

## Power BI Rest APIs with Python



### **Cognitive Convergence**

<http://www.cognitiveconvergence.com>

+1 4242530744

[shahzad@cognitiveconvergence.com](mailto:shahzad@cognitiveconvergence.com)

**Cognitive Convergence** is Subject Matter Expert in Office 365, Dynamics 365, SharePoint, Project Server, Power Platform: Power Apps-Power BI-Power Automate-Power Virtual Agents.

We offer Power BI consulting services covering solution architecture refinement, customization, integration, transformation, visualization and analytics to uncover insights hidden within data and enhance data exploration.

## Contents

Objectives.....	2
Background.....	2
<i>Power BI Rest API</i> .....	2
<i>Python</i> .....	2
<i>Microsoft Azure Active Directory Authentication Library (ADAL) for Python 2</i>	
Python integration with power BI rest API.....	3
<i>Pre-requisite</i> .....	3
Register APP in Azure.....	3
Power BI Rest API.....	11
Python PACKages to Integrate Power BI Rest API.....	11
<i>Pypowerbi</i> .....	11
<i>ADAL Library</i> .....	11
Generating an Access Token.....	12
Using the API.....	13
<i>List of Groups</i> .....	13
<i>List of Reports</i> .....	14
Demo of List of Groups.....	15
Conclusion.....	16



## OBJECTIVES

---

This paper will discuss about Python and its integration with power BI rest API. Their development patterns and implementation details along with some examples and solution architecture.



## BACKGROUND

---

Python library for Power BI is Loosely modelled after the C# Power BI library to keep things somehow consistent. It is use to integrate with Power BI Rest API.

### Power BI Rest API

REST APIs created to make data collected by Application Insights easily available. Using these APIs enables to build new visualizations of application's data and extend the capabilities of Application Insights. Power BI Rest API provide different development methods all demand Authentication of Power BI, either direct authentication to Power BI server or first registering app in Azure and then authenticate the app with help of application id. The use these rest API HTTP in development patterns to achieve required results.

### Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

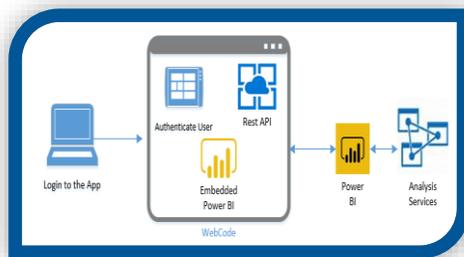


### Microsoft Azure Active Directory Authentication Library (ADAL) for Python

Azure Active Directory Library for Python is an authentication library which enables Python applications to authenticate and acquire tokens from Azure AD and ADFS to access protected web APIs (Microsoft APIs or applications registered with Azure AD).

## PYTHON INTEGRATION WITH POWER BI REST API

Python is an object oriented programming language and a large number of libraries available in python allows integration of Python with Power BI by using Rest APIs. These rest APIs allow accessing multiple features of Power BI with help of Power BI. It requires registering App in Azure and then authenticate user by generating access token in Python this Access token allow to generate embed token to embed in Power BI which then allow to access multiple features of Power BI using Rest APIs.



### Pre-requisite

Following are the pre-requisite for integration of Python with Power BI Rest API.

- An Organizational Active Directory, and a global admin on it
- A PowerBI Pro Licence
- A user in AD that is also logged in to Power BI.

### REGISTER APP IN AZURE

The Power BI API contains many useful features if looking to interact with Power BI at the API level. To use it, need to register an Azure App first.

- Go to <https://dev.powerbi.com/apps> and Sign In:

We will help you refresh all your Power BI  
Datasets with Python.

#### Cognitive Convergence

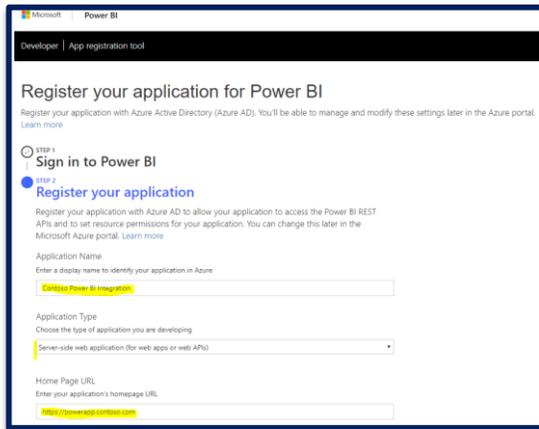
<http://www.cognitiveconvergence.com>

+1 4242530744

[shahzad@coognitiveconvernaence.com](mailto:shahzad@coognitiveconvernaence.com)



- Give the App a name, such as Contoso Power BI Integration, and select the Application type and home page URL. Here select Server-side web application:



- Enter a redirect URL and select which Power BI access the app should have. Select All, meaning the app has access to Reading, Writing and Creating:

Redirect URL  
Enter a URL where users will be redirected upon sign in so your application can receive an authorization code.

API access  
Select the APIs and the level of access your application needs. You can change these settings later in the Azure portal.  
Learn more

Select all

Read only APIs	Read and write APIs	Create APIs
<input checked="" type="checkbox"/> Read all datasets	<input checked="" type="checkbox"/> Read and write all datasets	<input checked="" type="checkbox"/> Create APIs
<input checked="" type="checkbox"/> Read all dashboards	<input checked="" type="checkbox"/> Read and write all dashboards	
<input checked="" type="checkbox"/> Read all reports	<input checked="" type="checkbox"/> Read and write all reports	
<input checked="" type="checkbox"/> Read all workspaces	<input checked="" type="checkbox"/> Read and write all workspaces	
<input checked="" type="checkbox"/> Read all capacities	<input checked="" type="checkbox"/> Read and write all capacities	
<input checked="" type="checkbox"/> Read all storage accounts	<input checked="" type="checkbox"/> Read and write all storage accounts	
<input checked="" type="checkbox"/> Read all dataflows	<input checked="" type="checkbox"/> Read and write all dataflows	
<input checked="" type="checkbox"/> Read all gateways	<input checked="" type="checkbox"/> Read and write all gateways	
<input checked="" type="checkbox"/> Read all Power BI apps		

By clicking Register, you agree to the terms of use.  
Note: An application registered here can't be used as a service principal. Learn how to register a service principal.

Register

- Once Registered, provided with an Application ID and an Application Secret. Click Close:

Success!

Your application has been registered.

Note: You can retrieve your client ID from the Azure portal, if needed. If you lose your client secret, you'll need to create a new one in the Azure portal!

Application ID:  
ea59eab2

Application secret:  
74XOw

Close

- The app has now been registered

Microsoft | Power BI

Developer | App registration tool

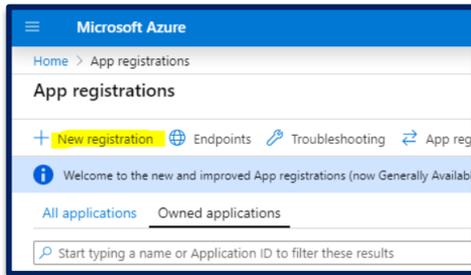
Success! Your application has been registered.

Copy these fields. You'll need them later

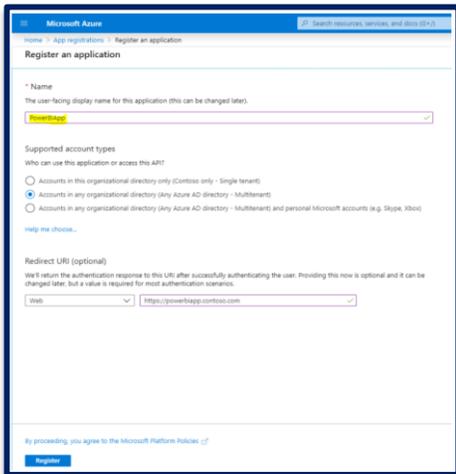
Application ID: ea59eab2

Application secret: 74XOw

- Open [https://portal.azure.com/#blade/Microsoft\\_AAD\\_RegisteredApps/ApplicationsListBlade/quickStartType//sourceType/](https://portal.azure.com/#blade/Microsoft_AAD_RegisteredApps/ApplicationsListBlade/quickStartType//sourceType/) and select New Registration:



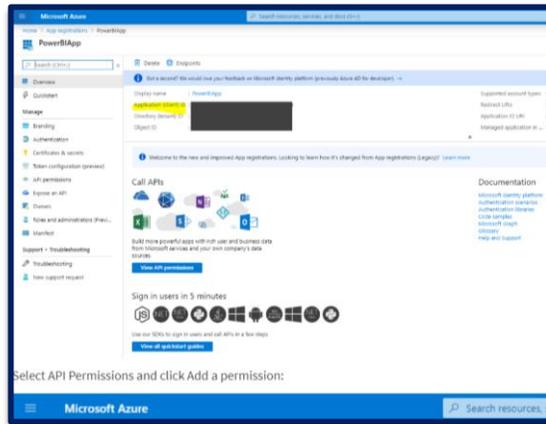
- Provide a name, e.g., PowerBIApp, and the supported account types, e.g., Single tenant, Multitenant etc, and click Register:



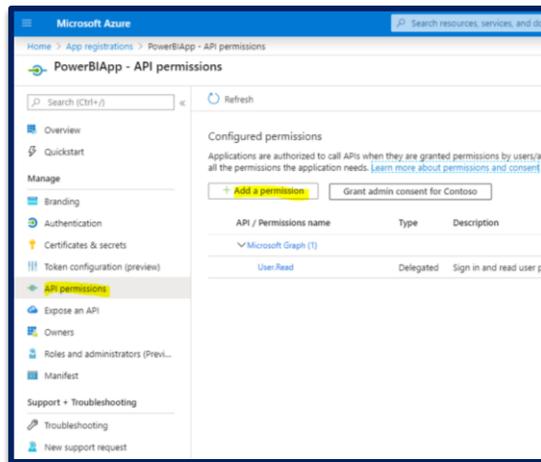
- Note the Application Client Id to use when integrat

We Automate Your Power BI Dataset Refresh with Python.

**Cognitive Convergence**  
<http://www.cognitiveconvergence.com>  
 +1 4242530744  
[shahzad@cognitiveconvergence.com](mailto:shahzad@cognitiveconvergence.com)



- Select API Permissions and click Add a permission:



- Select Power BI Service:

**Request API permissions**

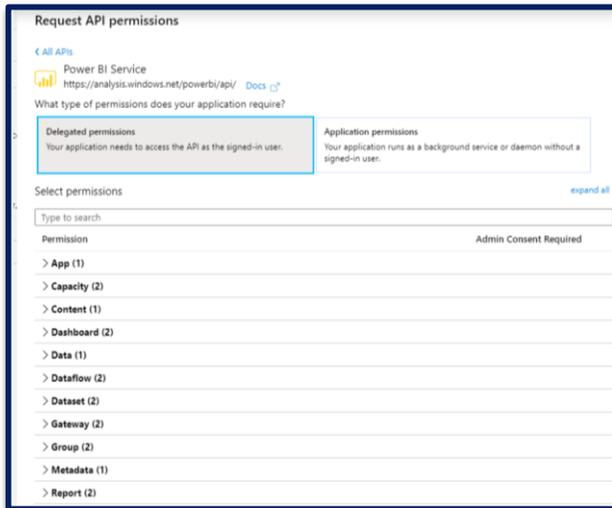
Select an API

---

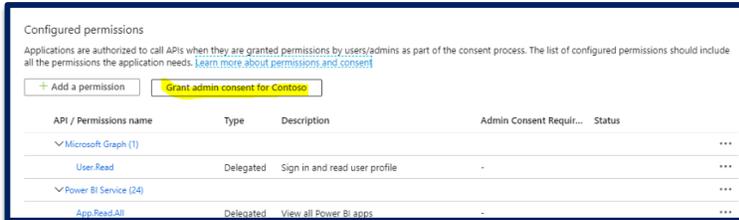
Commonly used Microsoft APIs

 <b>Microsoft Graph</b> Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more, through a single endpoint.		
 <b>Azure Rights Management Services</b> Allow validated users to read and write protected content	 <b>Azure Service Management</b> Programmatic access to much of the functionality available through the Azure portal	 <b>Data Export to Microsoft</b> Export data from CRM organization to a destination
 <b>Dynamics 365 Business Central</b> Programmatic access to data and functionality in Dynamics 365 Business Central	 <b>Dynamics CRM</b> Access the capabilities of CRM business software and ERP systems	 <b>Flow Service</b> Embed flow tasks
 <b>Intune</b> Programmatic access to Intune data	 <b>Office 365 Management APIs</b> Retrieve information about user, admin, system, and policy actions and events from Office 365 and Azure AD activity	 <b>OneNote</b> Create and manage files, and more
 <b>Power BI Service</b> Programmatic access to Dashboard resources such as Datasets, Tables, and Rows in Power BI	 <b>SharePoint</b> Interact remotely with SharePoint data	 <b>Skype for Business</b> Integrate real-time messaging, call capabilities

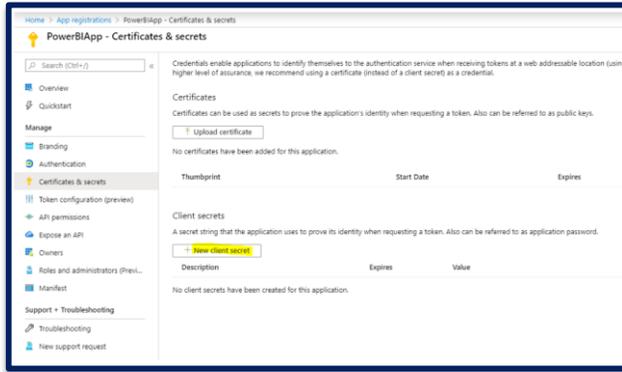
- I will grant access to several pieces of Power BI API functionality, and click Add Permissions:



- Next, click on Grant admin consent for org (or may run into the error "The user or administrator has not consented to use the application"):



- Once complete, go to Certificates and Secrets. Create a new secret that use when using the API:



- Give the secret a name and duration:

That's it, set up to integrate to the Power BI [API](#).

**Commented [A1]:** I am confuse about it is it okay or not? Should we tell them this or not

I am confuse about it is it okay or not? Should we tell them this or not

**Cognitive Convergence**  
<http://www.cognitiveconvergence.com>  
 +1 4242530744  
[shahzad@cognitiveconvergence.com](mailto:shahzad@cognitiveconvergence.com)

## POWER BI REST API

---

Power BI API require to integrate Python with Power BI. These Rest API's are

Operation group	Description
Admin	Operations for working with administrative tasks.
Apps	Operations for working with Apps.
Available Features	Operations that return available features.
Capacities	Operations for working with capacities.
Dashboards	Operations for working with dashboards.
Dataflow Storage Accounts	Operations for working with dataflow storage accounts.
Dataflows	Operations for working with dataflows.
Datasets	Operations for working with datasets.
Embed Token	Operations for working with embed tokens.
Gateways	Operations for working with gateways.
Groups	Operations for working with groups.
Imports	Operations for working with imports.
Push Datasets	Operations for working with push datasets.
Reports	Operations for working with reports.
Users	Operations for working with users.

## PYTHON PACKAGES TO INTEGRATE POWER BI REST API

---

### Pypowerbi

A python library for Microsoft's Power BI.

**Installation command:**

```
pip install pypowerbi
```

### ADAL Library

The ADAL for Python library enables python applications to authenticate with Azure AD and get tokens to access Azure AD protected web resources.

**Installation command**

```
pip install adal
```

## GENERATING AN ACCESS TOKEN

First thing first, need to generate an access token, that will be used to authenticate further communication with the API. For access token import following

- **import adal:** to get access token
- **import json:** to print token
- **from pypowerbi.client import PowerBIClient:** to get access of Power BI Client.
- **username='User\_NAME'**
- **password='PASSWORD'**
- **client\_id="client ID of Azure Registered Application.**

```
import adal
import json
from pypowerbi.dataset import Column, Table, Dataset
from pypowerbi.client import PowerBIClient
import requests

authority_url = 'https://login.microsoftonline.com/...'
resource_url = 'https://analysis.windows.net/powerbi/api'
api_url = 'https://api.powerbi.com'
GROUP_ID = '...'
REPORT_ID = '...'
username = '...'
password = '...'
client_id = '...'

context = adal.AuthenticationContext(
    authority_url,
    validate_authority=True,
    api_version=None
)

token = context.acquire_token_with_username_password(
    resource='https://analysis.windows.net/powerbi/api',
    username='...',
    password='...',
    client_id='...'
)

access_token = token['accessToken']

print('Here is the token:')
print(json.dumps(token, indent=2))
```

It returns access token



## List of Reports

In order to get list of reports available in workspace Get method is used along with the API and authorization.

**ENDPOINT:** [https://api.powerbi.com/v1.0/myorg/groups/{group\\_id}/reports](https://api.powerbi.com/v1.0/myorg/groups/{group_id}/reports)

Replace group\_id with the workspace id, whose reports are required. Gets the list of reports available in a workspace.

**Method:** GET

**Headers:**

**Authorization:** Bearer <token you saved before>

**Content-Type:** application/json; charset=utf-8

It should return the list of reports available in the workspace.



## DEMO OF LIST OF GROUPS

Save the value `access_token` from the previous call. In order to get list of workspaces in Power BI use following

- from `pypowerbi.dataset` import `Column`, `Table`, `Dataset`
- **Rest API:** <https://api.powerbi.com/v1.0/myorg/groups>
- Client-id, username and password.

```
print('List of Groups in my organization')

def get_groups(client_id, username, password):
    endpoint = "https://api.powerbi.com/v1.0/myorg/groups"
    headers = make_headers(client_id, username, password)
    return requests.get(endpoint, headers=headers).json()

get_groups(client_id, username, password)
```

It will return the list of workspaces.

```
List of Groups in my organization
{'@odata.context': 'http://wabi-us-east2-b-primary-redirect.analysis.windows.net/v1.0/myorg/$metadata#groups',
 '@odata.count': 8,
 'value': [{'id': 'f7d9abc3-9dfa-4eff-99c7-1995b1cc5bf7',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'All Company'},
          {'id': '2f8a34f7-2c70-44c0-b7e6-83e6c6bdc663',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Cognitive Convergence'},
          {'id': 'c07c4d8a-ce7b-41c2-b8fc-41abfd1701d6',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Blue Team'},
          {'id': 'b3e5e69d-89fc-4ae8-869c-986bbf0010f4',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Contoso Inc'},
          {'id': 'b920f778-c9be-48ab-aebd-f5caee582fe2',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'QuickBooks Online'},
          {'id': '33a2dd40-5973-42e7-8379-45779e8e66f2',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Test'},
          {'id': '88930a2d-1f6d-479e-845e-b53e8dd57b1b',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Titanic'},
          {'id': '1ba17cea-2950-458e-ba40-a19ac57ecaa1',
            'isReadOnly': False,
            'isOnDedicatedCapacity': False,
            'name': 'Kaban DashBoard'}]}}
```

## CONCLUSION

---

This paper will discuss about Python and its integration with power BI rest API and Power BI Desktop. Their development patterns and implementation details along with some examples are also discussed.

Cognitive Convergence will provide consulting services that help in designing, deploying, managing, enhancing, or troubleshooting on-premises, cloud-based or hybrid Power BI environment. Cognitive Convergence will provide help to start fresh with Power BI to modernize current business analytics solution or revamp existing Power BI deployment by incorporating new data sources or adding new services.



### Contact Us

#### Cognitive Convergence

<http://www.cognitiveconvergence.com>

+1 4242530744

[shahzad@cognitiveconvergence.com](mailto:shahzad@cognitiveconvergence.com)