSTMAN is not using any cookies when making the calls to the data endpoint API. If it is, delete those Cookies (by click the blue Cookies text near the top right, and delete the cookies by clicking the X next to the cookie names.

				ø 4 🌍	
	Manage Cookies				
			Add domain		
	kippy-tsc.appspot.com 2 cookies		×		
ے History					
	Domains Allowlist	C14	ar All Cookies		
	II OIXanisacion .				







	Manage Cookies					
				Add domain		
	kippy-tsc.appspot.co	om 0 cookies				
	Domains Allowlist					

As mentioned earlier, Microsoft recently introduced an undocumented Power BI change that stopped headers called "Authorization" being sent by Power BI. Therefore, as a work around, instead we will pass an Authorization header called "Autho" to kippy, which has made a change to handle this situation.

So, once again, clear all the cookies in Postman and change the Authorization type to No Auth.



Home Workspaces	Q Search Postman 🐥 Invite	्रि 🗘 🌍 Upgrade 🗸	
ိ My Workspace New Import	⊗ Overview GET Get data ● GET https://kippy-tsc.appsp ● + ~ ~		
다 + 코 ** Collections · REST API basics: CRUD, test & variable ⓒ orr Get data	mttps://kippy-tsc.appspot.com/api/v4/query/?username=owner@test.kippy.jwt.powerbi.com&organi GET https://kippy-tsc.appspot.com/api/v4/query/?username=owner@test.kippy.jwt.powerbi.com&organi	🖺 Save 🗸 🖉 🗐 </td <td></td>	
APIs Post Post data PUT Update data Environments 0E. Delete data	Params Authorization Headers (6) Body Pre-request Script Tests Settings Type No Auth This request does not use any authorization. Learn more abou		
	Body Cookies (1) Headers (9) Test Results	46 B 🛱 Save as example ••• C Q	

Now, pass in a header with the key "Autho" and the value starting with the word "Bearer", followed by a space, followed by the token.

Home	Workspaces 🗸 🛛 API Network					🔑 Invite 🔯	û 🌍 Upg	rade \vee
ို My Wor	rkspace	New Import			GET https://kippy-tsc.appsp • +			
Collections		°°°	₩ https://kippy-tsc.apps	pot.com/api/v4/query/?us	ername=owner@test.kippy.jwt.powerbi.com	&organi 🖺 Save		
APIs	GET Get data		GET ~ https://	kippy-tsc.appspot.com/api	/v4/query/?username=owner@test.kippy.jwt.p	powerbi.com&organisation:	Send ~	
Environments	POST Post data PUT Update data DEL Delete data		Params • Authorization Headers © 6 hidden	Headers (7) Body				
4			Key		Value	Description ••• E		
History			🖌 Autho		Bearer eyJhbGciOiJSUzI1NiJ9.eyJpc3MiOi			
gọ								
			Body Cookies (1) Headers					
			Pretty Raw Previe					
			<pre>1 [{ 2 "key": { 3 "kindt: "User", 4 "id": 4590340738 5 }, 6 "propertyMap": { 7 "silent": false, 8 "role": "Admin", 9 "unlicensed": false, 10 "created": 17139 11 "organisation": 12 "employeeID": "" 13 "isFree": false, 4 "encrvoted": true.</pre>	646016 95284014, "test.kippy.jwt.power e.	⊇i.com",			
🗉 📀 Online							🚯 Cookies Tra	

As you can see, the result is returned as before.



These steps confirm that the API is returning as we need it to. Next, we will do the same from within Power BI.

Integration

In this section, we will set up Power BI Desktop to connect to the same datapoint API using the same JWT authentication mechanism using a header named "Autho" with the value of "Bearer" followed by the token.

1) The first thing to do in Power BI Desktop is to create a Blank Query.

File	Home	Trans
Close & Apply •	New Source •	Recent Sources
x	Excel Work	book
	SQL Server	d O-
	Analysis Se	rvices
	Text/CSV	.1
	Web	- 1
=	OData feed	
	Blank Quer	y
	More	

- 2) Next, create the following parameters to be passed to the m query with the following name and values
 - Username = owner@test.kippy.jwt.powerbi.com
 - Password = jBkQdi08QB3gPB2d
 - o Grant_type = password
 - Client_id = test.kippy.jwt.powerbi.com



- Properties Data Type: Text ▼ Z↓ Split Group III Use First Row : **b m** -8 Choose Remove Columns • Columns • Use First Row as He Data source settings lose & New Recent Enter Apply Source Sources Data Manage Refresh Parameters • Preview • III Manage • Split Group Column - By 2 Replace Values Close Data So Parameters Manage Columns Redu Sort Query Queries [6] < f_X = Csv.Document(Web.Contents(api_url, [Headers=[Authorization="Bearer" & " " &AccessTo 🗵 User - AB- CO **T** 🗄 Pa Manage Parameters 🧾 grant_typ client id client secret A^B_C Username × Usernam API user table 3 A^B_c Password Description A^B_C grant_type 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 A^BC client_id A^B_C client_secre Required Type Text Suggested Values ÷ Any value Current Value 24 25 26 27 OK Cancel 28
- Client_secret = test.kippy.jwt.powerbi.com.secret

3) Now, paste the following code in to the query.

```
let
    //IIRIs
    api url = "https://kippy-tsc.appspot.com/api/v4/query/?username=
owner@test.kippy.jwt.powerbi.com&organisation= test.kippy.jwt.powerbi.com&table=user",
//Insert your API endpoint
    token_url ="https://kippy-tsc.appspot.com/jwt/token", //Insert your token URL
    //Client credentials
client_credentials = "grant_type=" &grant_type& "&username=" &Username&
"&password="&Password&"&clienttype=User",
    //Get JSON Web Token via API
    EncodedCredentials = "Bearer "& Binary.ToText(Text.ToBinary(client_id & ":" &
client_secret), BinaryEncoding.Base64),
    Token_Response = Web.Contents(token_url, [Headers=
        [Autho=EncodedCredentials, #"Content-Type"="application/x-www-form-
urlencoded;charset=UTF-8"],
        Content=Text.ToBinary(client_credentials)]),
    //Get Access Token
    FormatAsJson = Json.Document(Token_Response),
    AccessToken = FormatAsJson[access token],
    //Get Data
    GetJsonQueryAPI = Csv.Document(Web.Contents(api_url, [Headers=[Authorization="Bearer " &
    &AccessToken]]),[Delimiter=";
    Encoding= TextEncoding.Utf8, QuoteStyle=QuoteStyle.None])
in
    GetJsonQueryAPI
```

4) If Power BI prompt Please specify how to connect.

- a. Click the Edit Credentials button
- b. Make sure to select Anonymous
- c. Click Connect





Please specify h	now to connect. Edit Credentials
	Access Web content ×
Anonymous	https://c
Windows	Use anonymous access for this Web content.
	Select which level to apply these settings to
Basic	https://
Web API	
Organizational a	account
	Connect Cancel

5) Ensure the encoding is UTF8.

To do this you might have to change "Encoding=1252" to "Encoding= TextEncoding.Utf8"



6) The result should be like below:

× √ f _X = Csv.Document(Web.Contents(api_url, [Headers=[Authorization="Bearer " & " " &AccessToken]]),[Delimiter=";",	Vuery S	attings	×
al A ⁸ , Column1	4 PROPE	RTIES	
	Name		
Valid 99%	API u	ser table	
• Endt 0%	All Dec		
1 6	AirPio	Jerues	
2 "key": {	APPLI	D STEPS	
3 "kind": "User",	ap	iurl 🤅	0
4 "id": 4594198290366464	tol	ken url 🔅	1
5 1.	cli	ent_credentials ①	0
6 "propertyMap": {	En	codedCredentials (0
7 "silent"; false.	To	ken_Response	*
8 "role": "Manager".	Fo	rmatAsJson 🔅	0∦
9 "unlicensed" false	Ac	cessToken (0
10 "created":1709125601718.	Ac	cessTokenHeader 🤅	1
11 "organisation"	×Ge	tJsonQueryAPI C	0 🕸
12 "isFree": false			
13 "encryoted":true.			
14 "isQuer" fale			
susubcribe" face			
16 "erade"."			
0 greet 1/109467748597			
a name".			
), and a second se			
0 "hoardid" "da018430-h2a1-4373-954a-3f4de5854419"			
20 Contract - Contract of Contract - Contrac			
21 Sugaran			
24 Int 75 Wayny			
23 myrt 1 26 "kindt"-"licee"			
77 "Ir"- 4554459399438336			
28 b			
2 "monertyMan" (
30 and 1 liker			
21 Total State			
3 "reated" 10015721669			
Jog urgenhautor			



7) Next, we need to transfer csv data format and then to JSON format

a. Transpose the table like so

1 ₉₂ Replace Values ▼ 🖫 Unpivot Columns ▼ ↓ Fill ▼ 🔡 Move ▼	Split Form	Merge Columns	X Stati	stics Standard Scientif		rigonometry ~ counding ~	Date Time	Duration	Run R Run Python	n						
Any Column	Column • •	ext Column		Number Co	olumn	ntormation *	Date & Time	Column	script script Scripts							
× √ f _X = Table.Transpose(Ge	tJsonQueryAP	I)											~	Qu	ery Settings	
ABC Column1	-	123 Column2	¥	ABC 123 Column3	٣	123 Column4	٣	123 Column	15 💌	ABC 123 Column6	٣	ABC 123 Column7	¥ 12	BC ∡ P	ROPERTIES	
Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	 Valid Error Empty 	100% 0% 0%	 Valid Error Empty 	100% 0% 0%	 Valid Error Empty 	100% 0% 0%	 Valid Error Empty 	100% 0% 0%	 Valid Error Empty 	100% 0%		API user table	
1 [["key": {		"kind": "User",		"id": 459419	3290366464	}.		"propertyMap": {		"silent": false,			Properties	
														1	api_url token_url client_credentials EncodedCredentials Token_Response FormstAslson AccessToken Access	rtyM

b. Merged the table to one column

/ Tools Help		^ (
t _{w2} Replace Values - ¹ / ₁₀ Unpivot Columns - pe □ Fill - □ Pivot Column Any Column Convert to List Any Column Tet Column Tet Column	s Xor Bart 10 ² Arigonometry Statistics Standard Solertific Number Column Number Column	
Y		Ouery Settings X
y y = Table.CompineColumns(a" Pransposed Table , { Columnation of the column (a transposed Table) { Columnation of the column (a transposed Table) { Columnation of the column (a transposed Table) { Columnation of the columnation of the column (a transposed Table) { Columnation of the columnation of the column (a transposed Table) { Column	nı, columnz, columns, column», columns, column», column», columne, columne, columne, columne, columne, columne,	PROPERTIES Name
• Error 0%		API user table
• Empty 0%		All Properties
1 [{ "key": { "kind": "User", "id": 4594198290366464 }, "propertyM		A APPLIED STEPS
		aniur
		token url
		client credentials
		EncodedCredentials ①
		Token_Response 🚸
		FormatAsJson ① 🕸
		AccessToken ①
		AccessTokenHeader ①
		GetJsonQueryAPI ① 🕀
		Transposed Table
		× Merged Columns 0
		Parsed JSON ()
		Expanded Merged
		Expanded Merged I SP
		Expanded Merged key
		Capanded Mergedikey

c. Convert the table now to JSON

0 ×



e 🐺 Fill * 🔢 Move * Split Form	Aff Statat シン 部語 AD % Rounding * Date Time Dura	Run R Run Mython
Pivot Column 🔋 Convert to List Column * Ter	をParse Column Number Column Date & Time Colum	script scripts
 fx Table.CombineColumns (#*Transpoor Widd Fror Empty (1 *key: (*kind: *User) (2 *key: (*kind: *User) (3 *key: (*kind: *User) (4 *key: (*kind: *User) (5 *key: (*kind: *User) (7 *key: (*kind: *User) (8 *key: (*kind: *User) (9 *key: (*kind: *User) (1 *key: (*ki	d Table", ("Columni", "Columni", "Columni, "Column	Lum?", "Columă", "Colum18", "Colum11", " Query Settings PROPERTIES Name Al user table All Properties APUED STEPS api_ort all Properties APUED

d. Start expand the table up to your need of data

8) Now you can start use the data for your visualization

Detect Data Ty Rename	♣ Replace Values ▼ e ▼ Fill ▼ Pivot Column	Move • Convert to List	Split Format	Ins XO Trigonometry - Statistics Standard Scientific	Date Time Duration	R Py Run R Run Python script script			
	Any Column		Text Column	Number Column	Date & Time Column	Scripts			
<	$\times \sqrt{f_x}$	= Table.TransformCol	umns(#"Merged Columns".{}.Js	on.Document)			~	Query Settings	\times
	ABC Merged		5.0					▲ PROPERTIES	
	123	E.	and to New Perus					Name	
	Valid Frror	EX	pand to New Rows					API user table	
	 Empty 	EX	tract values					All Properties	
	1 List								
								APPLIED STEPS	
								api_url	0
								token_url	0
								client_credentials	1
								EncodedCredentials	0
								Token_Response	*
								FormatAsJson	0*
								AccessToken	()
								AccessTokenHeader	0
								GetJsonQueryAPI	0 *
								Margad Columns	0.8
								X Parrod ISON	0
								Expanded Merged	0
								Expanded Merged1	-8-
								Expanded Merged prop	ertvM 🗧
								Expanded Merged key	-5-

The query can now be used to create dashboards and visualisations in Power BI Desktop, which can be 'published' for other Power BI users to use and interact with.



Anna Canada and Anna Anna Anna Anna Anna Anna	1 100			Dashboard		And Sec. of	1.00	
19	20	¹ 0 ¹ 10	11	Partness 11K	Attact Torres 18K	Dijactive by Scott	n Propancy Record Process Proc.	
Peer Here & Fast			10 10 10 10 10 10 10 10 10 10 10 10 10 1			And have a set		
			n tarkas					

What next?

You can now change the username and password to your live kippy instance to ensure the information is returned.

You can also create other Power BI Desktop queries for different datapoint APIs.

This should allow you to create visualisations based on the information from one or more queries and your other data sources.

Conclusion

Kippy APIs can be called using JWT tokens to pass information to Power BI Desktop.

Other mechanisms are also available for connectivity and access your data in kippy.

Contact us at support@kippy.me for more details.