1. User Guide: Configuring the Email Meeting Scheduler

Introduction

This guide provides step-by-step instructions for configuring a Power Platform solution that utilizes a custom connector to interact with the Microsoft Graph API. To ensure security and proper functionality in your own environment (tenant), you must create and configure an Azure Active Directory Application Registration and link it to the custom connector.

By following this guide, you will:

- Install the provided Power Platform solution from market place.
- Create and configure an App Registration in your Azure tenant.
- Securely connect the custom connector to your Azure application.
- Set up all necessary connections and environment variables.
- Test the solution to ensure it is working correctly.

Prerequisites

Before you begin, please ensure you have the following:

- Microsoft account
- A Power Apps ond Power Automate license.
- Permissions to create and manage App Registrations in your Azure Active Directory (Azure AD). This typically requires a role like *Application Administrator* or *Global Administrator*.
- An Outlook account to create the required connection.

Configuration Steps

The configuration process is divided into three main parts:

- **Part A: Initial Power Platform Setup – Installing the product from market place**
- **Part B: Azure AD Application Registration - Creating the secure identity for your connector.**
- **Part C: Final Power Platform Configuration - Linking everything together and testing. Part A: Initial Power Platform Setup**

In this part, we will install the solution and find the **Redirect URL**, which is required for the Azure setup.

Part A: Initial Power Platform Setup

1. **Install the Solution:**
   - ○ Go to marketplace: https://marketplace.microsoft.com/en-us/product/dynamics-365/arpatech1626357569648.email_meeting_scheduler?tab=Overview
   - ○ Select the correct environment
   - ○ If no environment is available than ask your power platform admin to assign one to your user
   - ○ Install the solution
2. **Locate the Custom Connector and Redirect URL:**
   - ○ Go to power apps https://make.powerapps.com/
   - ○ Select your environment from top right corner if not selected automatically
   - ○ Once the solution is installed, open it from the solutions list in managed solution tab.

   Find Arpatech_EmailMeetingScheduler solution in managed solution tab
   - ○ Click to open the solution
   - ○ In the left panel under objects, find and click on the **Custom Connector**.
   - ○ Click on the custom connector name
   - ○ Click the **Edit** button in the command bar. This will open the connector's configuration wizard.
   - ○ Navigate to the **2. Security** tab.
   - ○ Under the **OAuth 2.0** section, you will see a field labeled **Redirect URL**.

- ○ **Copy this URL** to a notepad. You will need it in the next part. Do not close this browser tab.

Part B: Entra ID Application Registration

Now, we will create the application in your Azure tenant that the connector will use to authenticate.

1. **Create a New App Registration:**
   - ○ Open a new browser tab and navigate to the Azure Portal at portal.azure.com.
   - ○ Use the search bar at the top to find and select **Entra ID**.
   - ○ From the left menu, expand manage and select **App registrations**, then click **+ New registration**.
2. **Register the Application:**
   - ○ **Name:** Give your application a descriptive name (e.g., "Power Automate Graph API Connector").
   - ○ **Supported account types:** Select **Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)**. This ensures maximum compatibility.
   - ○ Leave the **Redirect URI** section blank for now. We will add it in the next step.
   - ○ Click **Register**.
3. **Configure Authentication:**
   - ○ Once the app is created, you will be taken to its overview page. From the left menu, select **Authentication**.
   - ○ Click **+ Add a platform** and select **Web**.
   - ○ In the **Redirect URIs** field, **paste the Redirect URL** you copied from the custom connector in Part A.
   - ○ Click **Configure**.
4. **Create a Client Secret:**
   - ○ From the left menu, select **Certificates & secrets**.
   - ○ Click **+ New client secret**.
   - ○ Provide a description (e.g., "Connector Secret") and choose an expiration period. **Note:** You will need to renew this secret before it expires.
   - ○ Click **Add**.
5. **Record Your Credentials:**
   - ○ The client secret **Value** will now be visible. **This is the only time it will be fully shown.** Copy the **Value** immediately and save it securely in your notepad.
   - ○ Go back to the **Overview** tab of your app registration.

○ Copy the **Application (client) ID** and the **Directory (tenant) ID** and save them in your notepad.

You should now have three critical pieces of information saved:

○ Client ID

○ Tenant ID

○ Client Secret

Part C: Final Power Platform Configuration

Now we will return to Power Apps to finalize the setup.

○ **Update the Custom Connector Security:**

○ Go back to the browser tab where your custom connector is open for editing (from Part A, Step 2).

○ Navigate to the **2. Security** tab. Select generic Oauth 2 ○ Fill in the fields with the credentials you just saved from Azure:

- Client ID: Paste your Application (client) ID.
- Client secret: Paste your Client Secret Value.
- Tenant ID: common
- Authorization URL: https://login.microsoftonline.com/common/oauth2/authorize
- Resource URL:
- Enable on-behalf-of login: false
- Scope: openid profile offline_access User.Read Mail.Read Mail.ReadWrite Calendars.Read Calendars.ReadWrite

○ Update connector and Click on test tab in connector

○ Create a new connection and connect with your outlook account

○ Click **Update connector** at the top.

○ **Create Connections:**

○ In the Power Apps portal (make.powerapps.com), on the left navigation pane, go to **Data** > **Connections**.

○ Click **+ New connection**.

○ Search for **Outlook.com** and create a connection by signing into your account.

○ **Update Solution Connection References:**

○ Go back to **Solutions** and open **default solution**.

○ You will see **Connection References** listed.

○ Search for the connection references that were imported from the installed solution

- ○ Go to your managed solution and note down the names of connection references used
- ○ Create connection for each of the connection reference and link with it

○ **Update Environment Variables (if applicable):**

- ○ Go to default solution and click on environment variables
- ○ Click on each one and provide a **Current Value** that is appropriate for your environment. Click **Save**.

○ **Environment Variable Details**

- ○ **outlook_timezone: Set this value according to your outlook time zone**
- ○ **UTC_Conversion_TimeZone: Set this value to convert UTC into your selected time zone**
- ○ **WorkStart: Set this value as the starting time of your work in this format (HH:MM.SS)**
- ○ **WorkEnd: Set this value as the ending time of your work (HH:MM.SS)**

2. Testing the Flow

Your configuration is now complete. The final step is to run the Power Automate flow to ensure everything works as expected.

1. **Run the Flow:**
   - ○ Within your solution, find and click on the **Cloud Flow**.
   - ○ Make sure connection references are connected otherwise repeat the steps mention above
   - ○ From the command bar, click on turn on
   - ○ Test the flow by sending a testing email to your outlook email
   - ○ Send another email on the same time to your outlook email from other ID
2. **Check the Run History:**
   - ○ After starting the flow, go to the **28-day run history** section on the flow's details page.
   - ○ Click on the run that just started. After a few moments, the status should change to **Succeeded**. If it fails, you can click into the run history to inspect the error and troubleshoot.

Congratulations! Your custom connector and solution are now fully configured and operational in your tenant.