Configuring Office 365 for Universal Connector

12/18/2023 12:26 pm EST

Contents

Purpose

Normal OAuth requires user input user/password for authentication. Obviously, it is not suitable for background service. In this case, you can use the OAuth 2.0 client credentials grant, sometimes called two-legged OAuth, to access web-hosted resources by using the identity of an application. It only works for Office365 users, it doesn't work for personal Hotmail accounts.

Return to Contents

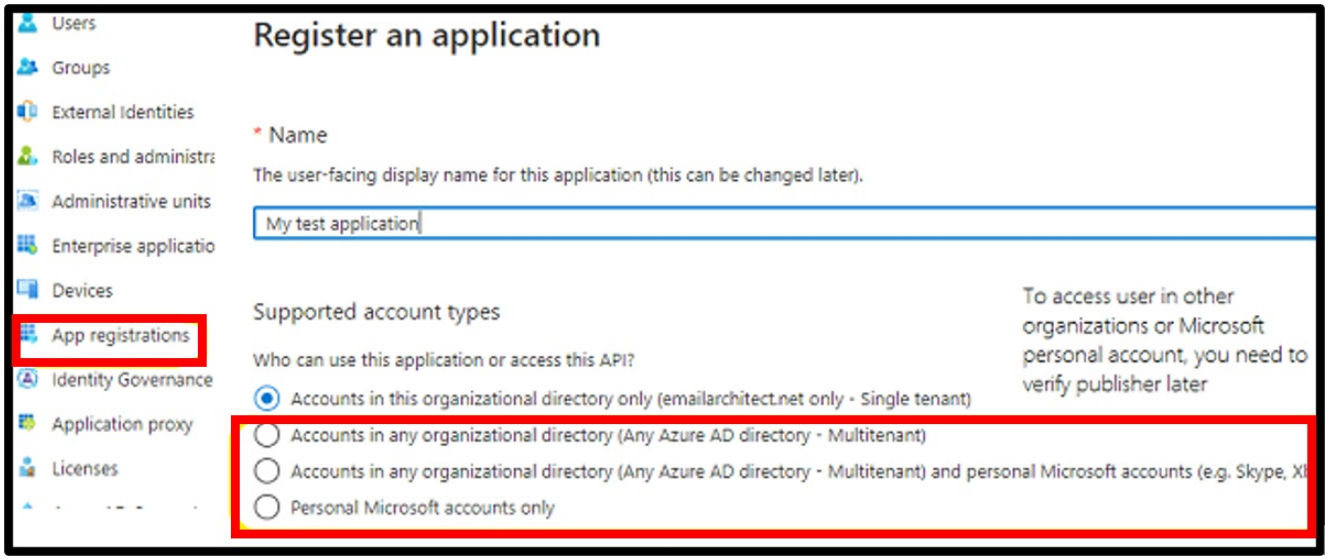
Create your application in Azure Portal

To use Microsoft/Office365/Live OAuth (Modern Authentication) in your application, you must create an application in Azure Portal.

You can use any Microsoft user to create the application, it doesn't require the application owner to be an administrator in your Office365 domain. However, your Office365 administrator must authorize the application to access the user mailbox.

- Sign in to the Azure portal using either a work or school account or a personal Microsoft account.
- If your account gives you access to more than one tenant, select your account in the top right corner, and set your portal session to the Azure AD tenant that you want.

- In the left-hand navigation pane, select the Azure Active Directory service, and then select App registrations -> New registration



Return to Contents

Single tenant and multitenant in account type

When the register an application page appears, enter a meaningful application name and select the account type. Select which accounts you would like your application to support.

- If your application only supports the users in your directory or organization, please select Single tenant type;
- If your application needs to support all users in Office 365 and Microsoft personal accounts (hotmail.com, outlook.com), please select Multitenant type, and you must verify publisher.

Because we just need to support Offic365 user in our organization, select Accounts in this organizational directory only (single tenant).

Do not select supporting Microsoft personal account, because there is no way to access Microsoft personal account in background service.

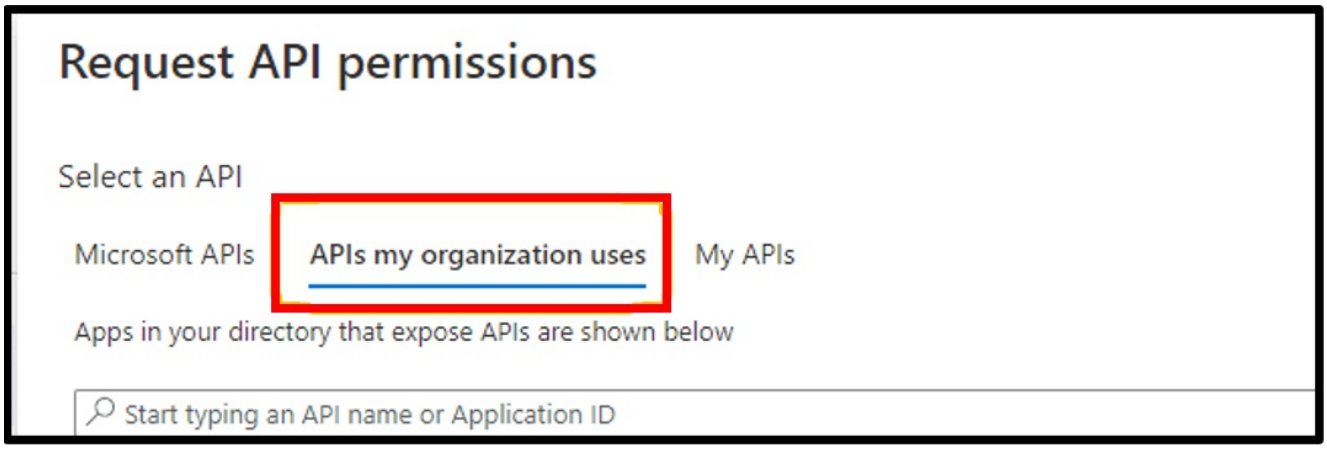
If you don't verify publisher for a multi-tenant application, your application will not request an access token successfully.

Return to Contents

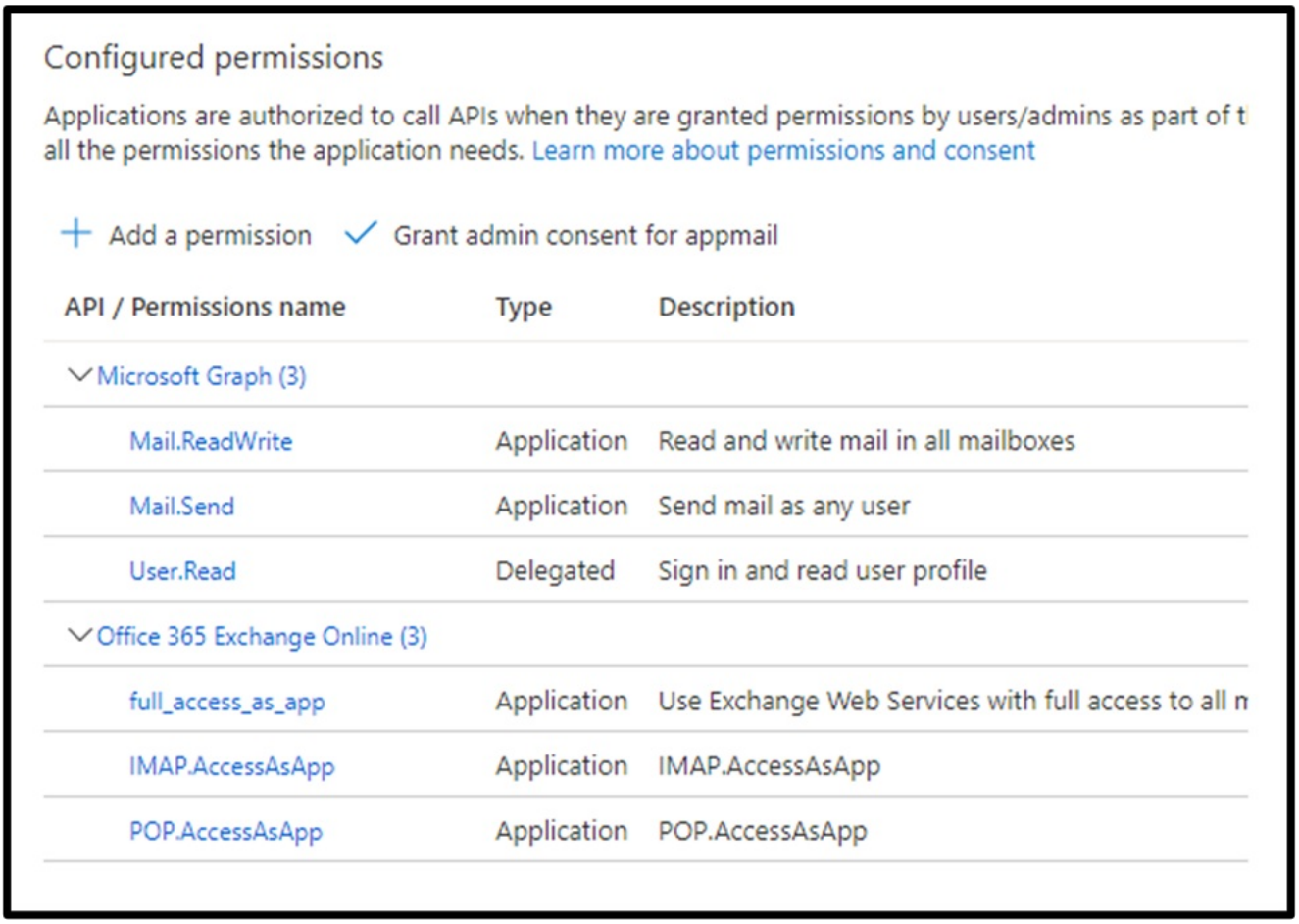
API Permissions

- Click API Permission -> Microsoft Graph -> Delegated Permission -> User.Read.

- Click API Permission -> Microsoft Graph -> Application Permission -> Mail.Send, Mail.ReadWrite.
- Click API Permission -> Add a permission -> APIs in my organization uses -> Office 365 Exchange Online -> Application Permission -> Other permission -> full_access_as_app, IMAP.AccessAsApp and POP.AccessAsApp.



Here is a permissions list:



Return to Contents

Client ID and Client Secrets

Now we need to create a client secret for the application, click Certificates and secrets -> client secrets and add a new client secret.

After client secret is created, store the client secret value to somewhere.

Please store client secret value by yourself because it is hidden when you view it next time.

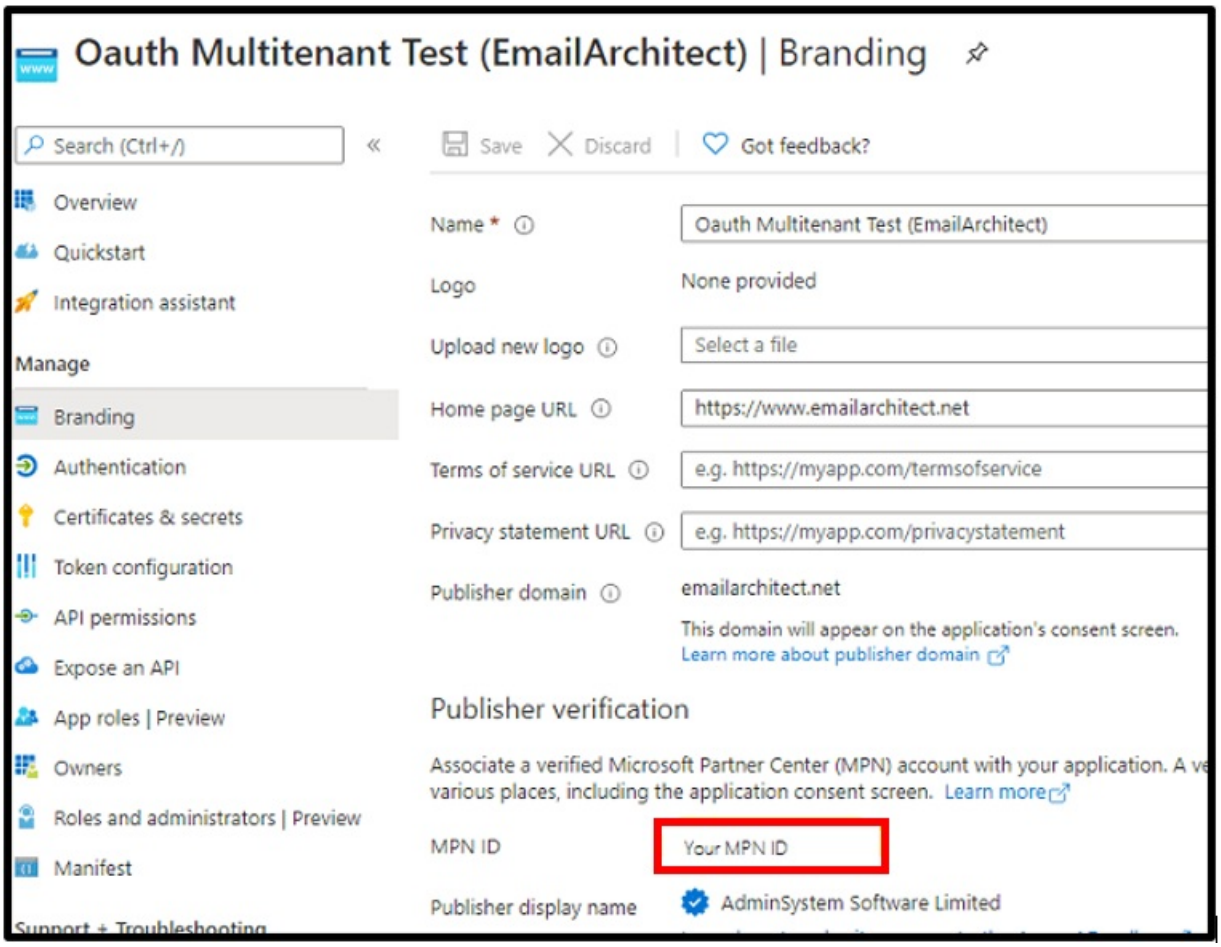
Return to Contents

Branding and verify publisher

Only do this setup if you are using multitenant. Now we click Branding, you can edit your company logo, URL and application name. If your application supports multitenant (access user in all Office 365 and Microsoft personal account), you must complete the publisher verification.

If the application only accesses the accounts in your organization, you can skip publisher verification.

It is not difficult, you can have a look at publisher verification. After publisher verification is completed, your branding is like this:

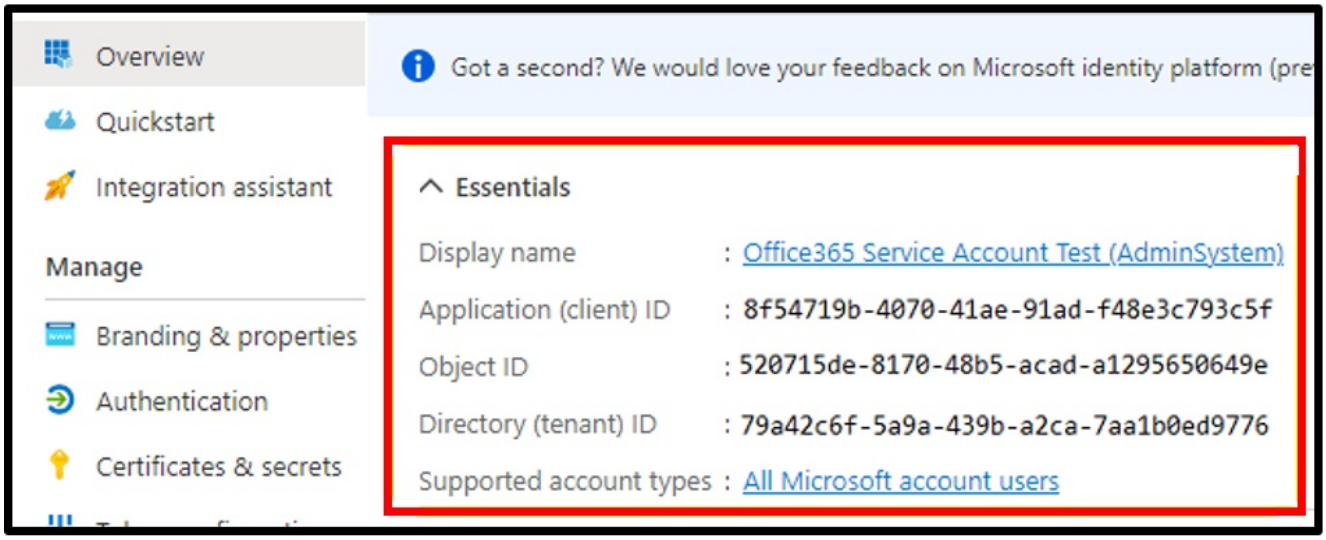


You must complete the publisher verification for multi-tenant application, otherwise, your application will not request access token correctly.

Return to Contents

Client ID and tenant

Now you can click Overview to find your ClientID and tenant.



Return to Contents

Grant admin consent

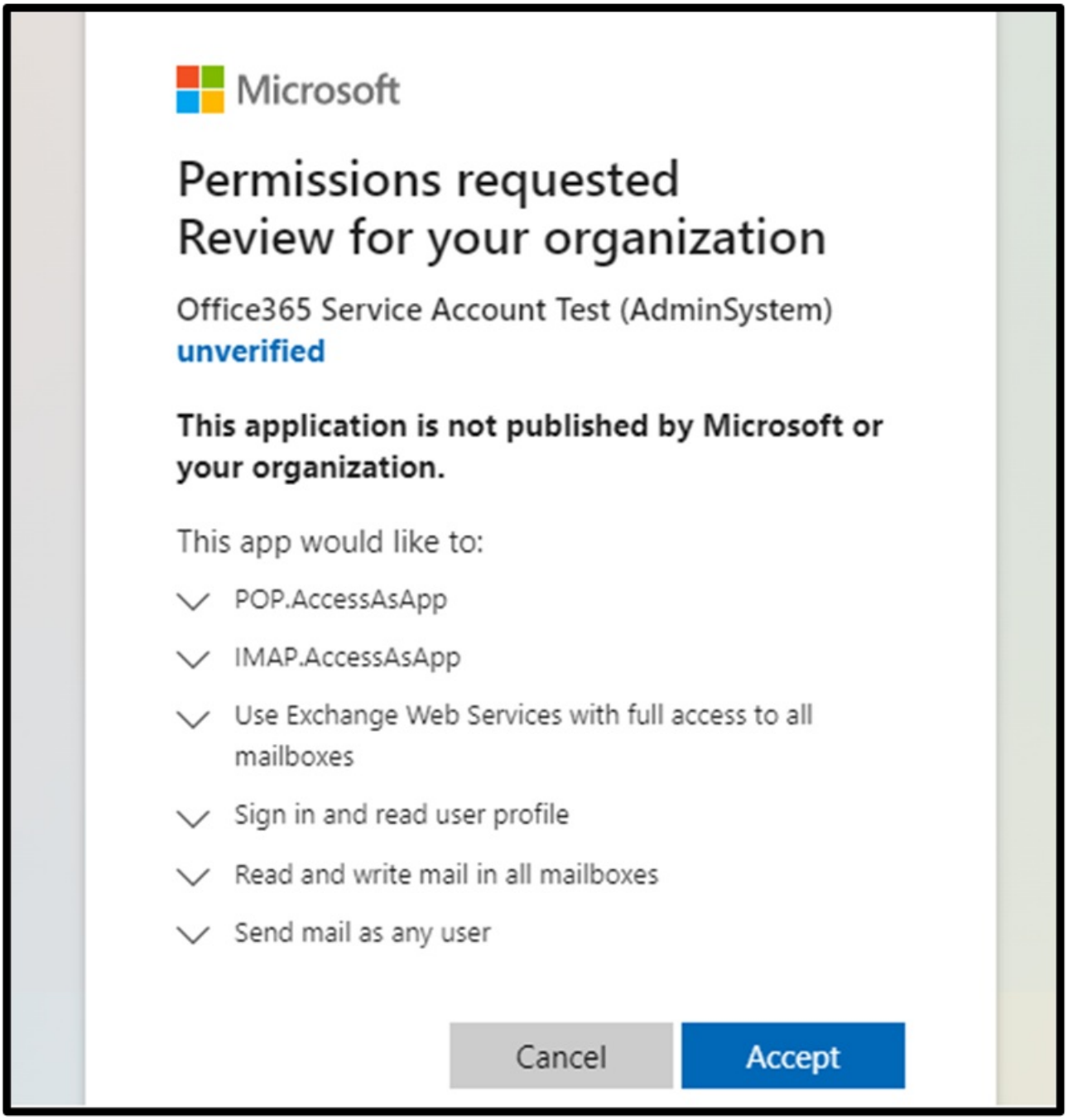
To use your application to access the user mailbox in Office365 domain, you should get admin consent by Office365 domain administrator.

- If you created the application and you're the Office365 administrator:
  In API Permission -> "Click grant admin consent for ..." to grant consent to the application.

- If you created the application and you're not the Office365 administrator:
  send the link to Office365 administrator, please change client_id to yours

https://login.microsoftonline.com/common/adminconsent?client_id=8f54719b-4070-41ae-91ad-f48e3c793xyzf&state=12345&redirect_uri=https://login.microsoftonline.com/common/oauth2/nativeclient

- Administrator can open above link in web browser, if administrator agrees with the permissions the application requires, grant consent. If not, click cancel or close the window.



- Administrators can change/cancel the permissions by Signing in to the Azure Portal -> Select Azure Active Directory then Enterprise applications.
- After the administrator grants consent, the web browser will redirect to the following URL, and send tenant value to the application developer.

https://login.microsoftonline.com/common/oauth2/nativeclient?admin_consent=True&tenant=79a42c6f-5a9a-439b-a2ca-7aa1b0ed97xyz&state=12345

After the administrator authorized the permissions, you can use the application to access any user's mailbox in Office365 domain by EWS or Graph API.

Return to Contents

Grant consent on behalf of a specific user

Instead of granting consent for an entire organization, an admin can also use the Microsoft Graph API to grant consent to delegated permissions on behalf of a single user. For a detailed example that uses Microsoft Graph PowerShell, see Grant consent on behalf of a single user by using PowerShell

Return to Contents

Limit user access to an application

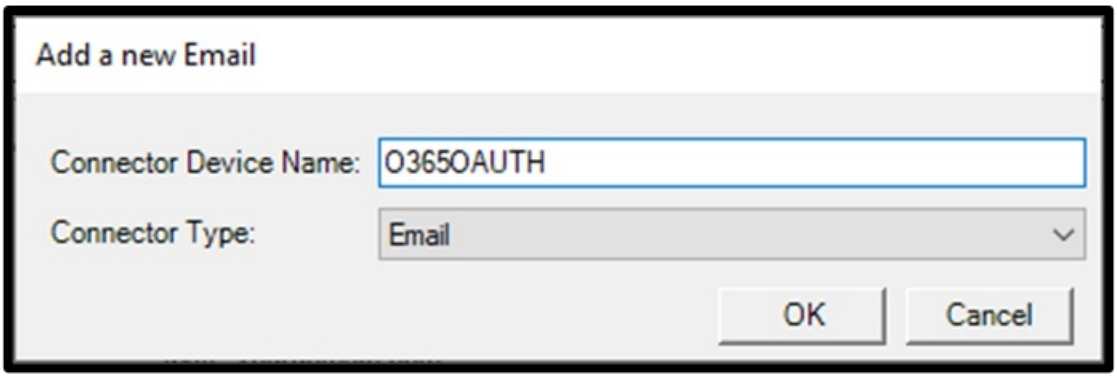
User access to applications can still be limited, even when tenant-wide admin consent has been granted. Configure the application's properties to require user assignment to limit user access to the application. For more information, see Methods for assigning users and groups.

For a broader overview, including how to handle other complex scenarios, see Use Azure AD for application access management.

Return to Contents

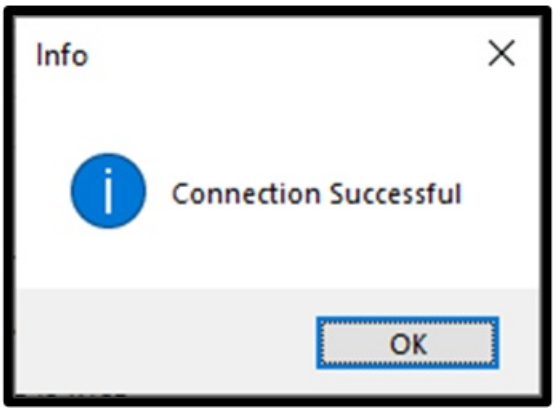
Create email connector

In MediaGateway, navigate to Universal Connector -> Connector -> Add. Enter a Connector Device Name and select Email for the Connector Type. Click OK.



For the Format Type, choose O365 OAUTH. Email User will be the email that has an active inbox that you want to monitor. Tenant Id will be the tenant Id from your app registration. Client Id will be the client Id from your app registration. Client Secret will be the client secret from your app registration.

Test your connector by clicking the Test button. A successful test will look like this.

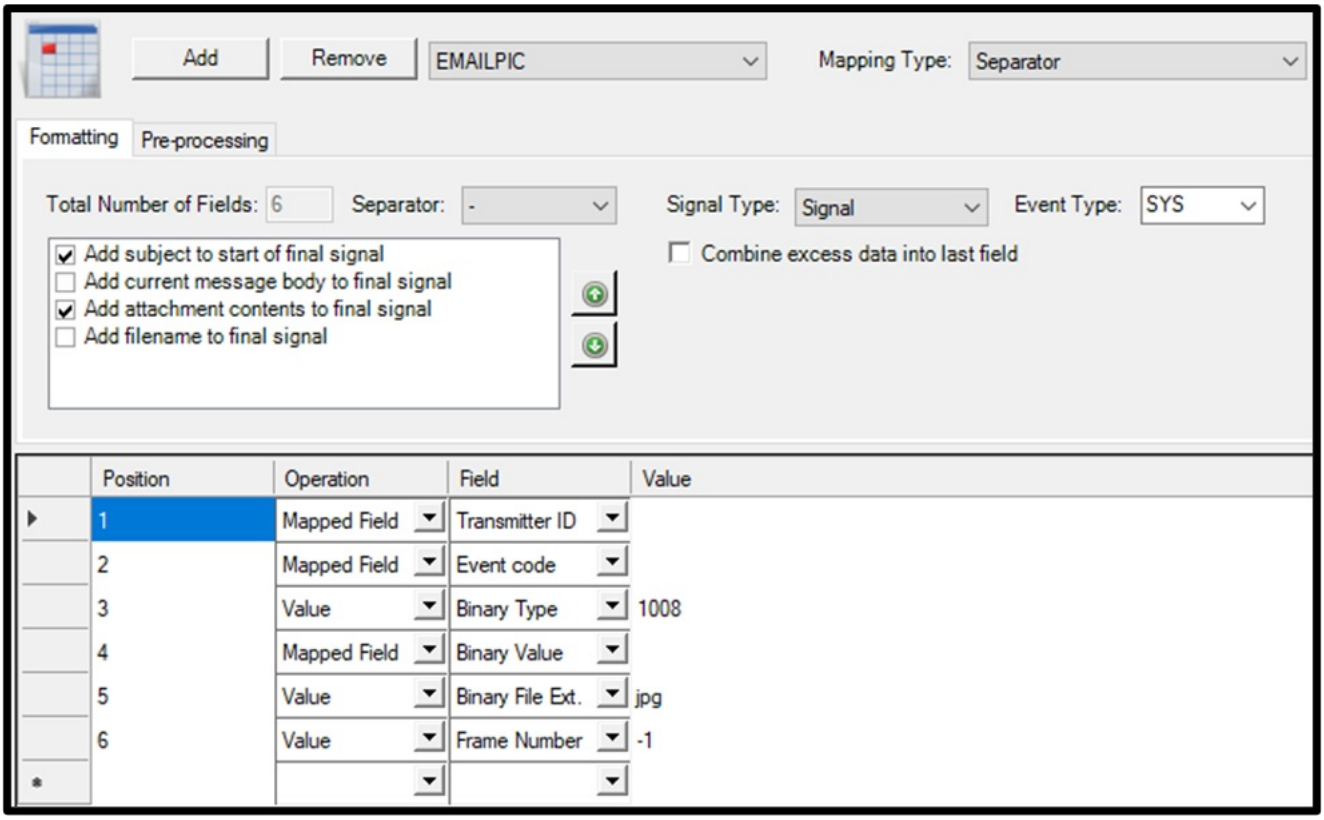


Click Update then File->Save.

Return to Contents

Create data map

Create a Data Map to parse/process your data. UniversalConnector->Data Map

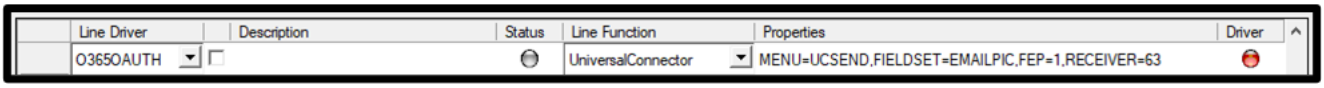


File>Save.

Return to Contents

Create line driver

Create a line driver for the new connector.



File>Save.

Return to Contents