Password Expiration Notification using Power Automate

Purpose –

This flow will help us to send account password expiration

Plan –

To create these alerts, we use a flow in Power Automate,

First, we’ll need an app registration in Azure AD with the right permissions to get a list of all the licenses, using Graph API.

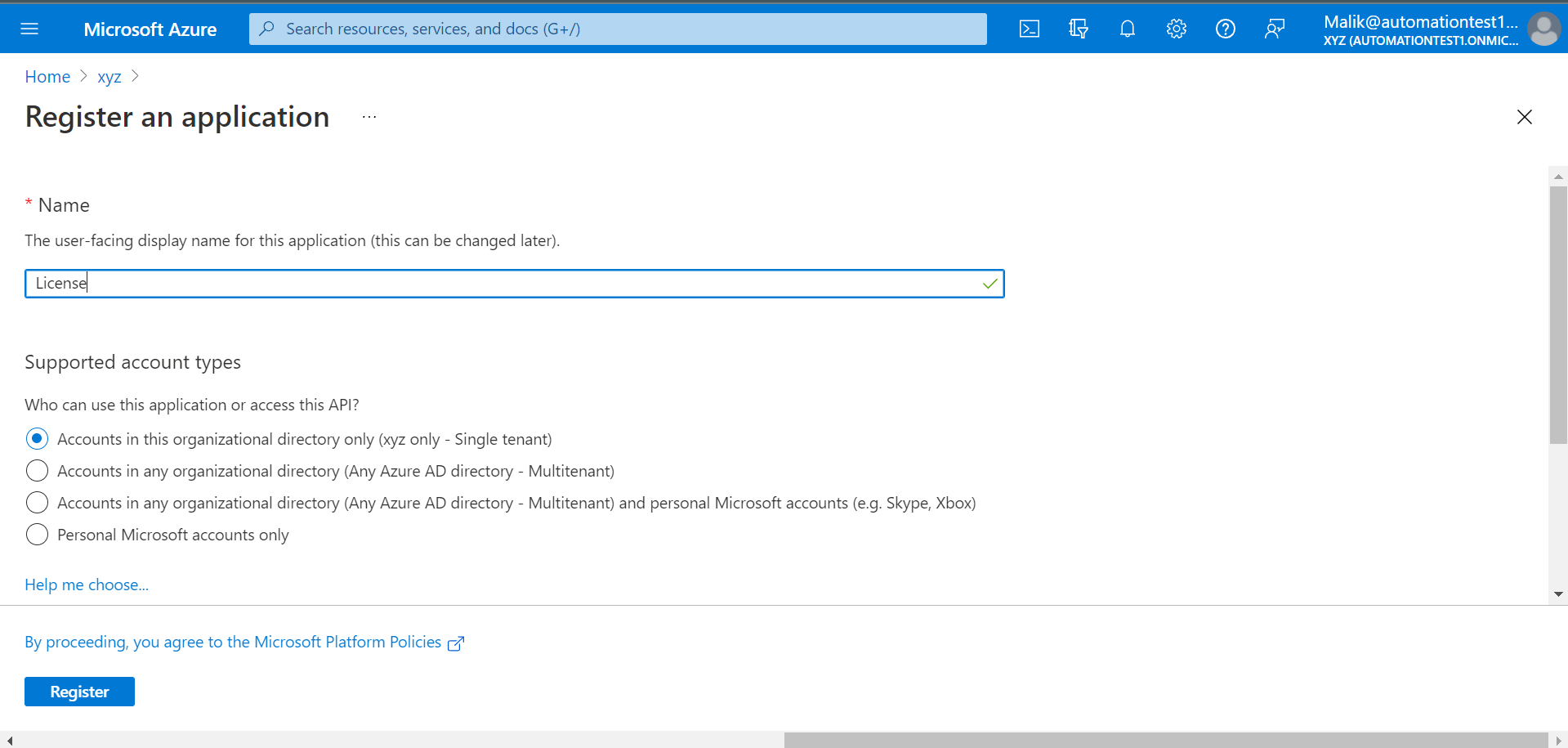
Then the HTTP function we use in Power Automate is a premium feature.

Let’s get start it –

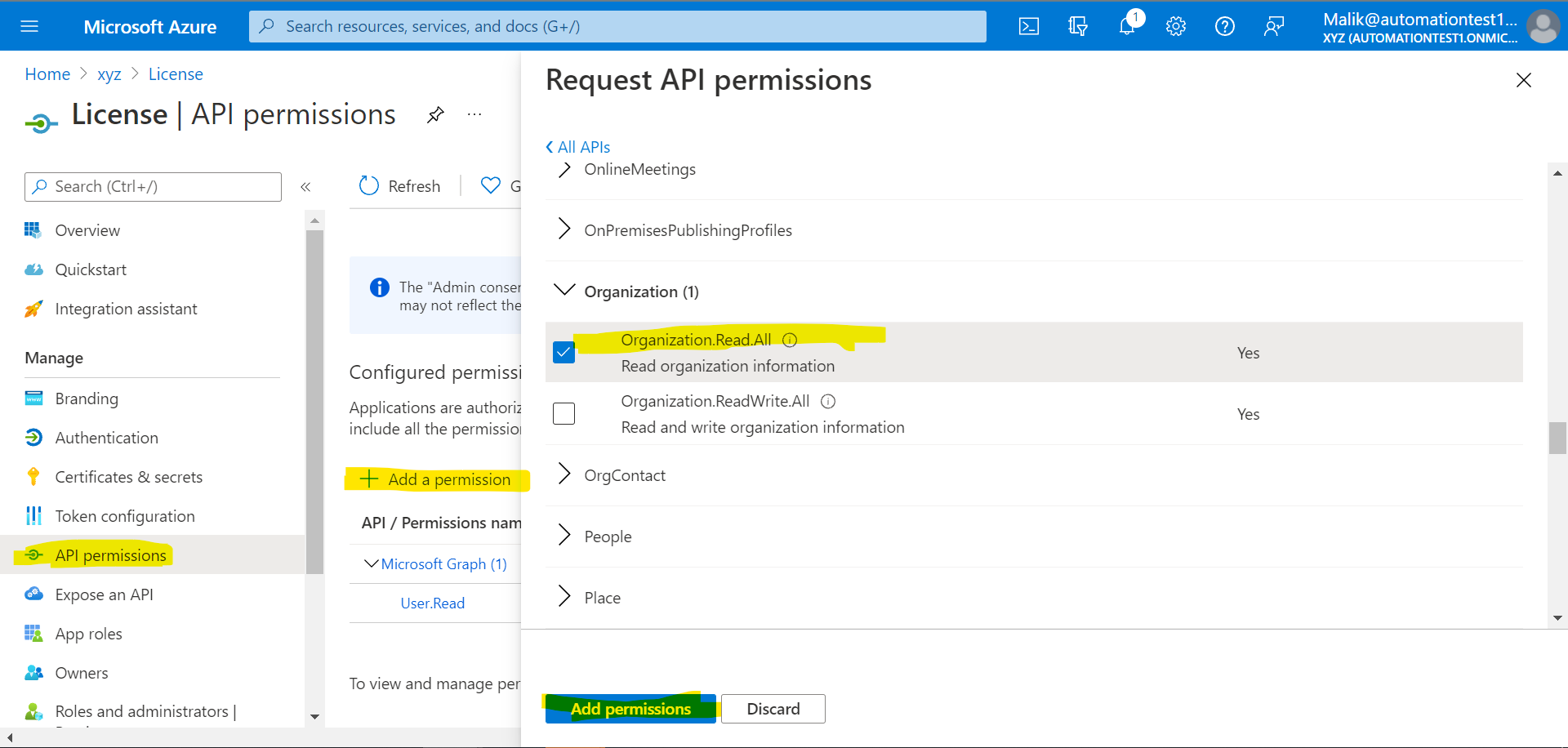
App Registration -

Let’s start with creating an app registration in Azure AD. It will read data from the Azure portal which includes license count.

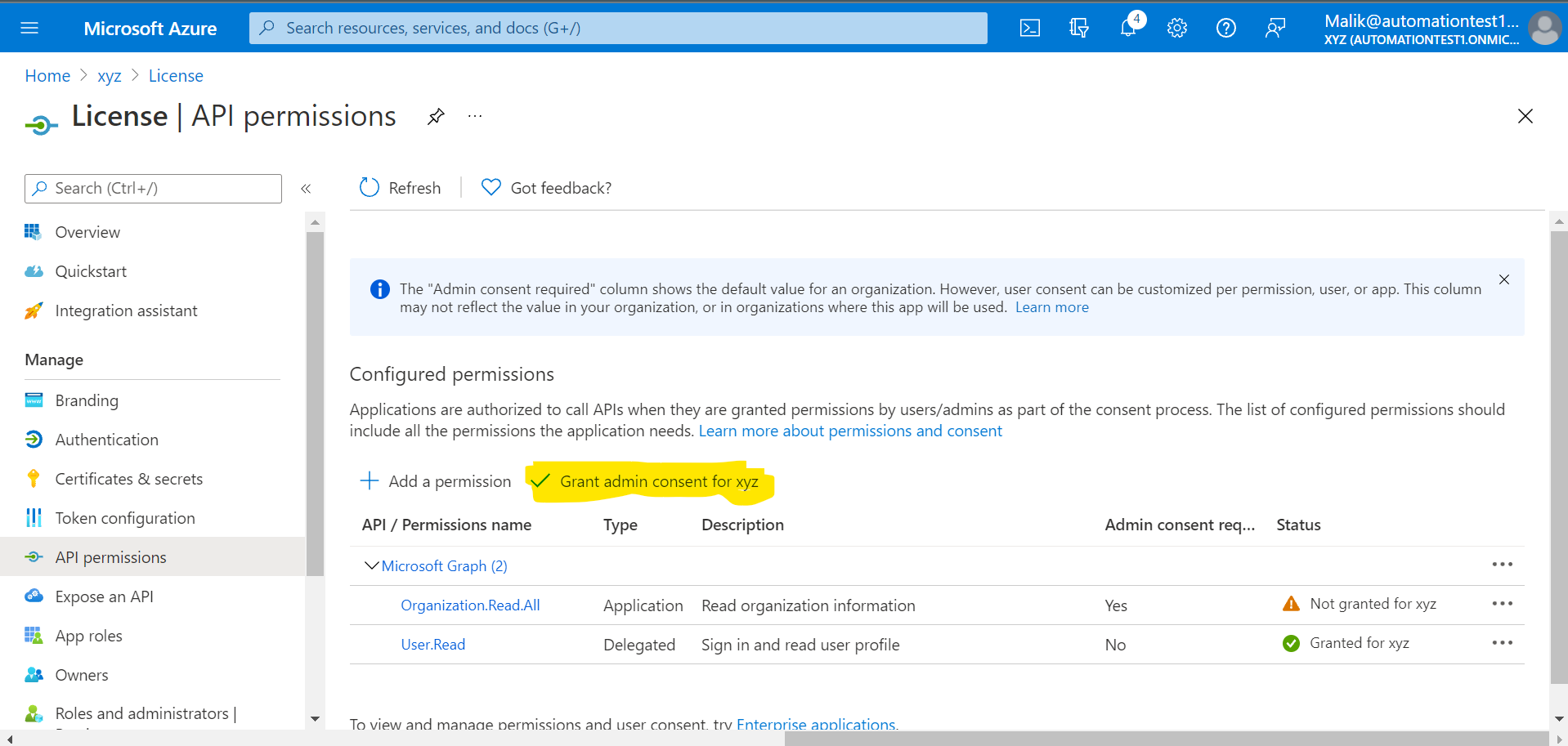
You can find your app registrations under Active Directory -> App registrations in the [Azure portal](https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/RegisteredApps).



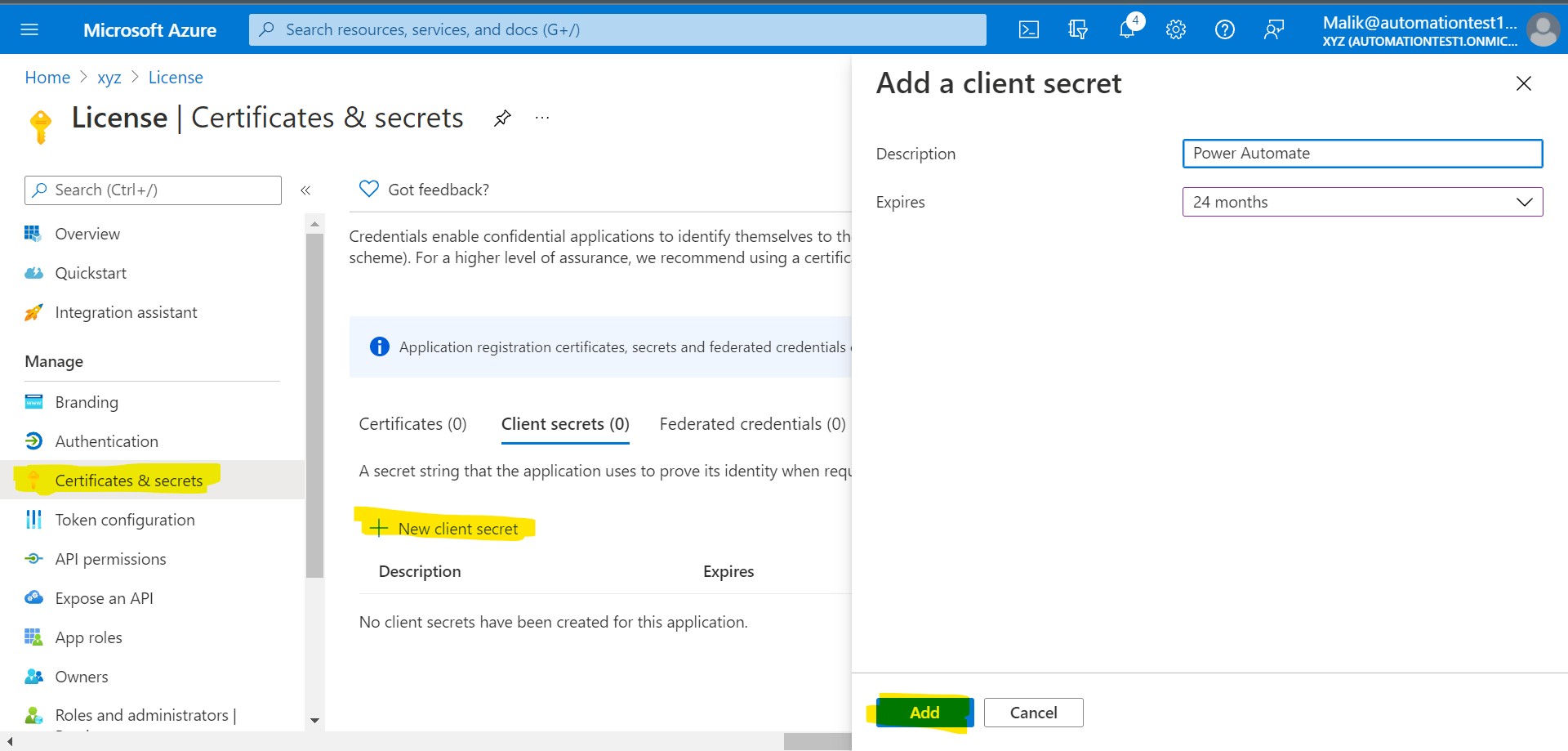
Next, make sure the app registration has the right permissions. Since we are using the [List subscribedSkus](https://docs.microsoft.com/en-us/graph/api/subscribedsku-list?view=graph-rest-1.0&tabs=http) API, you’ll need at least *Organization.Read.All* application permissions for Graph API.



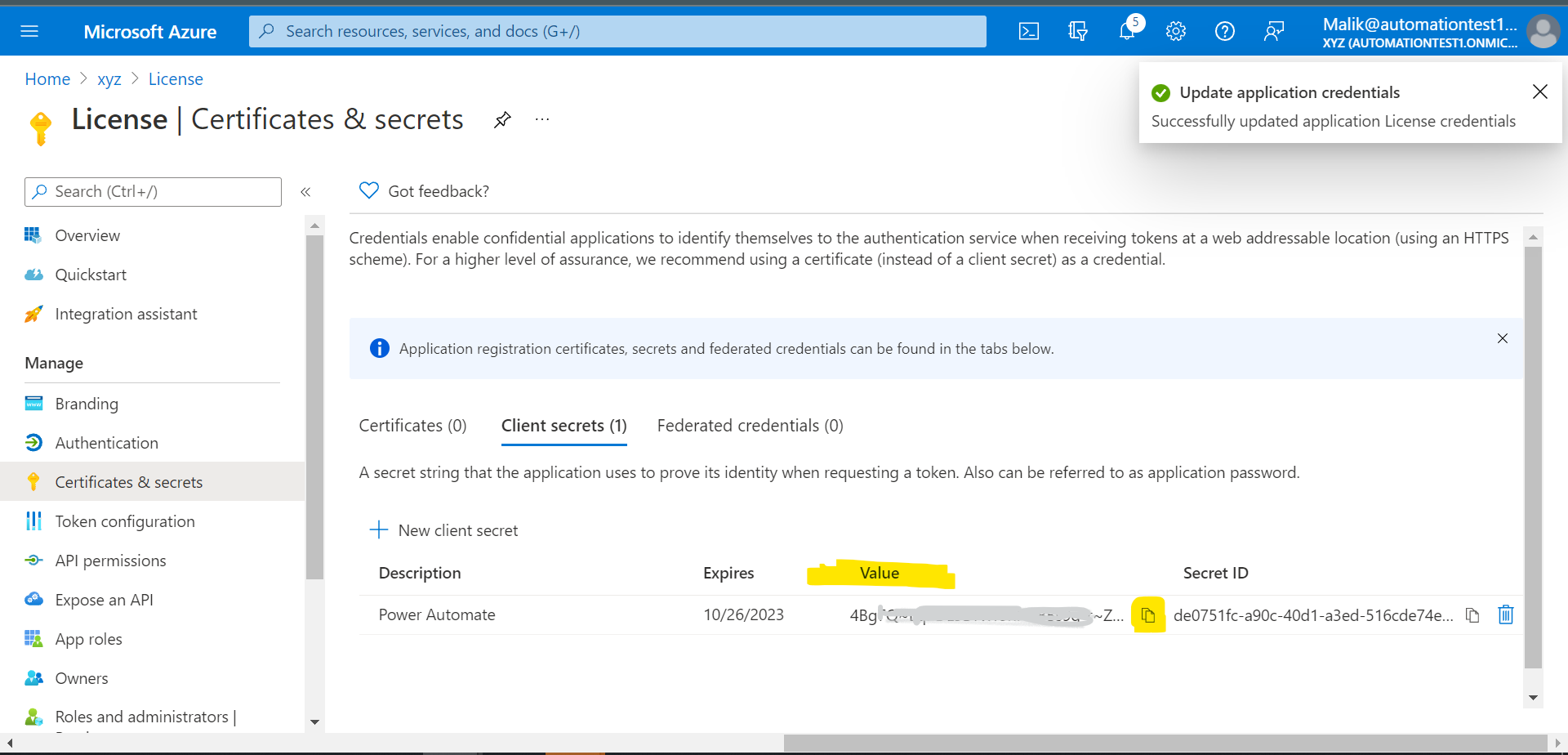
Don’t forget to configure the admin consent afterward.



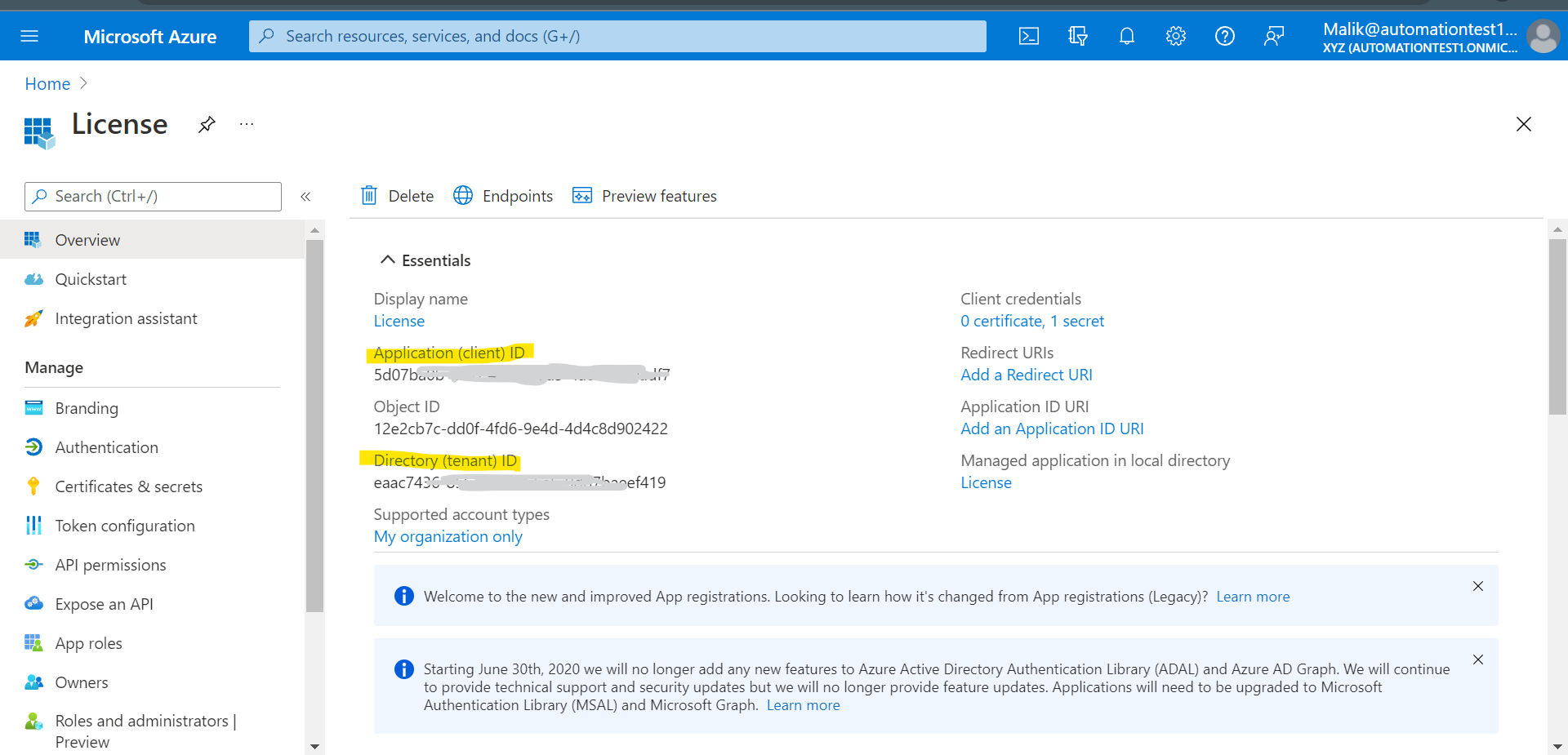
Next, we’ll need a client secret for authentication. Create one and copy the secret to the notepad. We need this in the next steps.



Don’t forget to copy the secret value (shown below).



Also, take note of the Application (client) ID en Directory (tenant) ID.

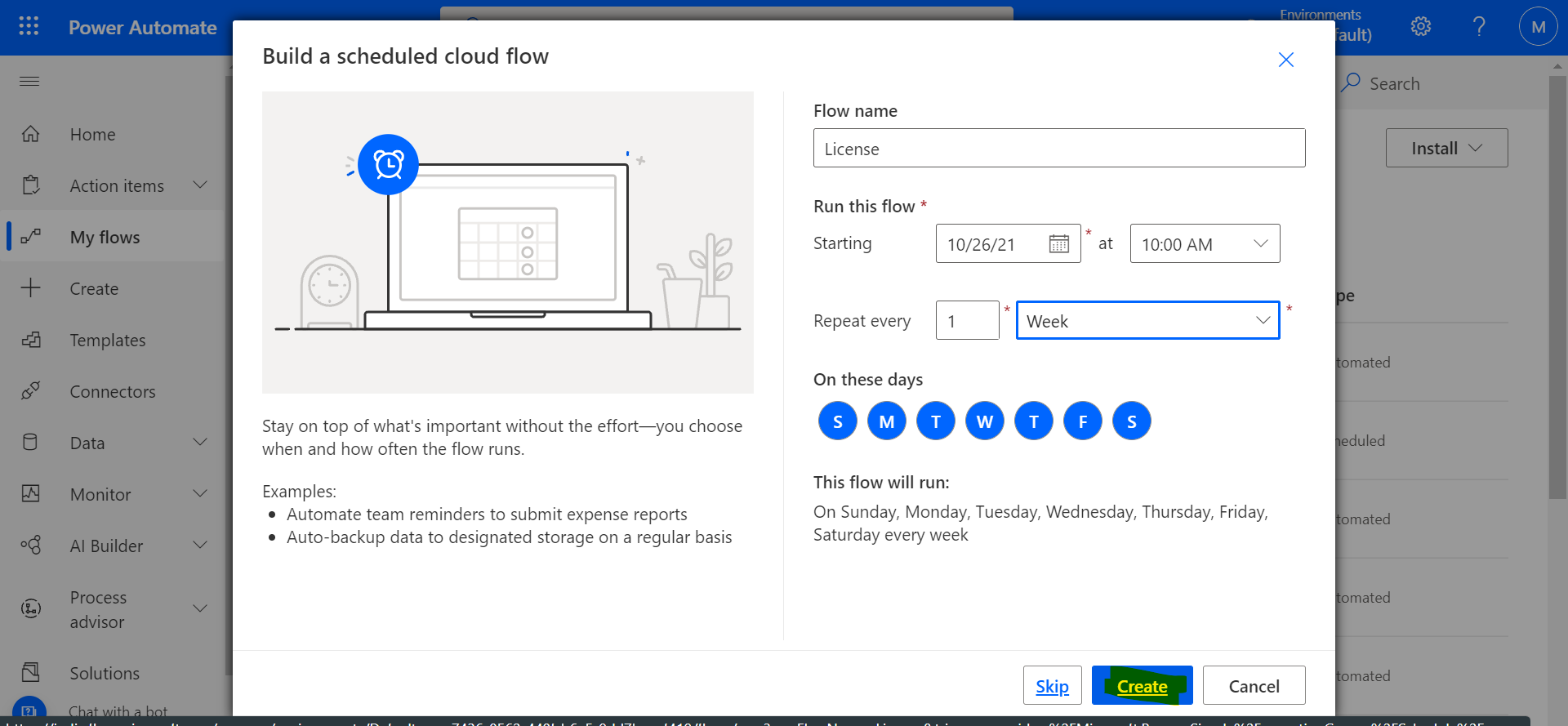


So if you followed all the steps, you end up with:

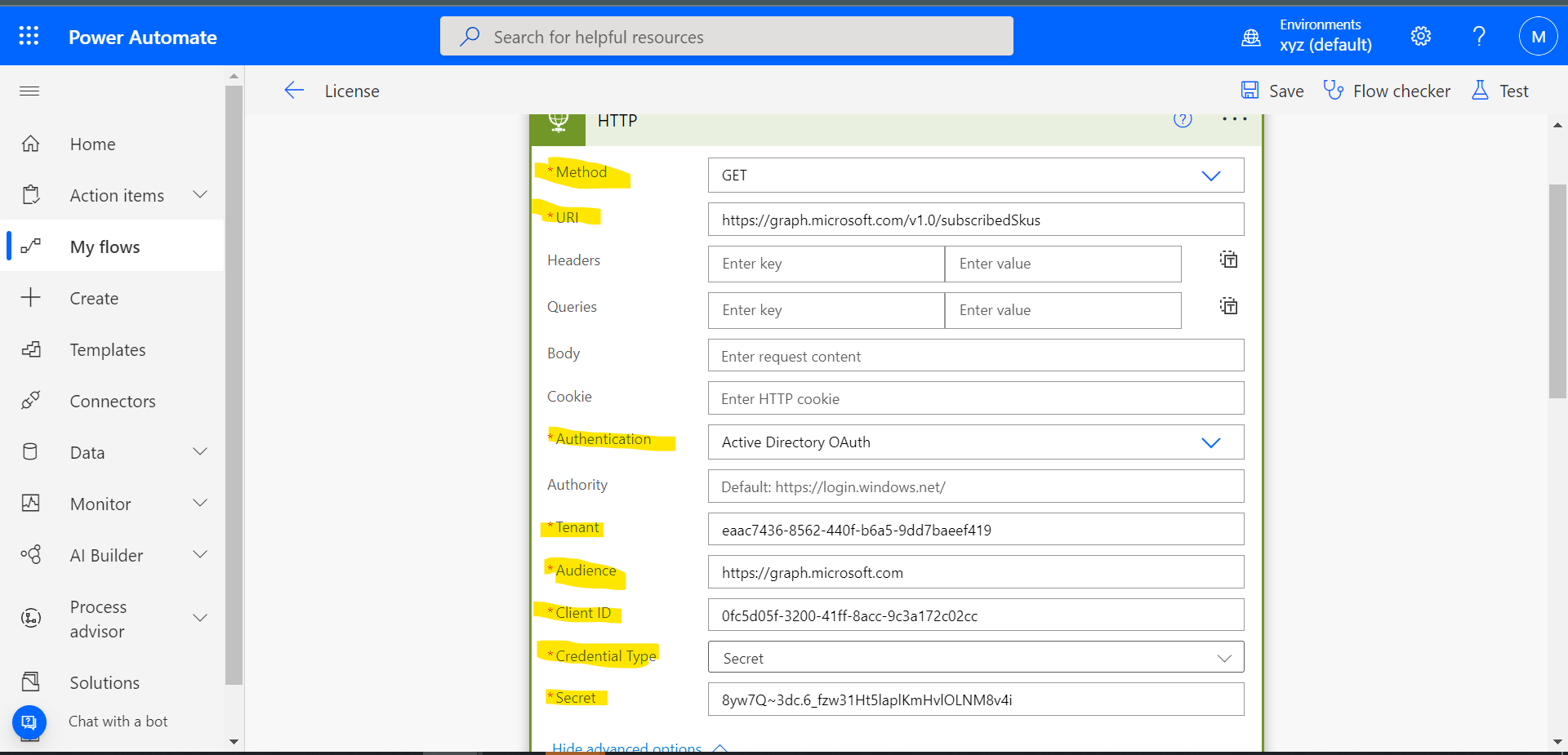
* An app registration with the right Graph API permissions
* A client secret
* The Application (client) ID
* The Directory (tenant) ID

Create the flow -

With all the previous steps in place, we can now go on and create a flow. Head over to your [Power Automate portal](https://flow.microsoft.com/), and start with an empty flow. Pick the scheduled flow to run this daily or weekly, depending on your needs. In our case let’s take it weekly.

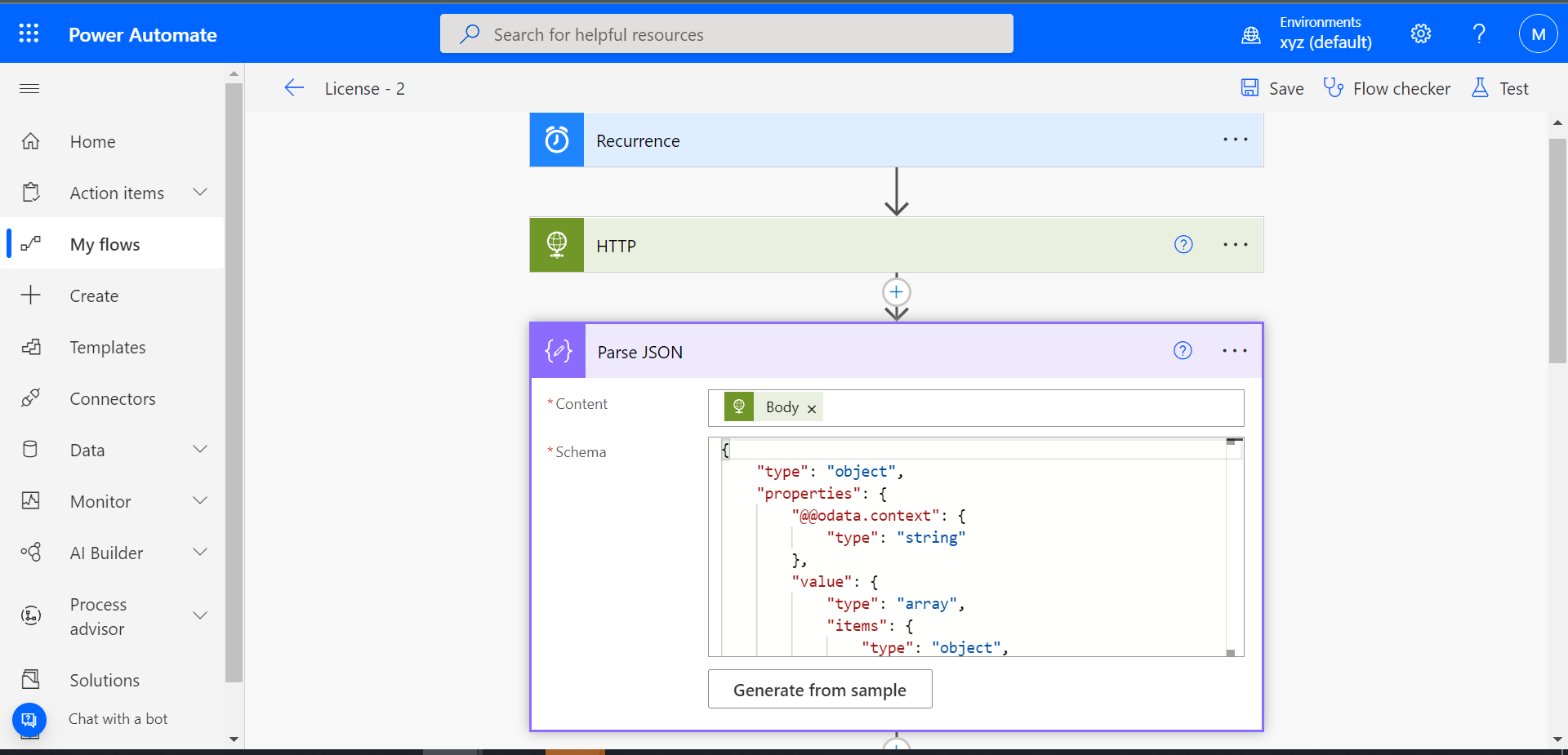


Our first step is an HTTP request to pull out the license information from our tenant.

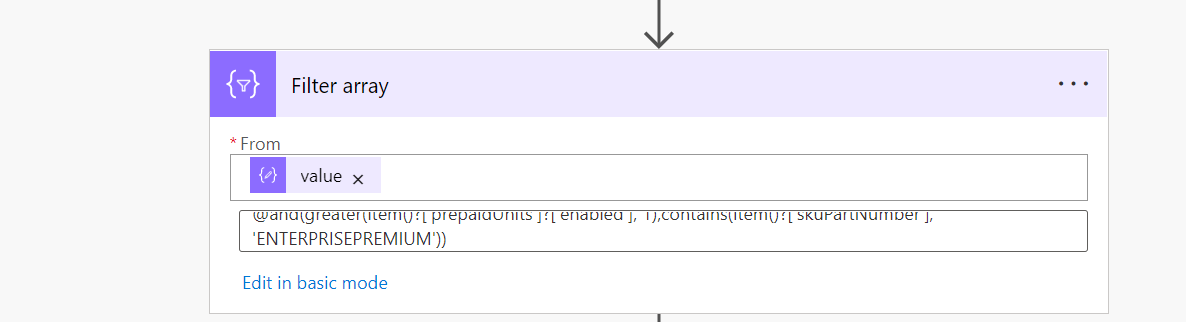


1. Use the GET method
2. Use the https://graph.microsoft.com/v1.0/subscribedSkus API
3. Select Active Directory OAth as your authentication method
4. Enter the tenant ID that you’ve captured in the previous chapter
5. Use https://graph.microsoft.com as your audience URL
6. Enter the Client ID that you’ve captured in the previous chapter
7. Enter your secret

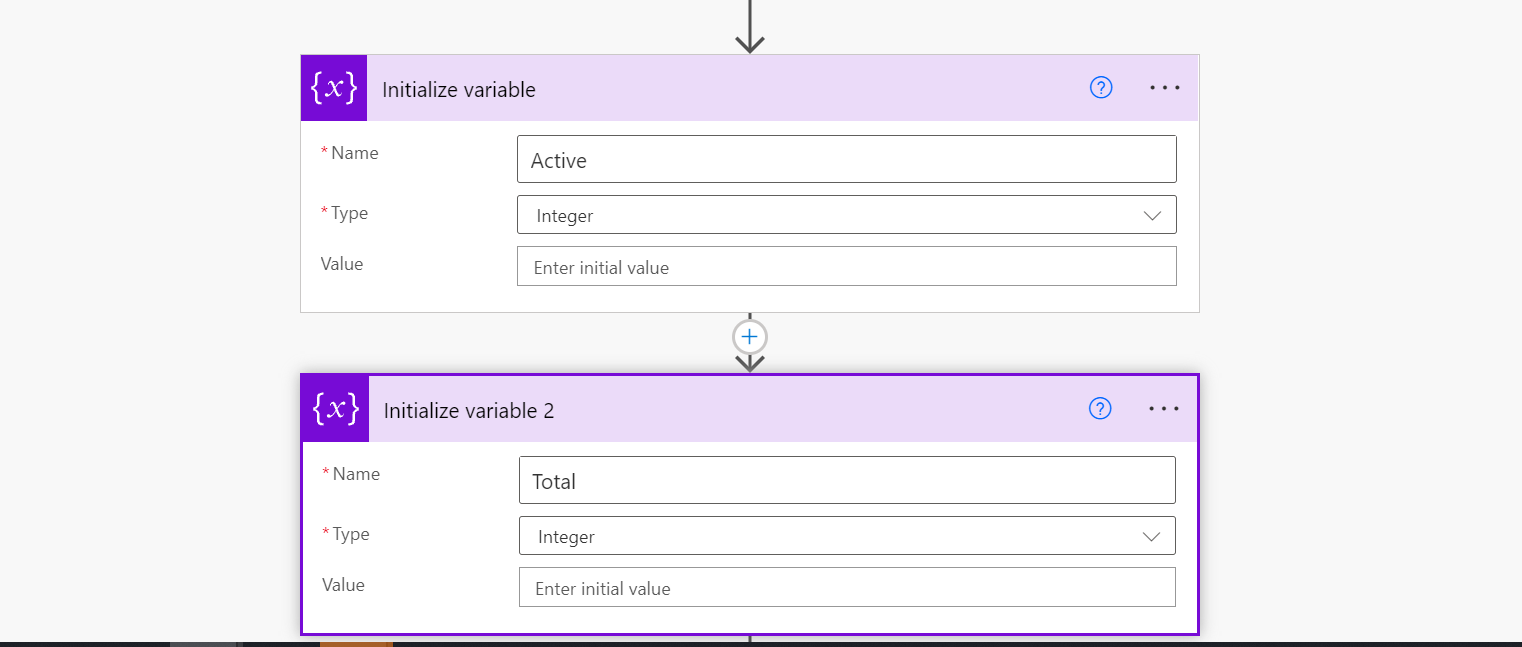
We can now go on and parse our data. Add a new step and select **Parse JSON**. Use the body of the HTTP request and use the output of your HTTP request as sample data to create the schema.



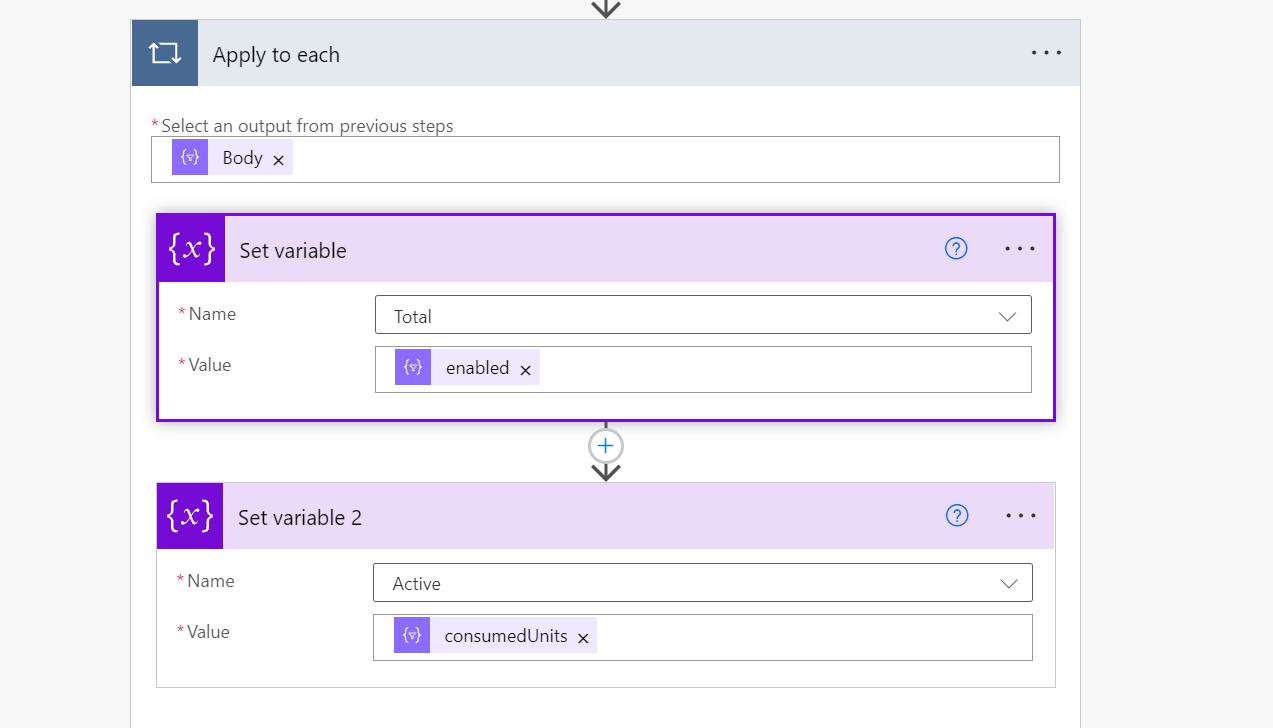
In our next step, we are going to filter the array. I use this step to filter out licenses that have **0** active licenses and filter it out for only e5 licenses.



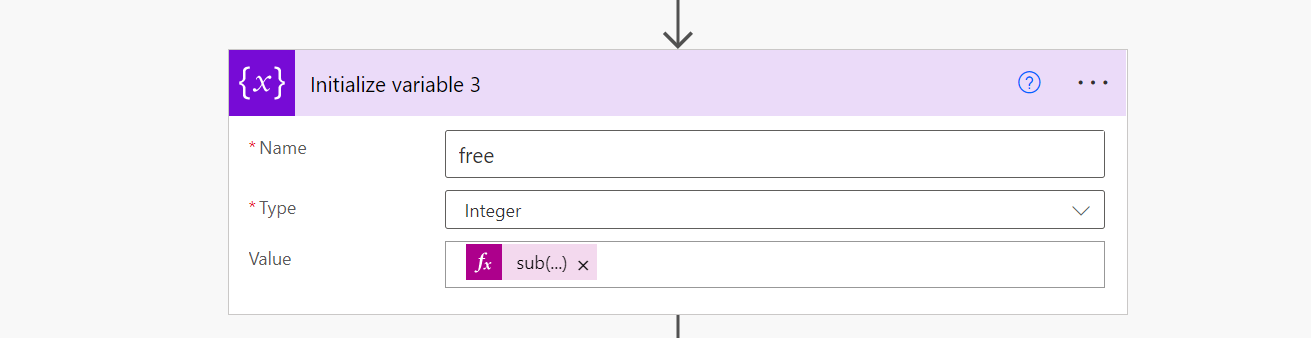
Next, we are going to initialize two variables.



In the next step, create an*Apply to each* action and set the variables according to the figure below.



Next, we need to find out free licenses. We can use an expression to substruct active licenses from total licenses.



Next, we can use conditions to check if free licenses are less than or equal to 100. And send an email if it is true.

