

Lab 12. Calling the Microsoft Graph

Learning objective: create a Flow that will create a Team by using the Graph API.

Prerequisites: calling the Microsoft Graph requires a Premium connector.

Duration: 35 minutes

Scenario: you will create a button Flow that will take the Team name as a parameter. The Flow will create a group and will attach a Team to this group.

First, you need to identify the Graph API you need.

Create a group:

- Documentation: <https://docs.microsoft.com/en-us/graph/API/group-post-groups?view=graph-rest-1.0&tabs=http>
- API: <https://graph.microsoft.com/v1.0/groups>
- permissions: Group.ReadWrite.All, Directory.ReadWrite.All

Attach a team to a group:

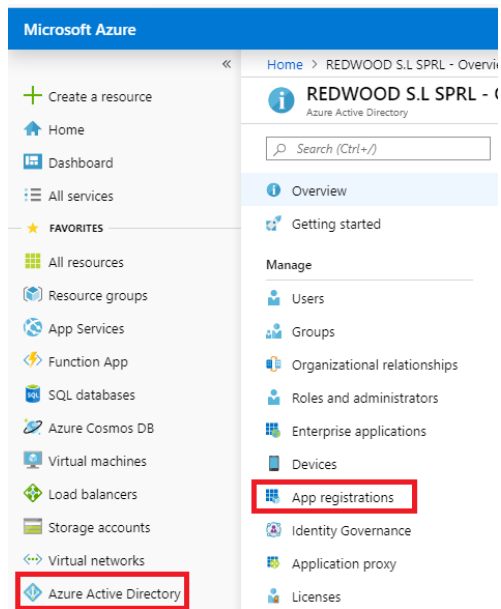
- Documentation: <https://docs.microsoft.com/en-us/graph/API/team-put-teams?view=graph-rest-1.0&tabs=http>
- API: <https://graph.microsoft.com/v1.0/groups/{id}/team>
- permissions: Group.ReadWrite.All

Add a group owner

- Documentation: <https://docs.microsoft.com/en-us/graph/API/group-post-owners?view=graph-rest-1.0&tabs=http>

Tasks:

1. Go to **Azure.com** portal, go to **Azure Active Directory**; click **Apps registrations**:



2. Click **New registration**:

[+ New registration](#) [🌐 Endpoints](#)

[i](#) Welcome to the new and improved A

[⚠](#) Looking to learn how it's changed
Still want to use App registrations

[All applications](#) [Owned applicati](#)

3. Name the application **FlowCreateTeam**:

Commented [A1]:

Commented [A2]: Pas complet

[Home](#) > [REDWOOD S.L SPRL - App registrations](#) > Register an application

Register an application

* Name

The user-facing display name for this application (this can be changed later).

FlowCreateTeam

Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory only (REDWOOD S.L SPRL only - Single tenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

[Help me choose...](#)

- Click **Register**.
- Now we will assign some permissions to this application.

Click **API permissions**:

[Home](#) > [REDWOOD S.L SPRL - App registrations](#) > FlowCreateTeam - API permissions

Search (Ctrl+ /)

Overview

Quickstart

Manage

Branding

Authentication

Certificates & secrets

API permissions

Expose an API

Owners

Roles and administrators (Previous)

Manifest

Support + Troubleshooting

Troubleshooting

New support request

API permissions

Application permissions

These are the permissions that your application can use to access the API.

Grant permissions

- Click **Add Permission**
- Select **Microsoft APIs**, click **Microsoft Graph**:

Request API permissions

Select an API

Microsoft APIs APls my organization uses My APIs

Commonly used Microsoft APIs

Microsoft Graph

Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

8. Choose **application permissions**.
9. In the Groups, select **Group.ReadWrite.All**:

▼ Group (1)

<input type="checkbox"/>	Group.Read.All Read all groups ⓘ	Yes
<input checked="" type="checkbox"/>	Group.ReadWrite.All Read and write all groups ⓘ	Yes

10. Click **Add permissions**
11. Follow the same steps to add the Directory permissions:

▼ Directory (1)

<input type="checkbox"/>	Directory.Read.All Read directory data ⓘ	Yes
<input checked="" type="checkbox"/>	Directory.ReadWrite.All Read and write directory data ⓘ	Yes

12. Click **Add permissions**
13. Click **Grant Consent for** <your tenant name>

API permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as all the permissions the application needs.

[+ Add a permission](#)

API / PERMISSIONS NAME	TYPE	DESCRIPTION
▼ Microsoft Graph (3)		
Directory.ReadWrite.All	Application	Read and write directory data
Group.ReadWrite.All	Application	Read and write all groups
User.Read	Delegated	Sign in and read user profile

These are the permissions that this application requests statically. You may also request user callable permissions dynamically through code. [See best practices for requesting permissions](#)

Grant consent

As an administrator, you can grant consent on behalf of all users in this directory. Granting admin consent will not be shown a consent screen when using the application.

[Grant admin consent for \[REDACTED\]](#)

14. Confirm: yes.

✓ Successfully granted admin consent for the requested permissions.

API permissions

Applications are authorized to call APIs when they are granted permissions by users as all the permissions the application needs.

[+ Add a permission](#)

API / PERMISSIONS NAME	TYPE	DESCRIPTION
▼ Microsoft Graph (3)		
Directory.ReadWrite.All	Application	Read and write data
Group.ReadWrite.All	Application	Read and write all
User.Read	Delegated	Sign in and read u

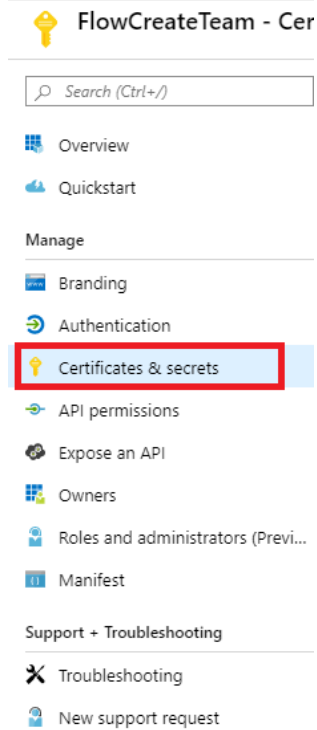
These are the permissions that this application requests statically. You may also request user callable permissions dynamically through code. [See best practices for requesting permissions](#)

Grant consent

As an administrator, you can grant consent on behalf of all users in this directory. Granting admin consent will not be shown a consent screen when using the application.

[Grant admin consent for REDWOOD S.L SPRL](#)

15. Go to the application panel and click **Certificates & secrets:**



Commented [A3]: C'est ou ce panel ????

16. Click **New client secret**.

17. Fill in the secret form and select **Expires: Never**

Add a client secret

Description

FlowcreateTeamsecret

Expires

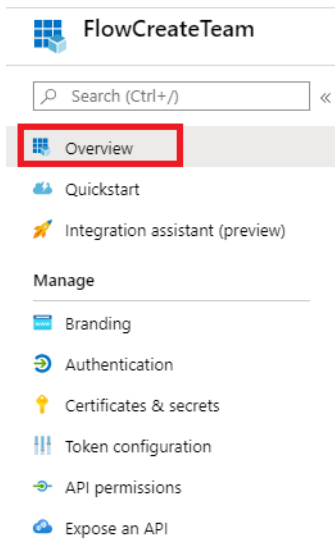
☐ In 1 year

☐ In 2 years

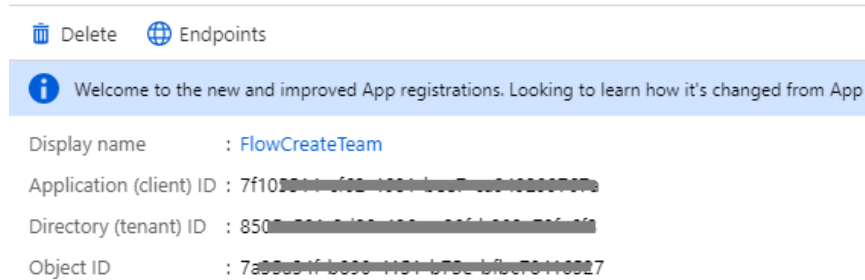
☒ Never

Add Cancel

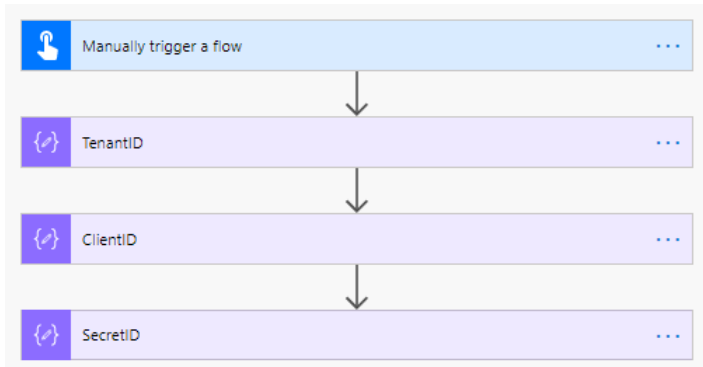
18. Click **Add**.
19. Copy the secret value in notepad.
20. Go back to **Overview**:



21. Copy the application ID and the Tenant ID into Notepad:



22. Create a Flow starting from a button, create 3 Compose actions Named **TenantID**, **ClientID**, and **SecretID** and store your values in these actions:



23. In the trigger define an input named **group name**:

The screenshot shows the configuration interface for the 'Manually trigger a flow' trigger. It features a text input field labeled 'group name' and a placeholder text 'Please enter your input'. Below the input field is a button labeled '+ Add an input'. The interface also includes an information icon (i) and a menu icon (three dots) in the top right corner.

24. Add an HTTP (premium action) and define its properties like **this**:

Commented [A4]: Pas sympa le screenshot a tout devoir recopier bof

The screenshot shows an HTTP client interface with the following fields:

- Method:** POST
- URI:** `http://graph.microsoft.com/v1.0/groups`
- Headers:** Two input fields labeled "Enter key" and "Enter value".
- Queries:** Two input fields labeled "Enter key" and "Enter value".
- Body:** A text area containing a JSON payload:


```
{
    "description": "group name",
    "displayName": "group name",
    "groupTypes": [
      "Unified"
    ],
    "mailEnabled": true,
    "mailNickname": "group name",
    "securityEnabled": false,
    "visibility": "Public"
  }
```
- Cookie:** One input field labeled "Enter HTTP cookie".
- Show advanced options:** A link with a downward arrow.

You can adapt the Body by reusing the following code:

```
{
  "description": "",
  "displayName": "",
  "groupTypes": [
    "Unified"
  ],
  "mailEnabled": true,
  "mailNickname": "",
  "securityEnabled": false,
  "visibility": "Public"
}
```

25. Click **Show Advanced options**.
26. For the **Authentication** field, select **Active Directory OAuth**.
27. Pass the TenantID, ClientID and SecretID
28. **Authority** should be: <https://login.microsoftonline.com>

29. **Audience** should be: <https://graph.microsoft.com>

30. Define the body as follows:

The screenshot shows the configuration for an HTTP action. The Method is POST and the URI is https://graph.microsoft.com/v1.0/groups. The Body is a JSON object with the following properties: description (group name), displayName (group name), groupTypes (Unified), mailEnabled (true), mailNickname (group name), securityEnabled (false), and visibility (Public). The Authentication is Active Directory OAuth, Authority is https://login.microsoftonline.com, Tenant is Outputs, Audience is https://graph.microsoft.com, Client ID is Outputs, Credential Type is Secret, and Secret is Outputs. A link to 'Hide advanced options' is at the bottom.

```
{
  "description": "group name",
  "displayName": "group name",
  "groupTypes": [
    "Unified"
  ],
  "mailEnabled": true,
  "mailNickname": "group name",
  "securityEnabled": false,
  "visibility": "Public"
}
```

31. Rename the HTTP action to **Create Group**.

32. Save the Flow, run it, pass a group name (add your name in the group name to make sure it is unique)

33. Check the Azure Active Directory; your new group should have been created:




Home > REDWOOD S.L SPRL > Groups - All groups

Groups - All groups

REDWOOD S.L SPRL - Azure Active Directory

« + New group Delete Refresh Columns Got feedback

Search groups x Add filters

NAME	OBJECT ID	GROUP TYPE
 Demoguests	d1ee0047-a473-46cf-a1c2-4...	Office
 groupx	f6024094-376e-4877-a7c2-...	Office
 teamcollab	13341317-fa3f-401d-9f80-1d...	Office

Settings

- General
- Expiration
- Naming policy

34. Add Compose action, rename it **GroupID** and grab the id value returned by the Create Group action:

HTTP - Create Group

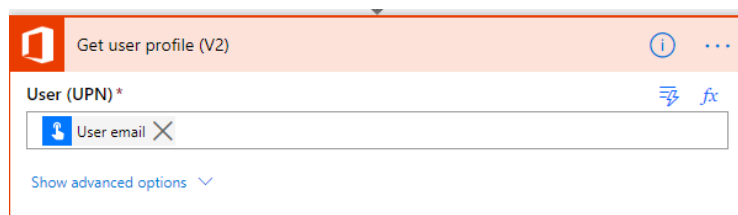
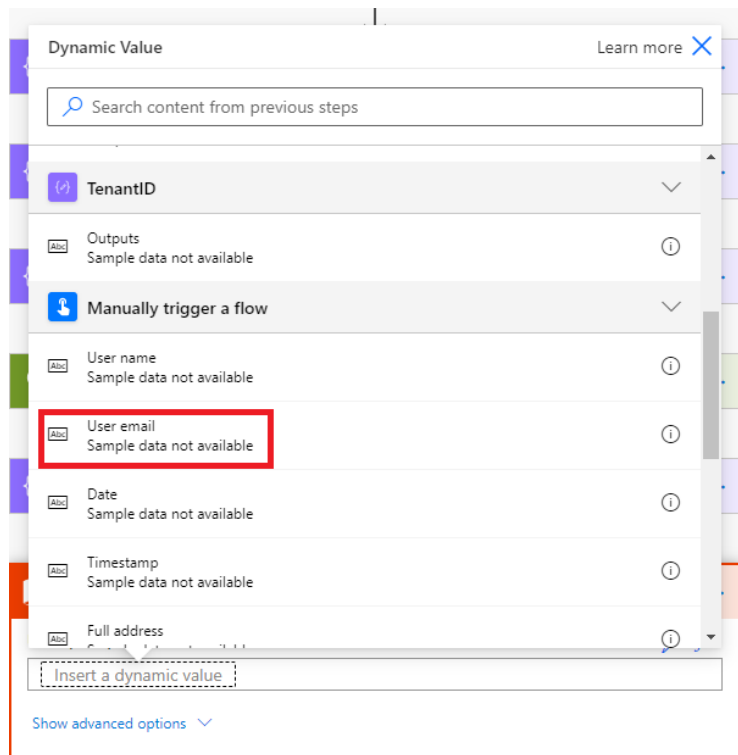
GroupID

Inputs *

body(...)

body('HTTP_-_Create_Group')['id']

35. A group must have an owner; the owner will be the user starting the Flow. We must find his e-mail and its object id in AAD.
36. Add an action Get User Profile (V2) and define the expression to retrieve the current user e-mail address:




37. Add a Compose action, name it **OwnerURI** and use the following expression:

```
concat(concat('https://graph.microsoft.com/v1.0/groups/', outputs('GroupID')), '/owners/$ref')
```





38. Add an **HTTP action** to add the owner to the group; in the body, pass the id field of the **Get user profile** action. Use the settings defined in our previous HTTP action (**secretID**, **tenantID**,...)

 HTTP - Add Owner to group ...


Method *

POST


URI *



 Outputs 



Headers

Enter key	Enter value	
-----------	-------------	---

Queries

Enter key	Enter value	
-----------	-------------	---

Body  

```
{  
  "@odata.id": "https://graph.microsoft.com/v1.0/users/ Id   
}
```

Cookie

Enter HTTP cookie



Authentication *

Active Directory OAuth

Authority

https://login.microsoftonline.com



Tenant *

 Outputs 

Audience *

https://graph.microsoft.com



Client ID *

 Outputs 

Credential Type *

Secret

Secret *


 Outputs 

[Hide advanced options](#) ^


39. Use another HTTP action to create the Team, make sure the Method is PUT and pass the previous compose action Id value into the group URL.


You can adapt the Body by reusing the following code:

```
{
  "memberSettings": {
    "allowCreateUpdateChannels": true
  },
  "messagingSettings": {
    "allowUserEditMessages": true,
    "allowUserDeleteMessages": true
  },
  "funSettings": {
    "allowGiphy": true,
    "giphyContentRating": "strict"
  }
}
```



HTTP - Create Teams








Method *

PUT


URI *

https://graph.microsoft.com/v1.0/groups/ Outputs  /team

Headers

Enter key	Enter value	
-----------	-------------	---

Queries


Enter key	Enter value	
-----------	-------------	---

Body

```
{
  "memberSettings": {
    "allowCreateUpdateChannels": true
  },
  "messagingSettings": {
    "allowUserEditMessages": true,
    "allowUserDeleteMessages": true
  },
  "funSettings": {
    "allowGiphy": true,
    "giphyContentRating": "strict"
  }
}
```

Cookie

Enter HTTP cookie

[Show advanced options](#) 

40. Run the Flow, pass a group/team name, and connect to <http://teams.microsoft.com> to check your team.

