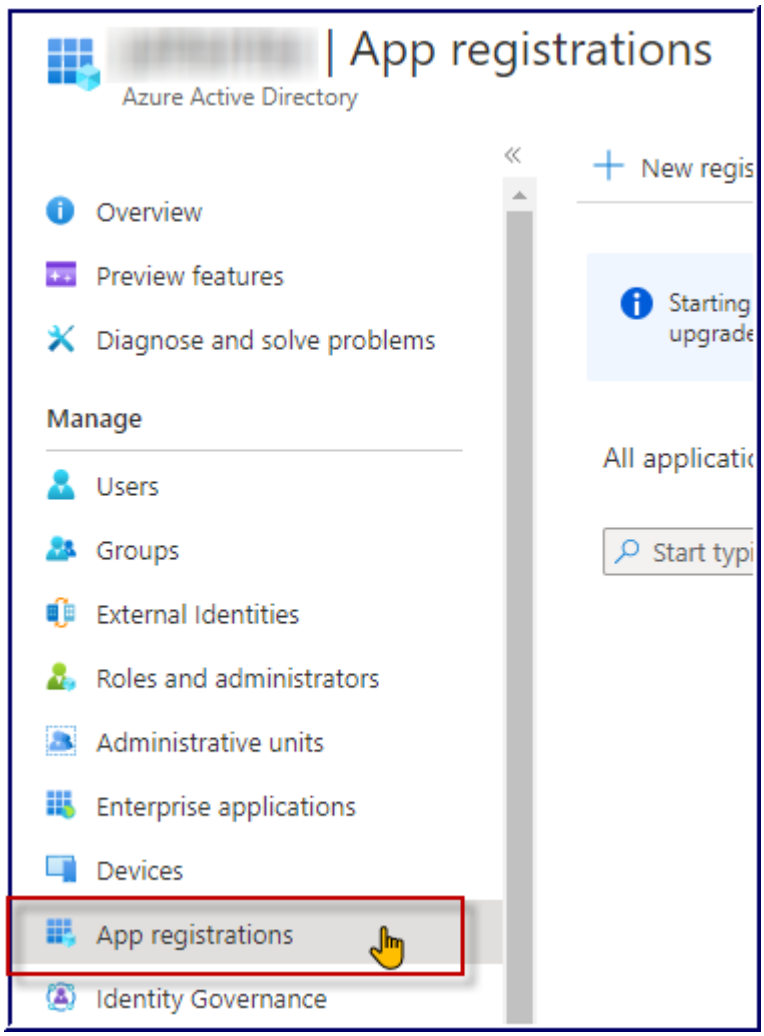
Deployment Guide

---

1. Register Azure App Registration for the back-end API

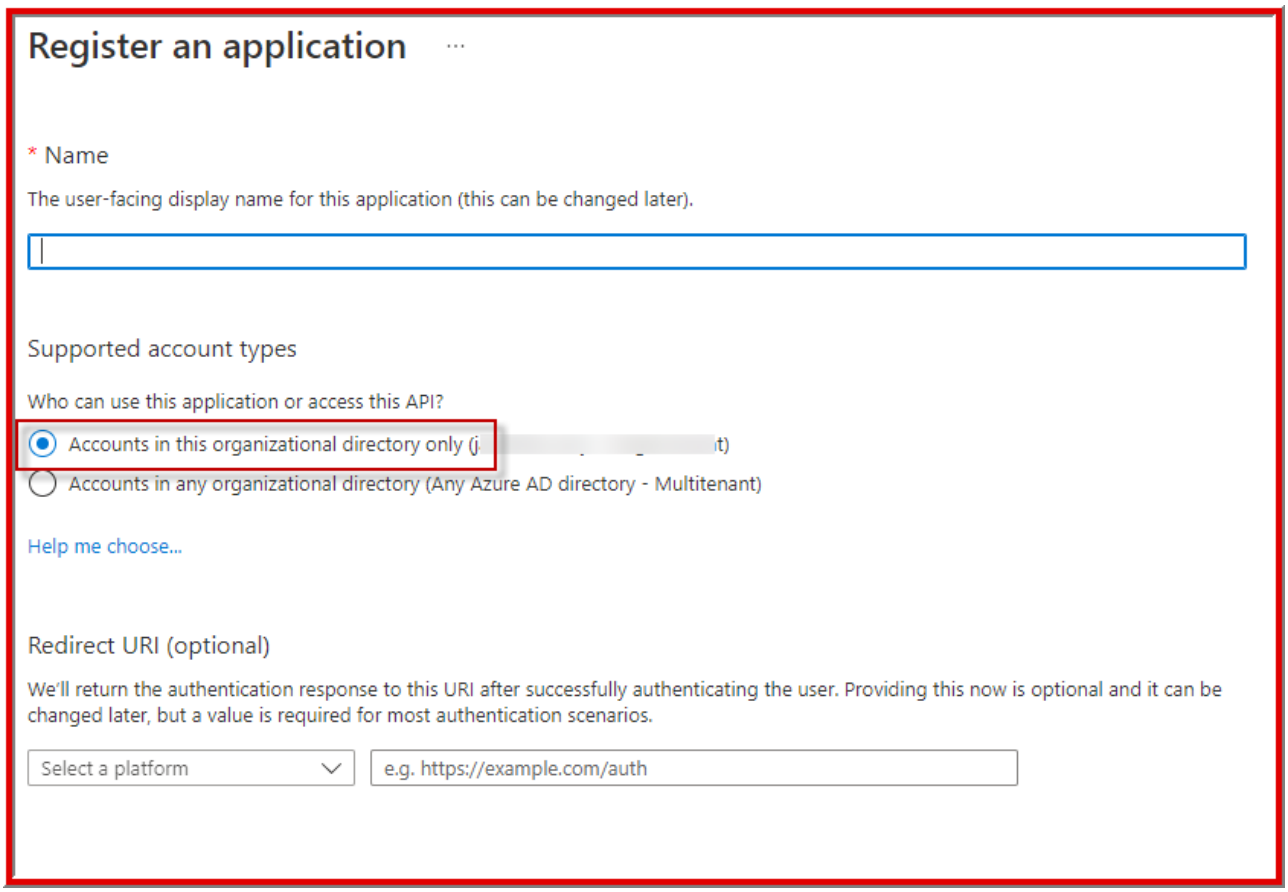
1. Log in to the Azure Portal for your subscription, and go to the App registrations blade.



2. Click **New registration** to create an Azure AD application.

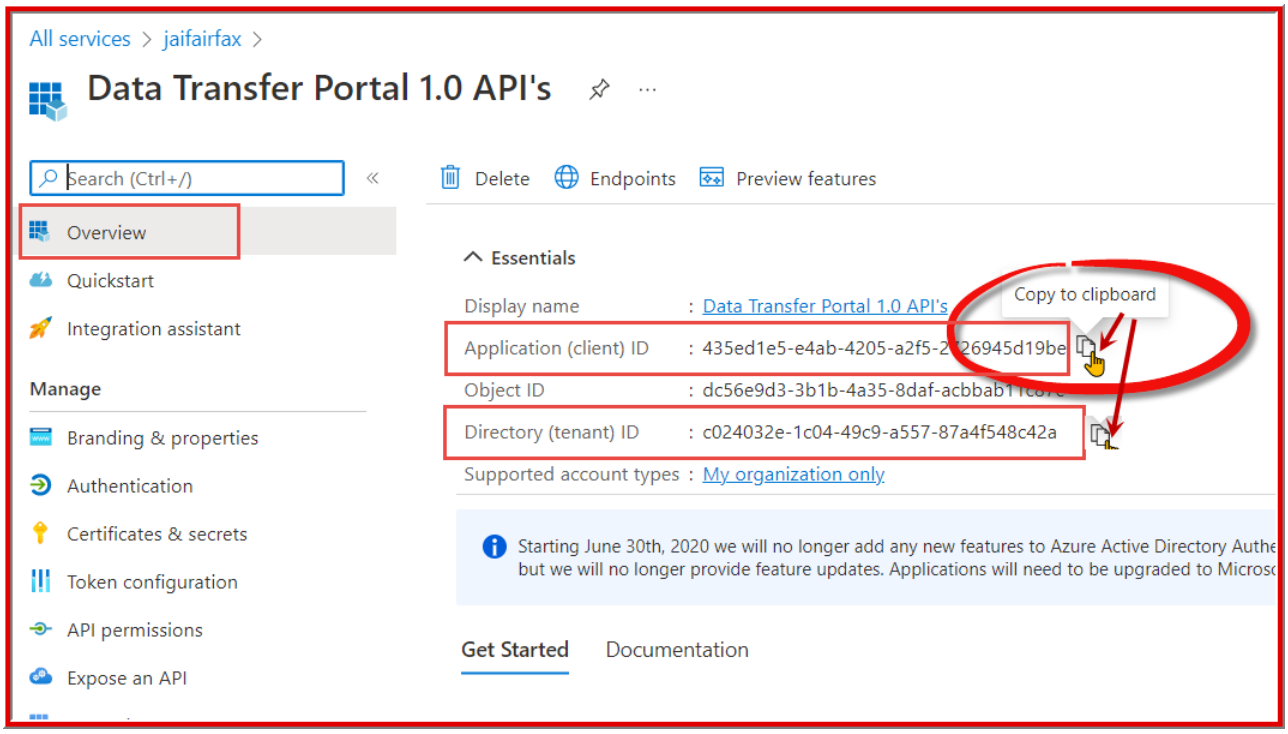
   - **Name**: Name for your App: **Data Transfer Portal API** (You can name your app registration to any other meaningful value as well)
   - **Supported account types**: Select "Accounts in this organizational directory only(Default Directory only - Single tenant)". (*refer image below*).

- Leave the "Redirect URI" field blank for now.



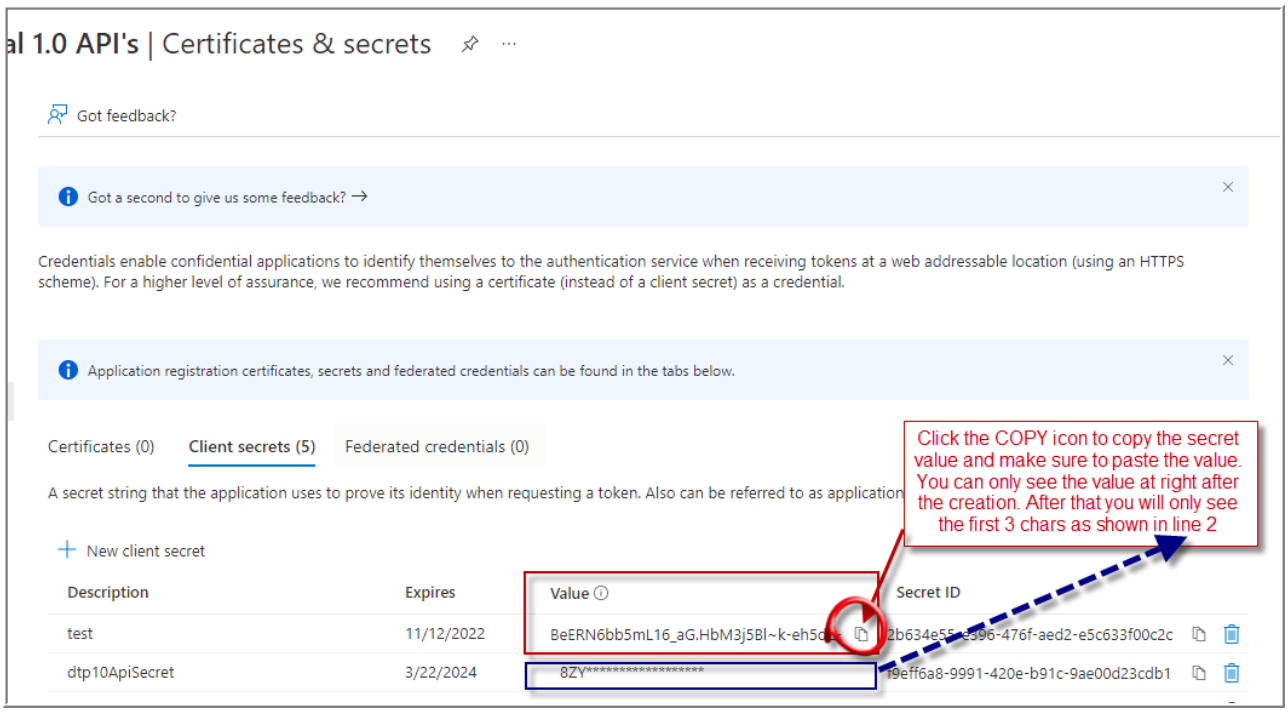
3. Click **Register** to complete the registration.

4. When the app is registered, you'll be taken to the app's "Overview" page. Copy the **Application (client) ID**; we will need it later.



5. On the side rail in the Manage section, navigate to the "Certificates & secrets" section. In the Client secrets section, click on "+ New client secret". Add a description for the secret, and choose when the

secret will expire. Click "Add".

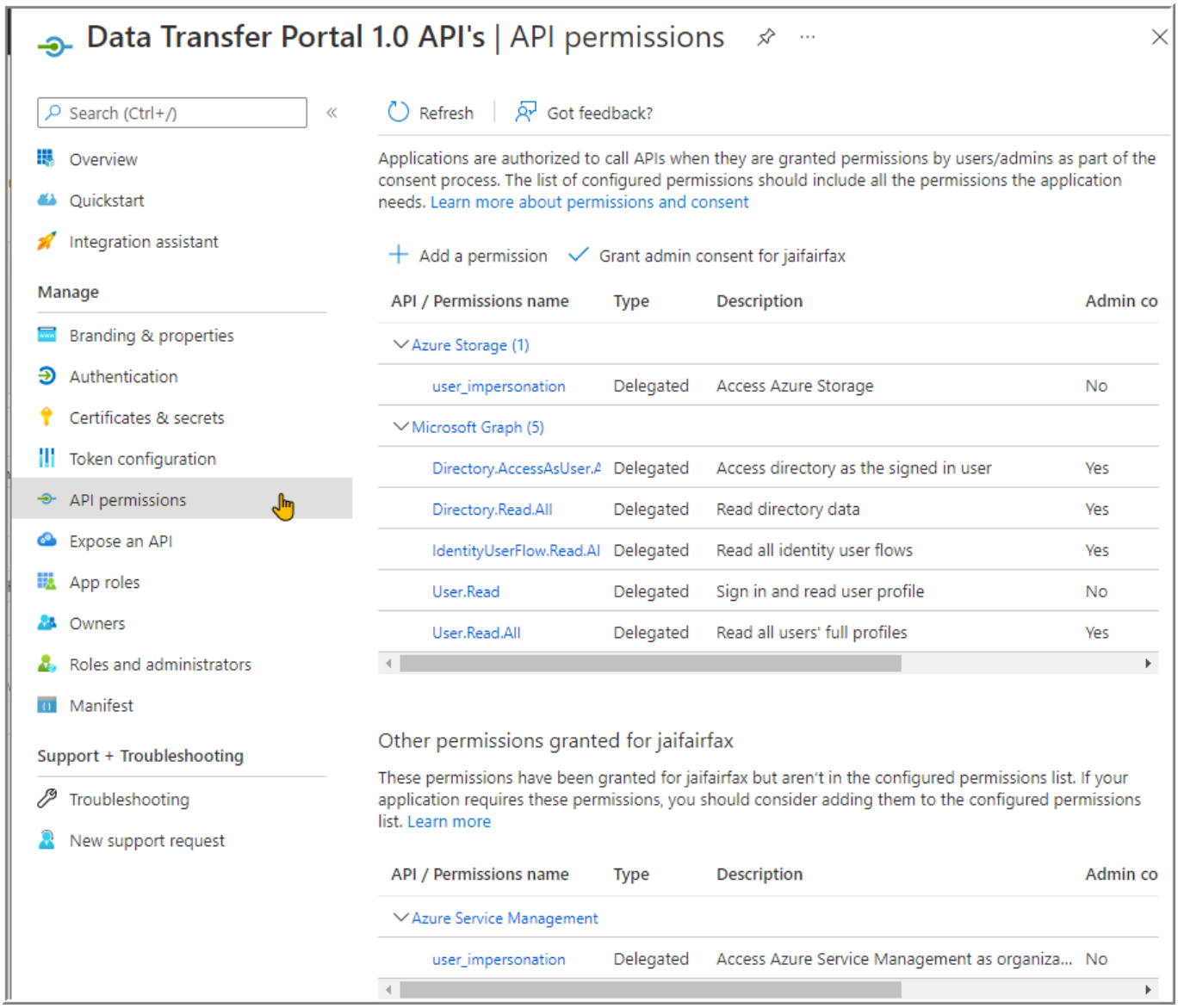


6. Once the client secret is created, copy and paste its **Value** (together with the app registration id and tenant id) in the table below; we will need it later.

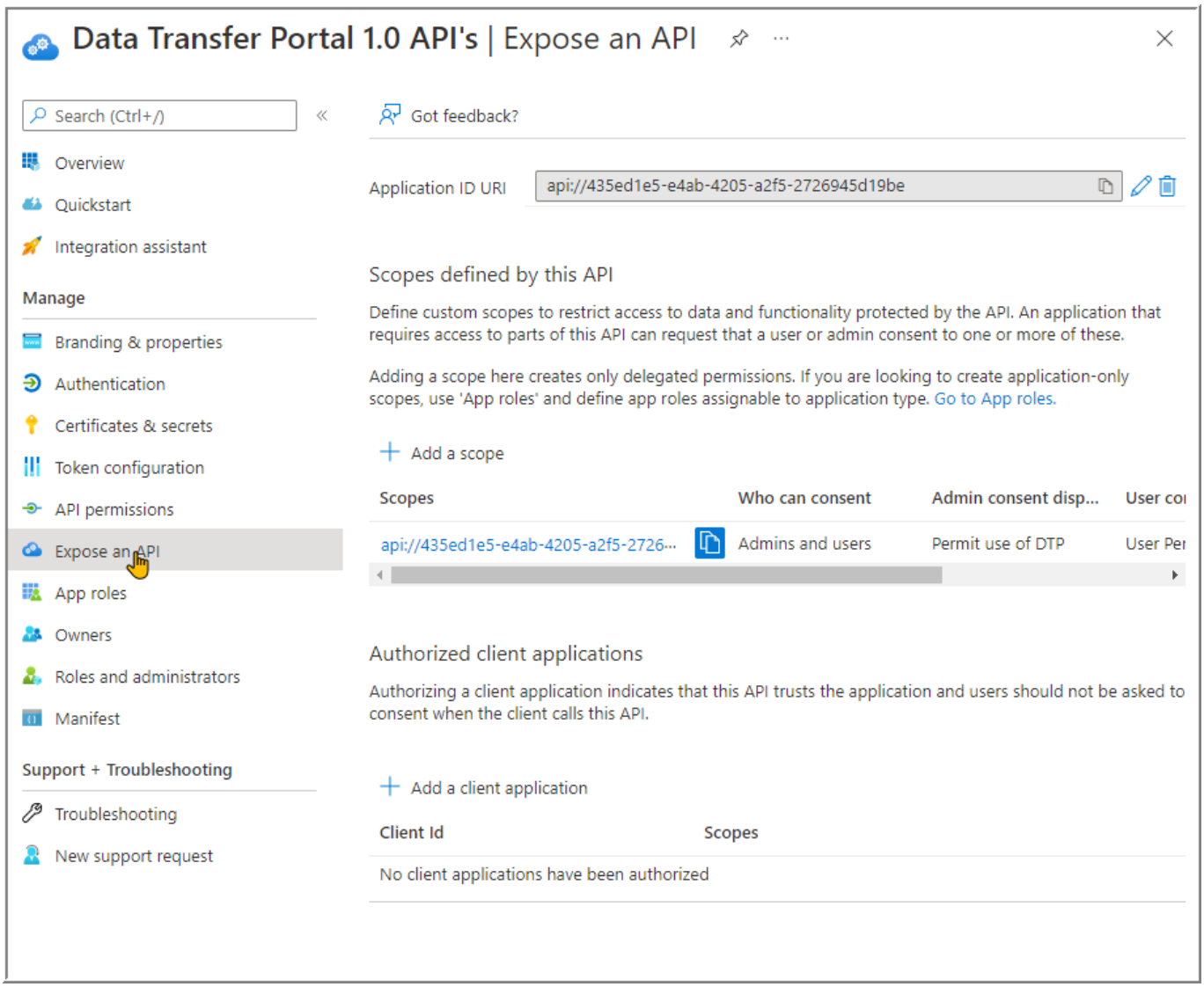
> Be aware that you can only see the secret value at the time of creation. You must copy and save the value at this time!

Data	Value
Tenant ID (Directory ID):	
API App Registration Display name:	
API App Registration ID (Client ID):	
API App Reg Secret Value:	

2. Set Up the API App Registration's API Permissions



3. Set Up the API App Regisration's scope [Expose an API]



4. Create the App Registration for the Web Client

1. Repeat steps 2-4 to create another Azure App Registration for the web site.

> Note: The WebClient does not need a secret to be created

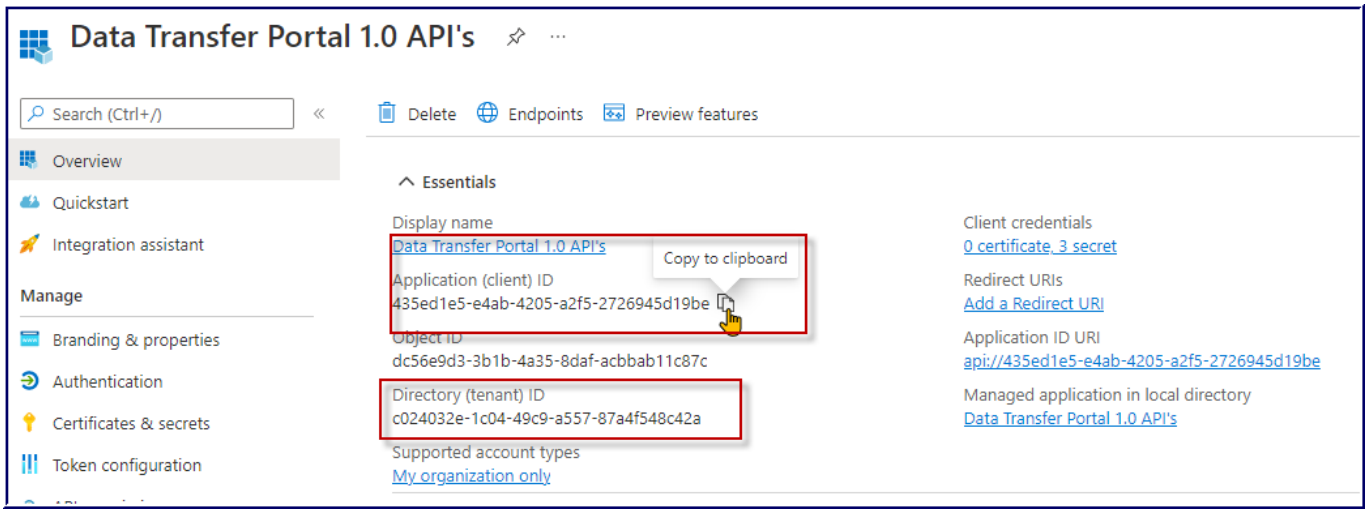
- **Name**: Name of the API - **Data Transfer Portal** (or any other name meaningful to you)
- **Supported account types**: Select "**Accounts in this organizational directory only**(Default Directory only - Single tenant)".
- Leave the "**Redirect URI**" field **blank**.

App Registrations Results

```
At this point you should have the following values:
(You can copy this table and paste it into OneNote/Word/Excel)
and populate the values for your records
```

Data	Value
Tenant ID:	

Data	Value
API App Registration Display name:	
API App Registration ID (Client ID):	
API App Reg Secret Value:	
Web Client App Registration Display Name	
Web Client App Registration ID	



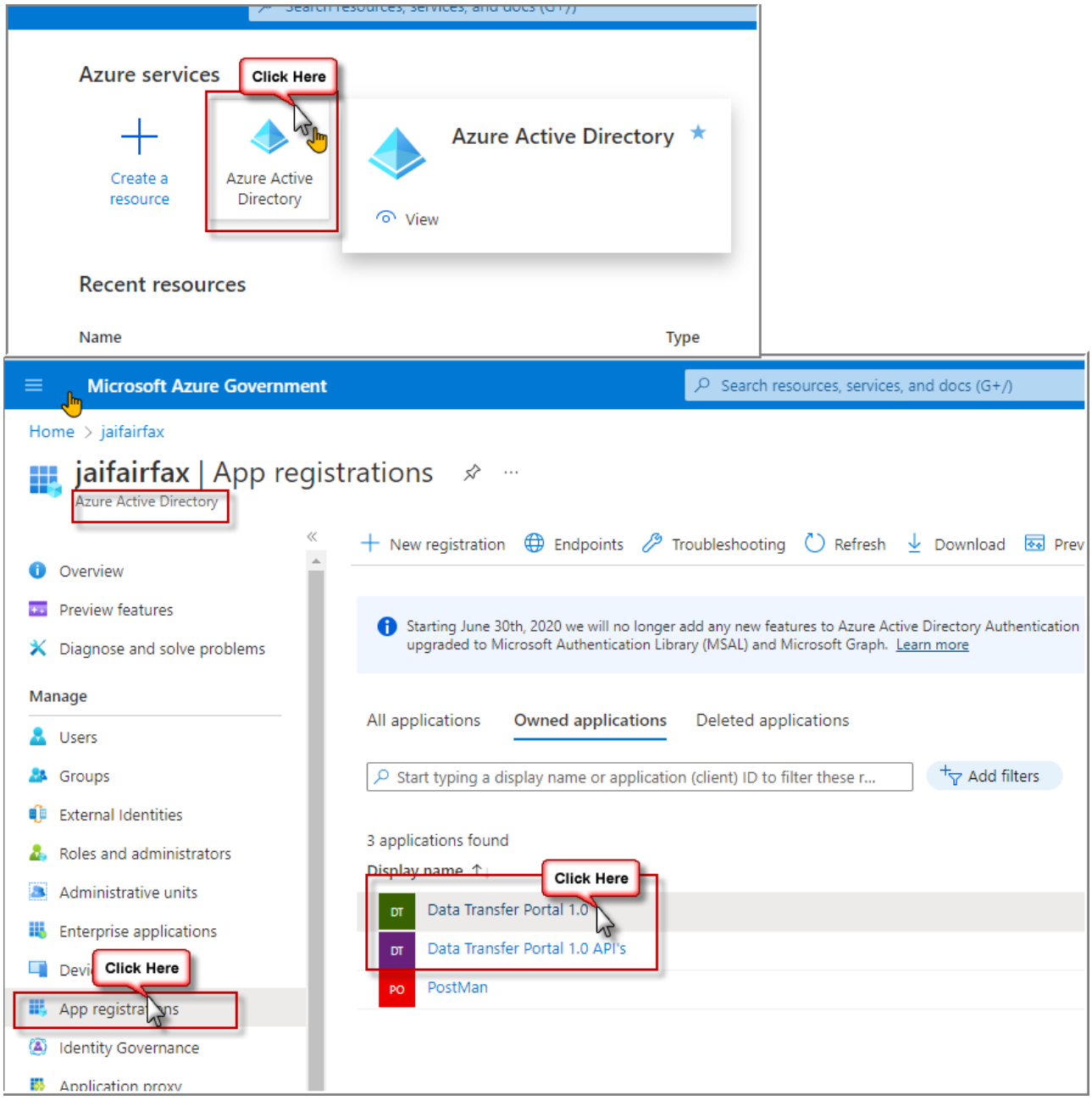
-[] Create App Registration (Service Principal) With Powershell

---

5. Set-up the WebClient's App Registration Authentication

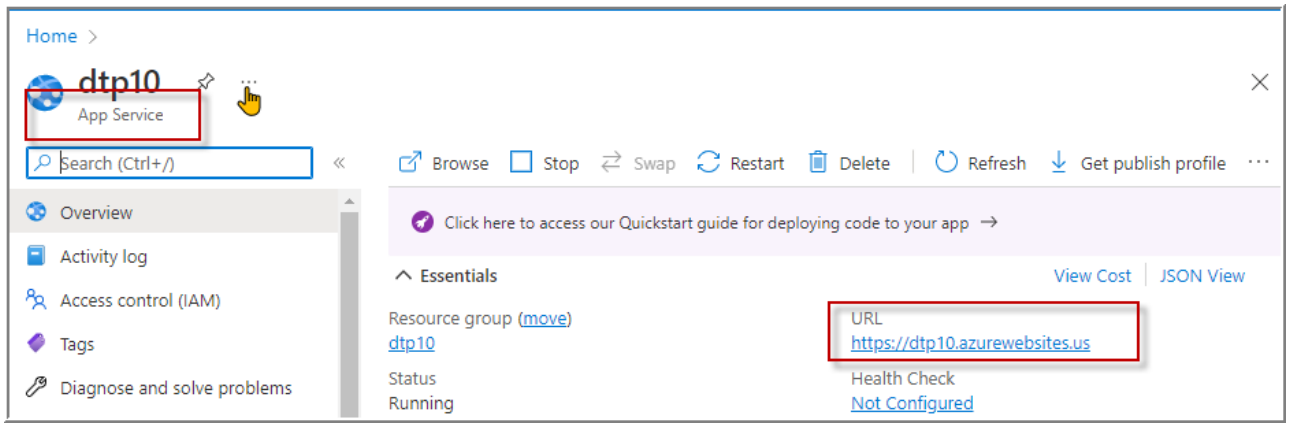
> NOTE: This step is to set-up authentication for Microsoft Graph Azure AD app. After the deployment script successfully creates the resources, navigate back to the Web Client App registration created in Step 7 to set up the authentication

1. Go to the **App Registrations** page in Azure Active Directory here and open the Microsoft Graph Azure AD app you created (in Step 1) from the application list.



2. Select the **WebClient App Registration** (as shown in the image above) Under **Manage**, click on **Authentication** to bring up authentication settings.

3. Add a new entry to **Redirect URIs**: - **Type**: Web - **Redirect URI**: Enter `https://%appDomain%/` for the URL e.g. `https://appName.azurewebsites.us/` (This is the URL from the App Service created by Bicep shown below)



4. Under **Implicit grant**, leave the **ID tokens** and **Access tokens** unchecked.

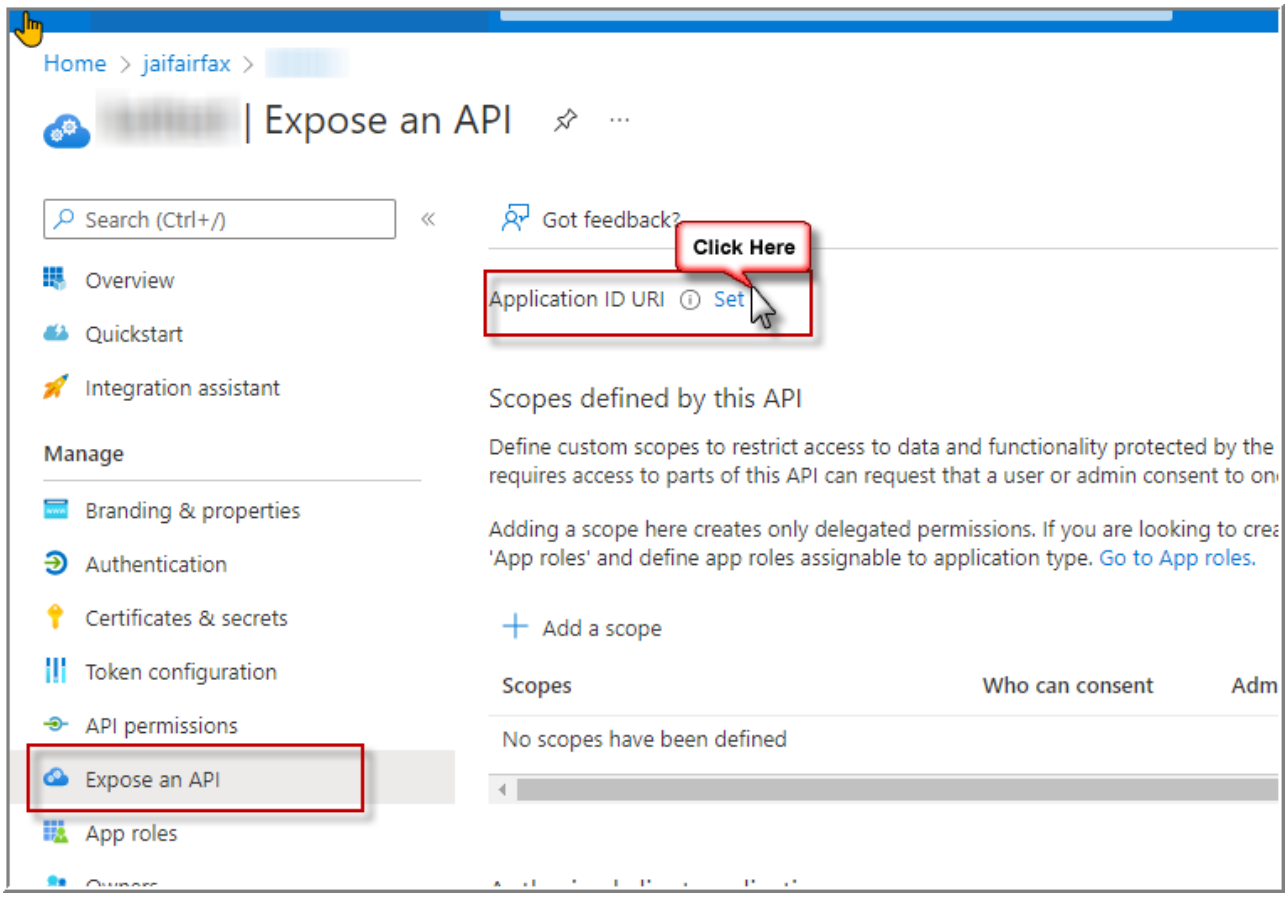
5. Under **Advanced Settings**, make sure that **Enable the following mobile and desktop flows** slider has **No** selected

6. Click **Save** to commit your changes.

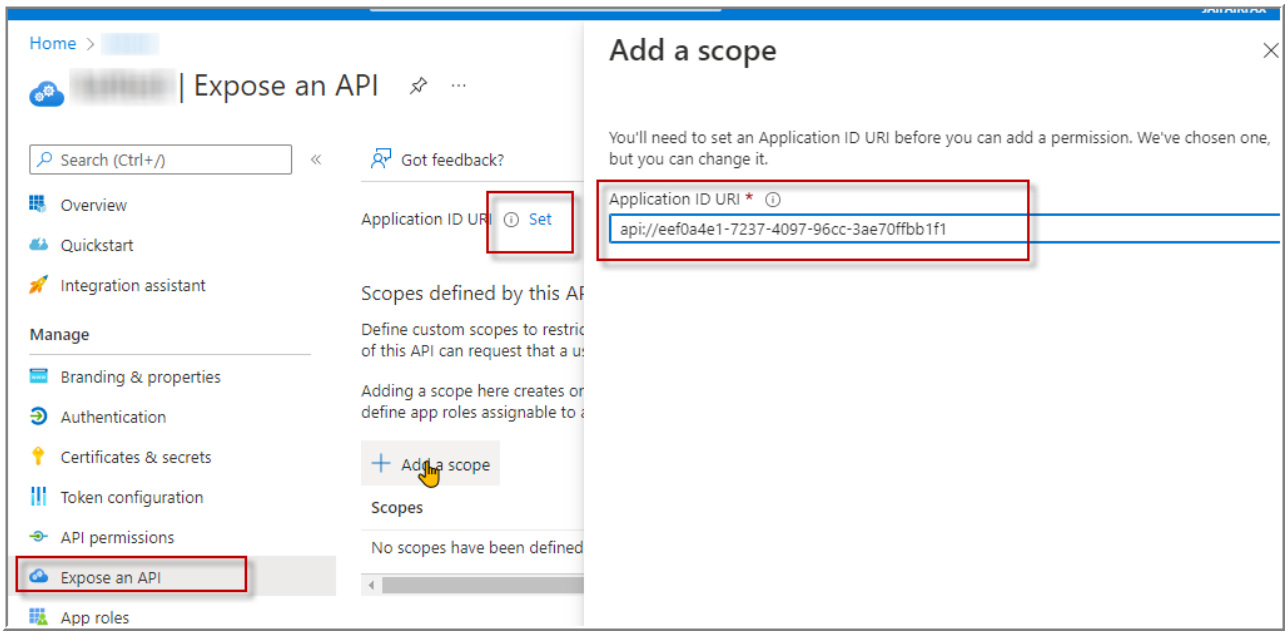
6. Set up the WebClient App's API Permissions

1. Back under **Manage**, click on **Expose an API**.

2. Click on the **Set** link next to **Application ID URI**, and change the value to `api://%appDomain%` e.g. `api://appName.azurewebsites.us`.



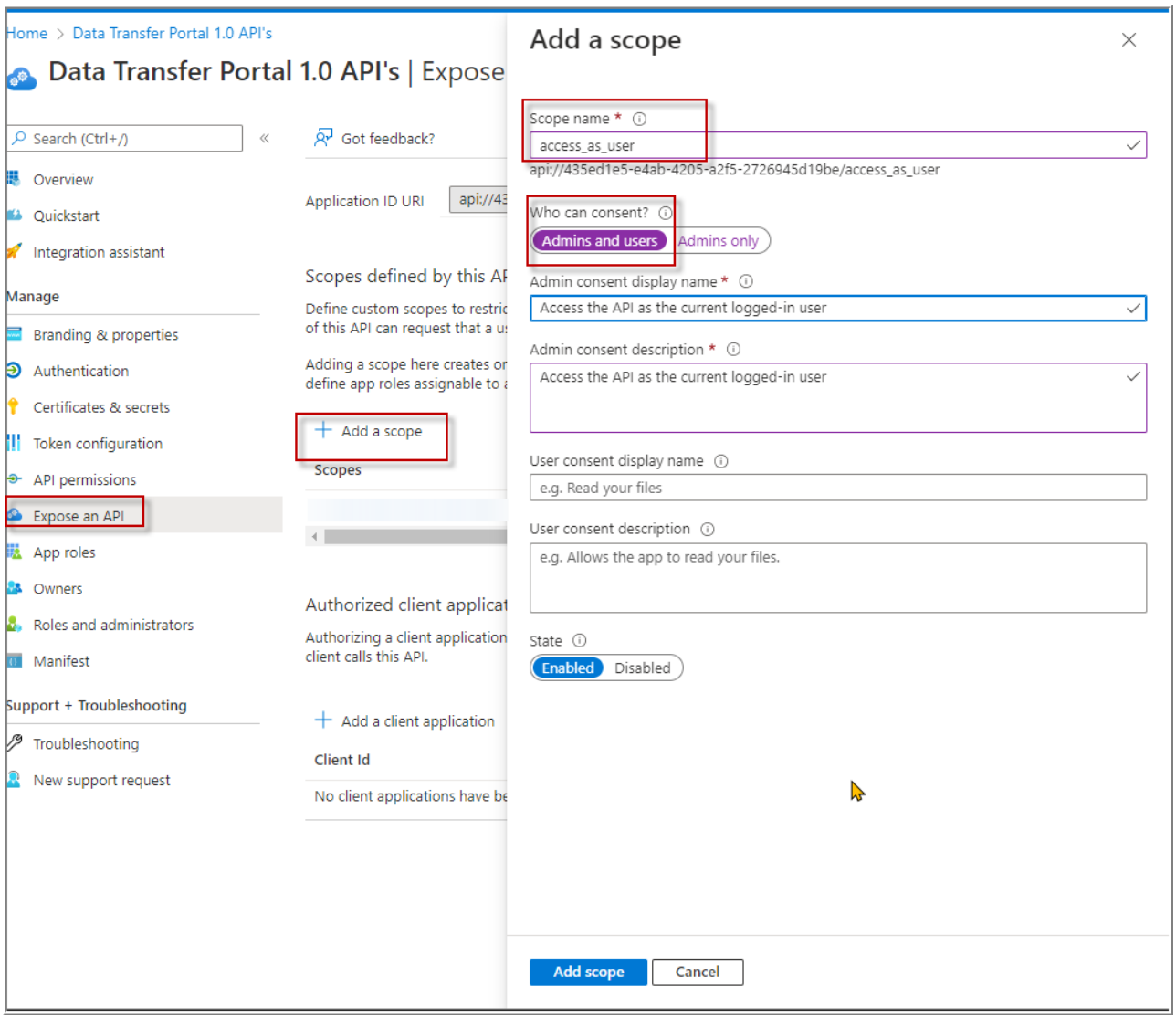
3. Click **Save** to commit your changes. You should now see the **Application ID URI** like below:



7. Add API App Registration Consent Scope

4. Under the section: **Scopes defined by this API**, Click on **Add a scope**, under **Scopes defined by this API**. In the flyout that appears, enter the following values:

Property	Value
Scope name:	**access_as_user**
Who can consent?:	**Admins and users**
Admin and user consent display name:	**Access the API as the current logged-in user**
Admin and user consent description:	**Access the API as the current logged-in user**



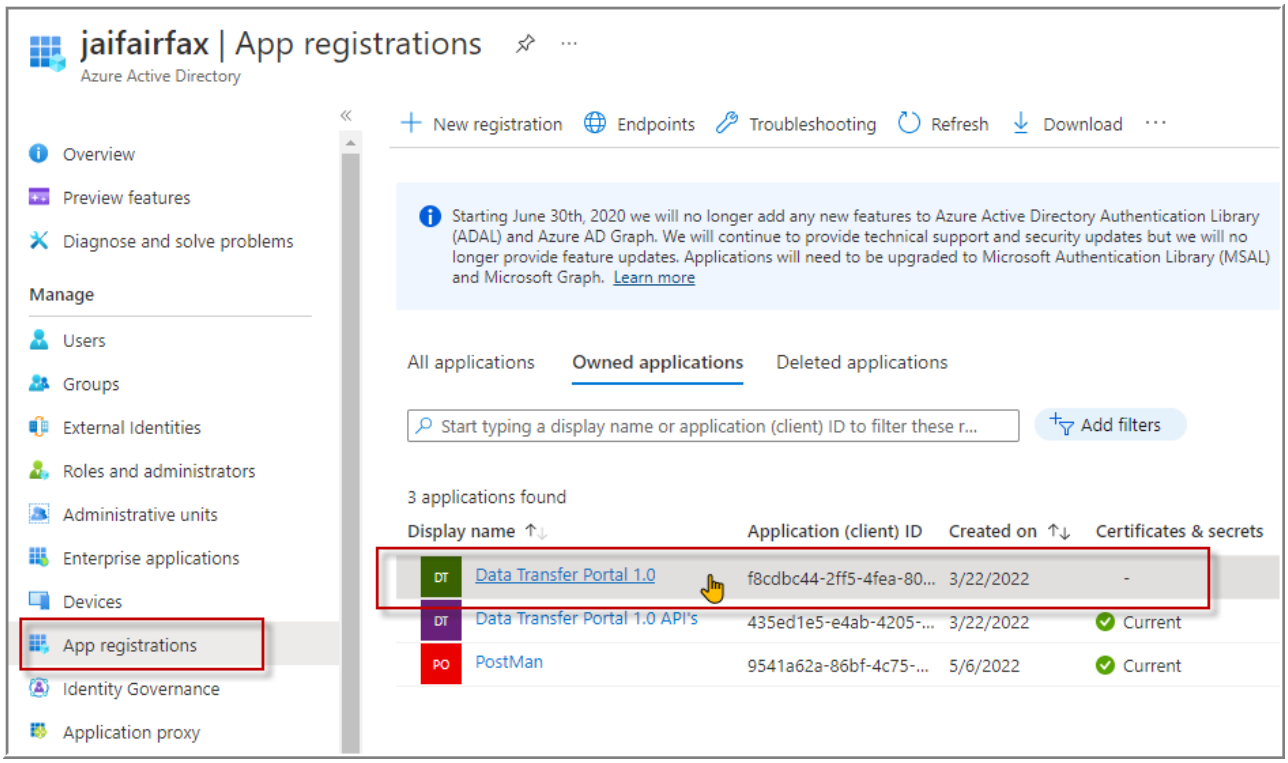
5. While still in the **Expose and API** blade, click **Add scope** then click on the **Save and continue** to commit your changes.

6. Click **Add a client application**, under **Authorized client applications**. In the flyout that appears, enter the following values: - **Client ID**: `5e3ce6c0-2b1f-4285-8d4b-75ee78787346` - **Authorized scopes**: Select the scope that ends with `access_as_user`. (There should only be 1 scope in this list.)

7. Click **Add application** to commit your changes.

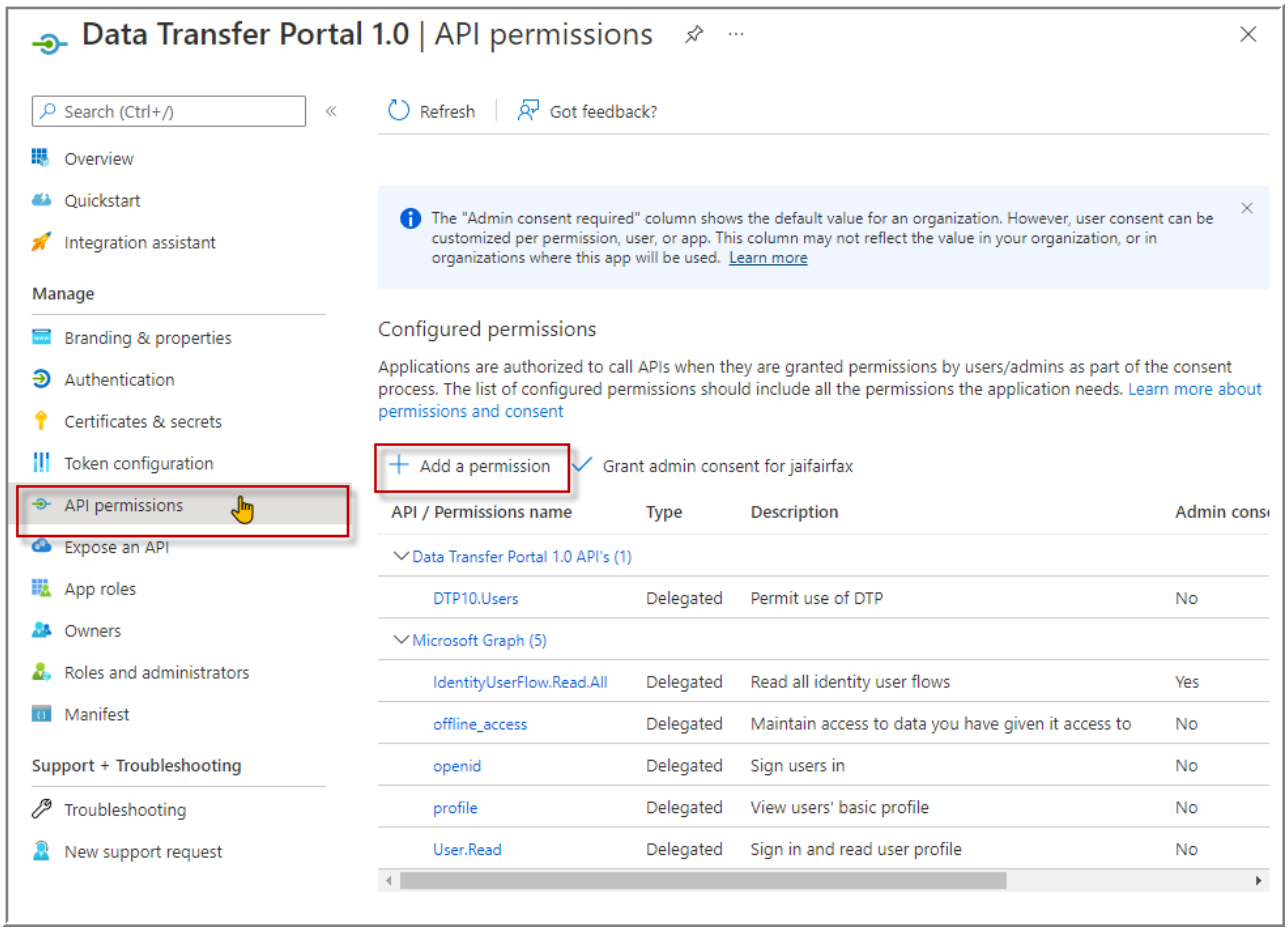
8. Add Permissions to your WebClient app

1. In **Azure Active Directory**, go back to **App Registrations** and select the **WebClient App Registration**



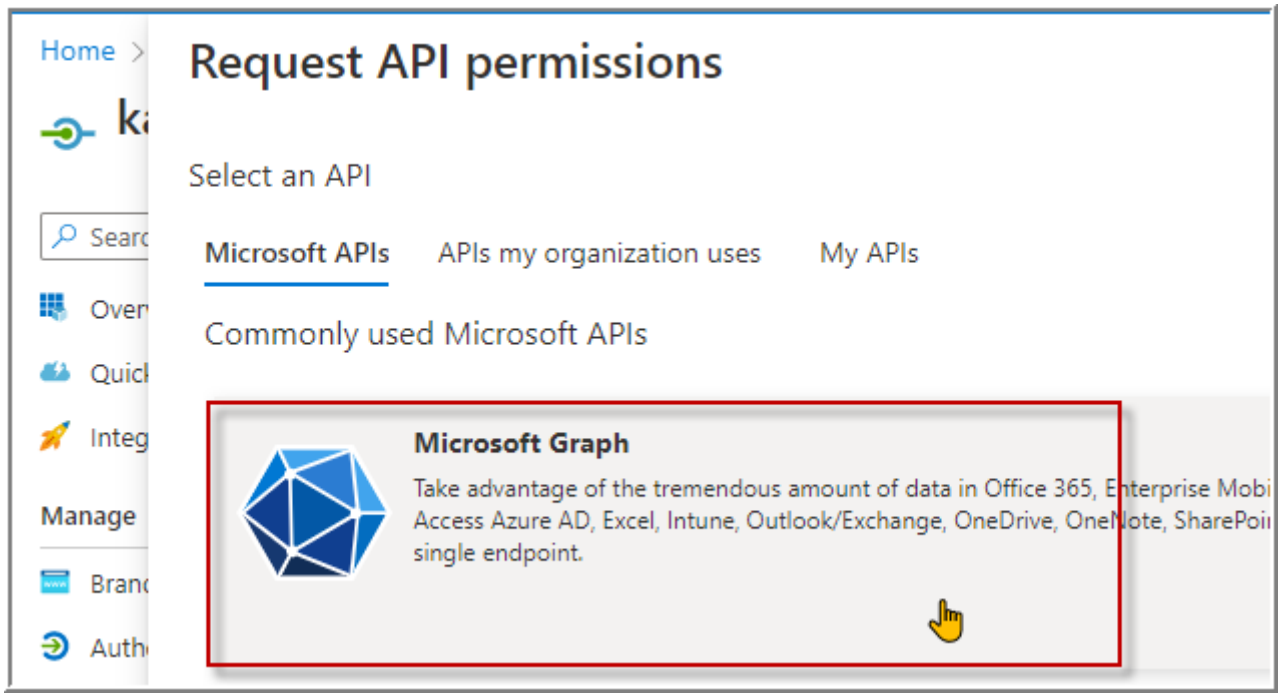
2. Select **API Permissions** blade from the left hand side.

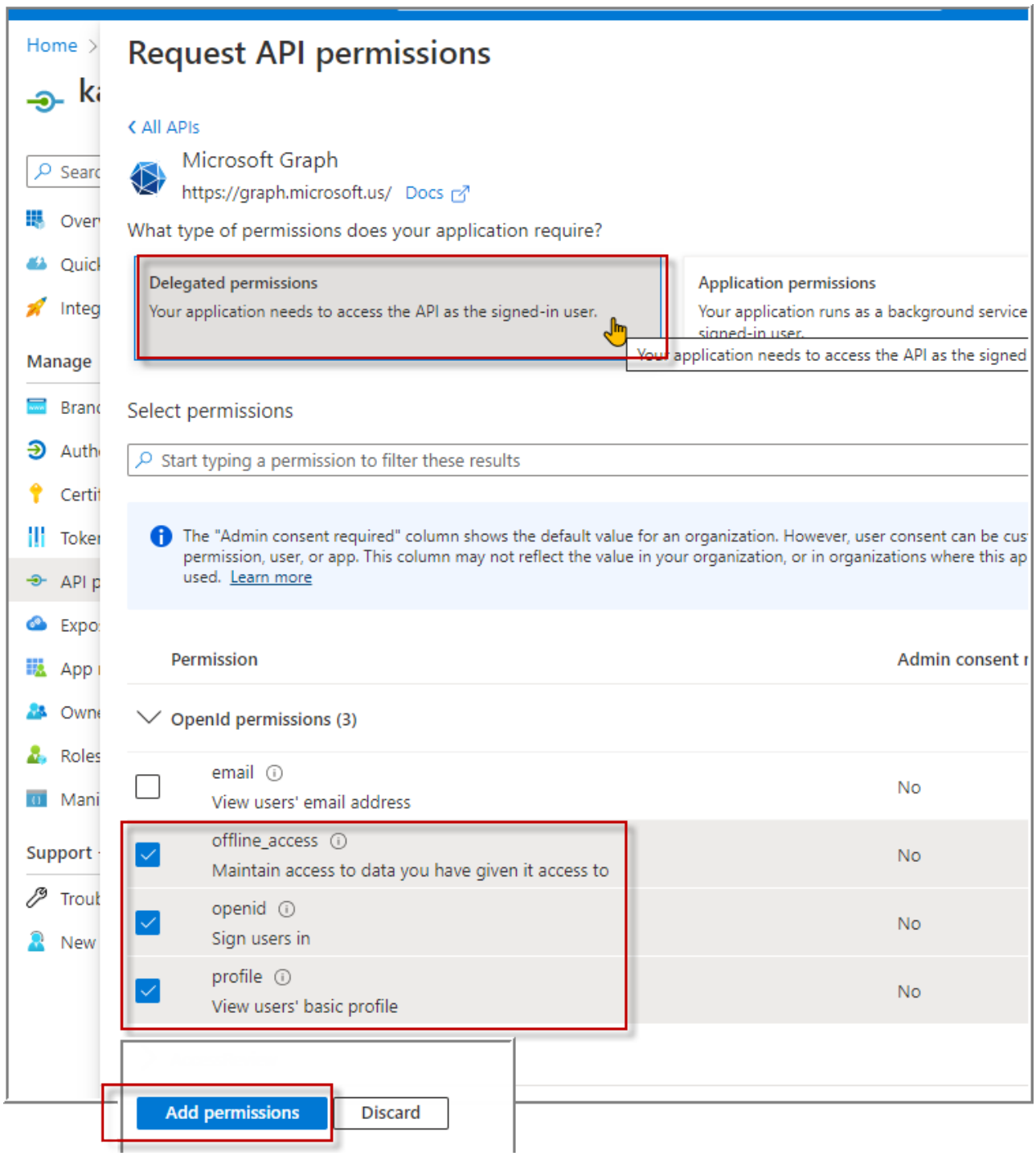
3. Click on **Add a permission** button to add permission to your app.

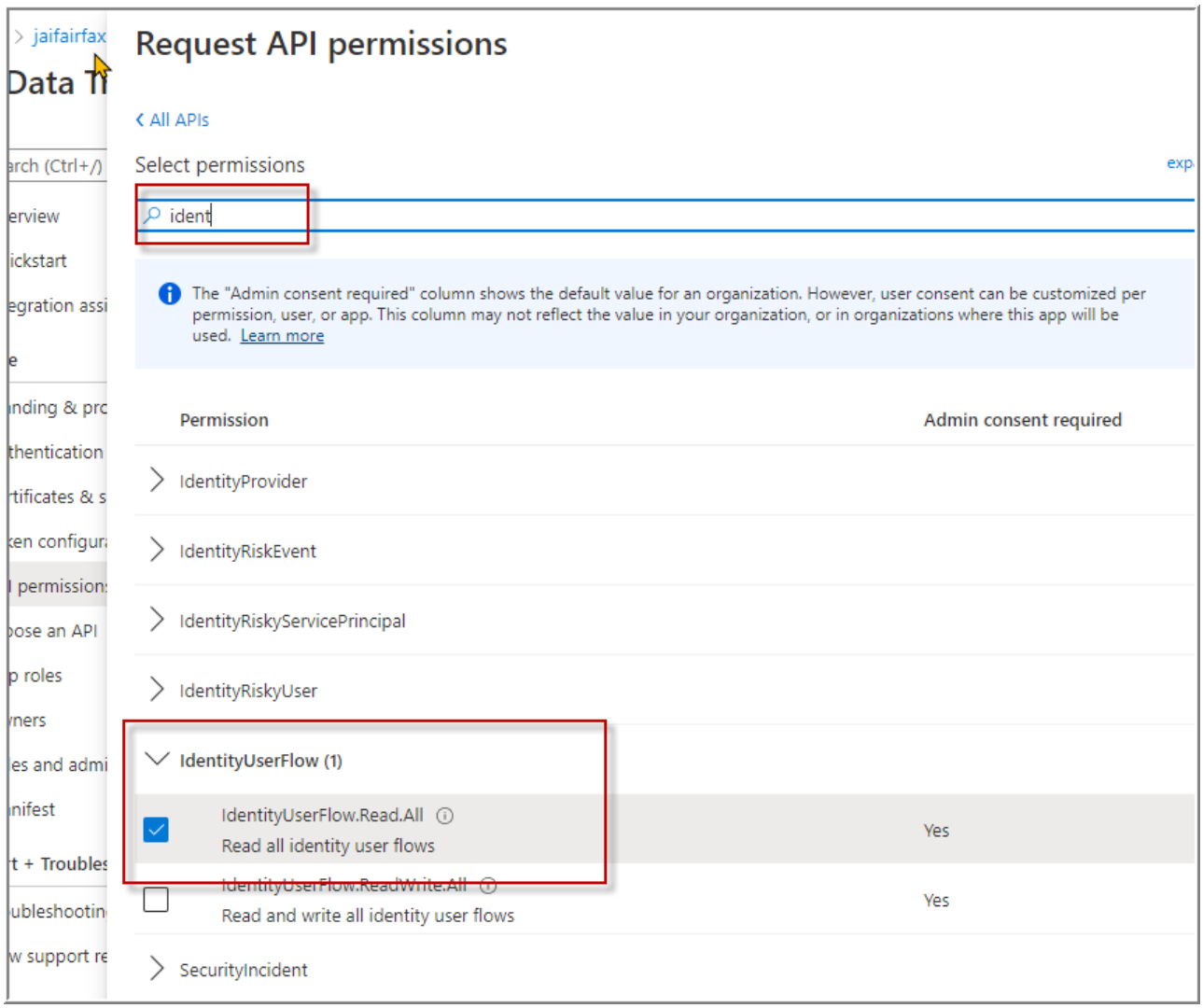


4. In Microsoft APIs under Select an API label, select the particular service and give the following permissions:

Permissions to add:
IdentityUserFlow.Read.All *start typing 'Identity' in the search field to show related permissions*
offline_access
openid
profile
User.Read

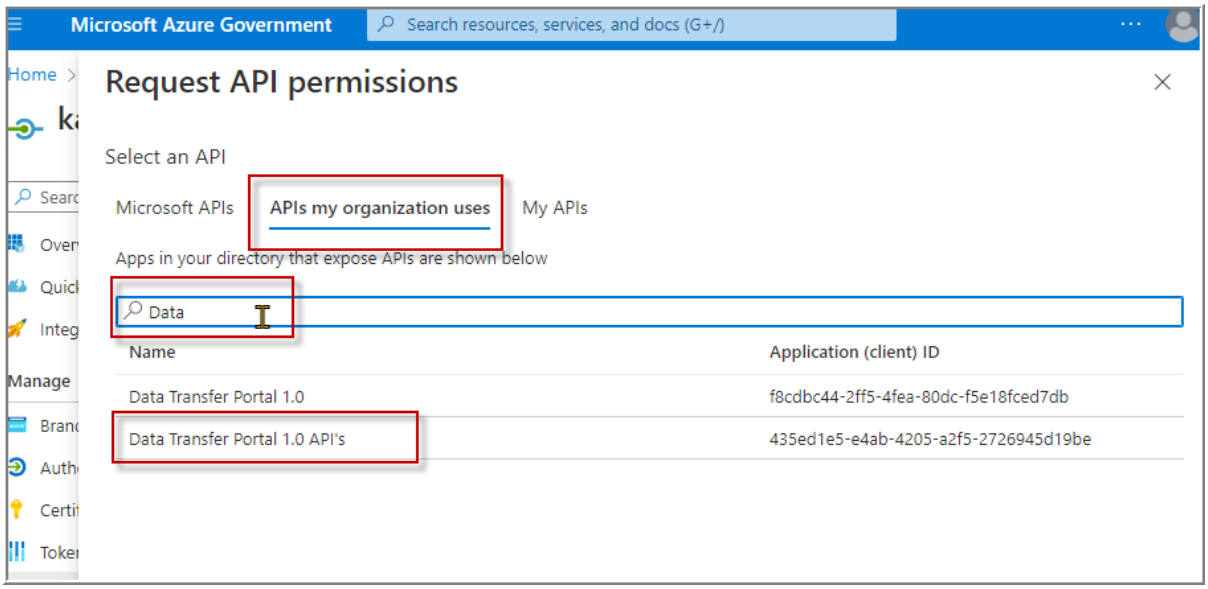



Home
Request API permissions
All APIs
Microsoft Graph
https://graph.microsoft.us/ Docs
What type of permissions does your application require?
Delegated permissions
Your application needs to access the API as the signed-in user.
Application permissions
Your application runs as a background service
Select permissions
Start typing a permission to filter these results
The "Admin consent required" column shows the default value for an organization. However, user consent can be cus
permission, user, or app. This column may not reflect the value in your organization, or in organizations where this ap
used. Learn more
Permission
Admin consent r
OpenId permissions (3)
email
View users' email address
No
offline_access
Maintain access to data you have given it access to
No
openid
Sign users in
No
profile
View users' basic profile
No
Add permissions
Discard
Manage
Support



5. Click on **Add a permission** again and select **APIs my organization uses**

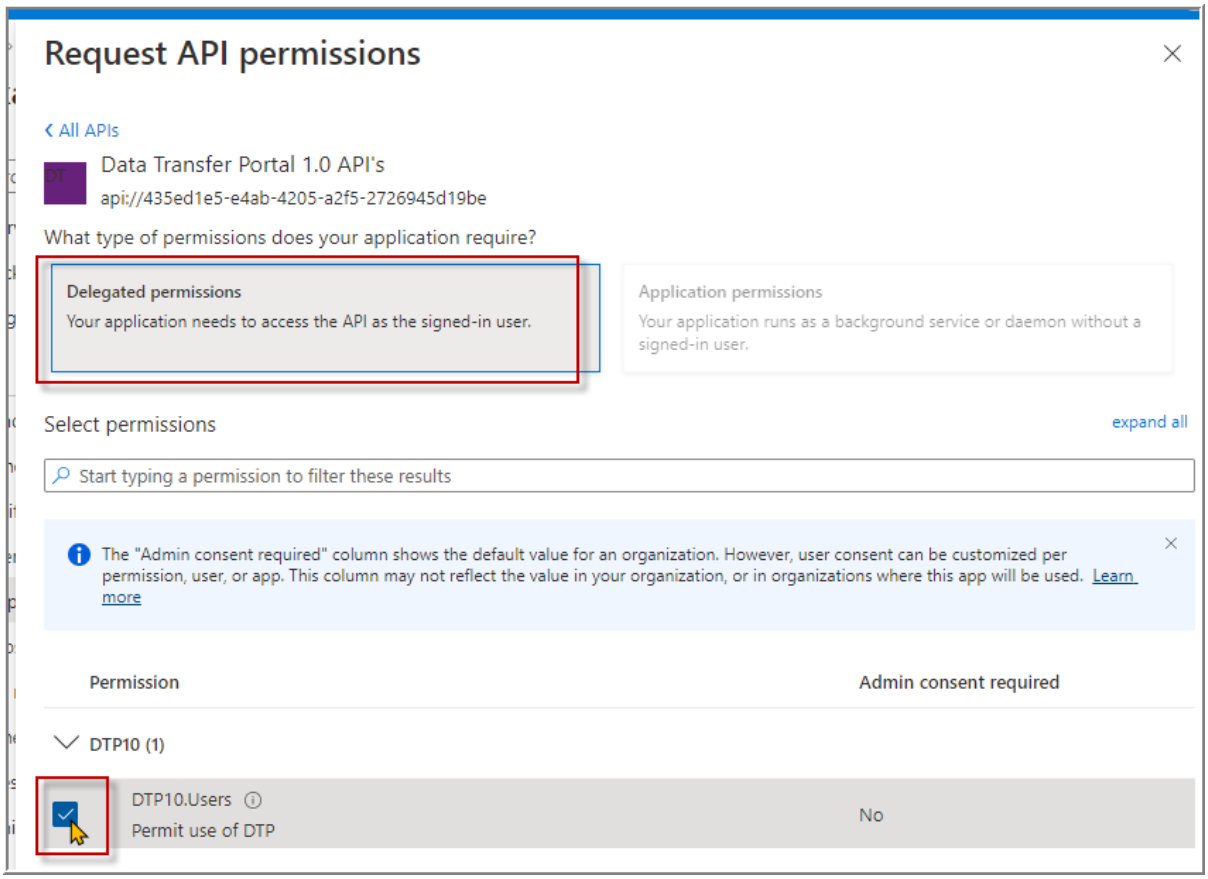
   - In the search field, start typing the name of your **API app registration name** (or simply click on it if it is readily visible) and select it



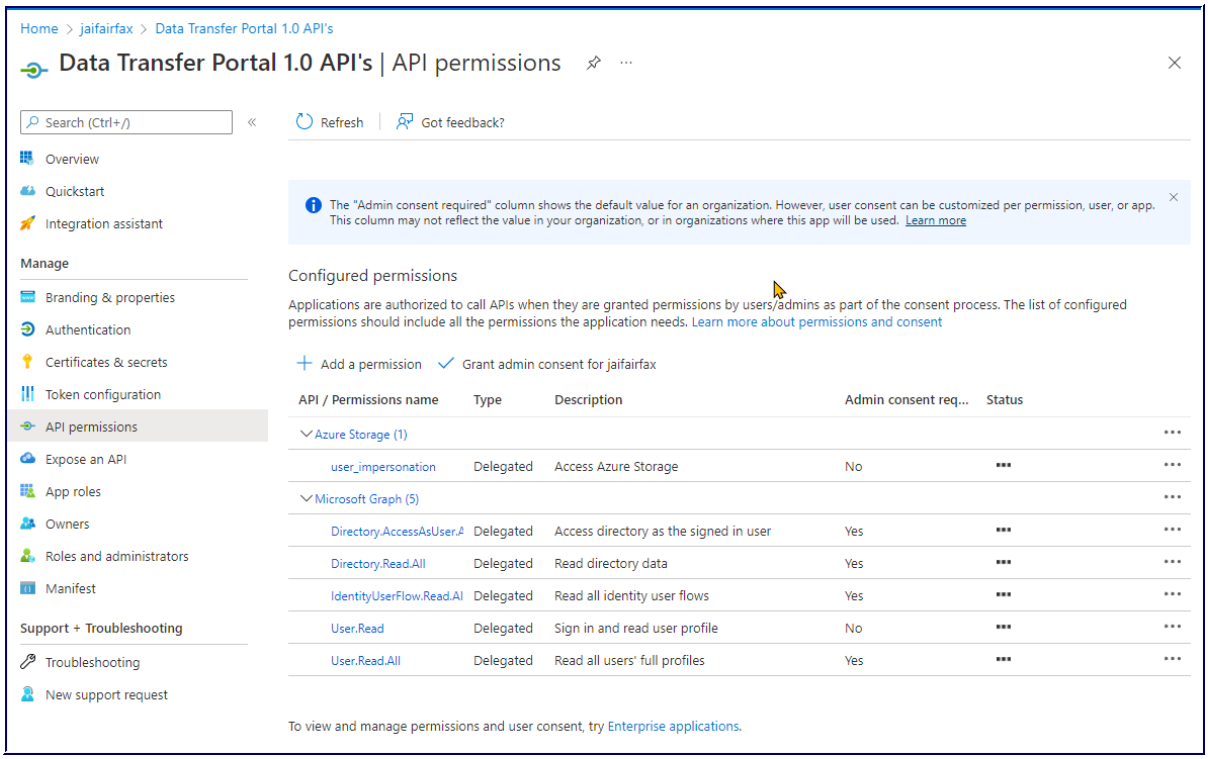
   - Select **Delegated Permissions**

   - Select the expanded permission: **XXX.Users**

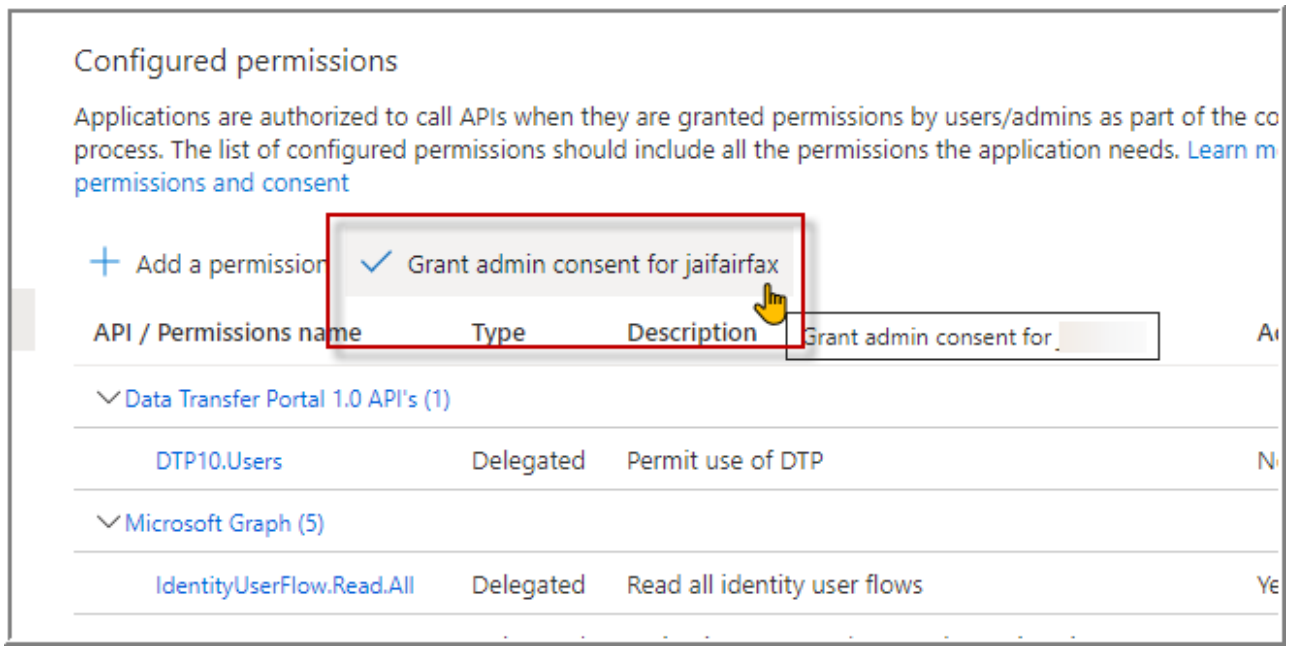
- Click **Add Permissions**



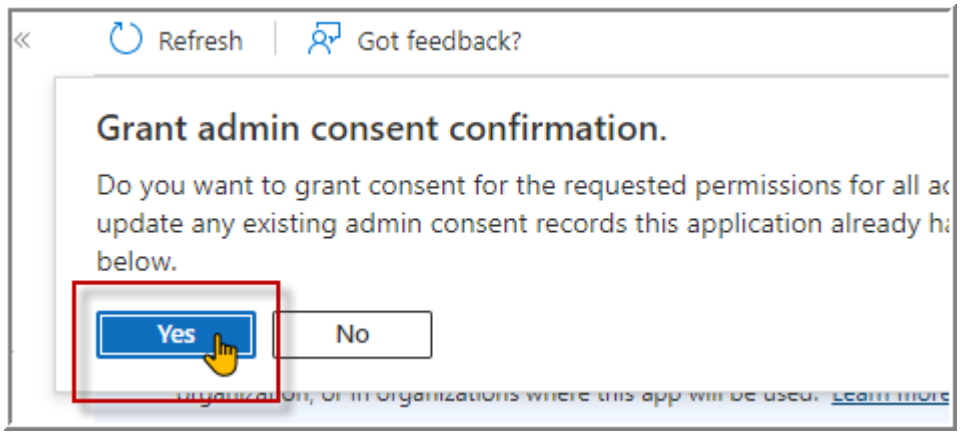
- Click on **Add Permissions** to commit your changes. You should see the results as shown below:



6. If you are logged in as the **Global Administrator**, click on the **"Grant admin consent for %tenant-name%"** button to grant admin consent, else inform your Admin to do the same through the portal.



Click **Yes** in the dialog:



---

9. Run Deployment Script

10. Troubleshooting

Please check the Troubleshooting guide.