



# Configuring Office 365 for Publisher

**Manitou®**

**Contents**

Configuring Office 365 for Publisher..... 2

    Office 365 Oauth 2.0 Client Credentials Grant ..... 2

    Create Your Application In Azure Portal ..... 2

    Single Tenant and Multitenant In Account Type..... 3

    API Permissions ..... 3

    Client ID and Client Secrets..... 4

    Branding and Verify Publisher ..... 5

    Client ID and Tenant..... 6

    Grant Admin Consent..... 6

    Grant Consent on Behalf of a Specific User..... 7

    Limit User Access to an Application ..... 8

    Configuring the Service Provider Device for the Publisher ..... 8

## Configuring Office 365 for Publisher

Publisher requires that EasendMail.exe is installed on the Publisher server. Contact support for the exe file to install.

## Office 365 OAuth 2.0 Client Credentials Grant

Normal OAuth requires user input user/password for authentication. It is not suitable for a 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 an Office365 user, it does not work for a personal Hotmail account.

## 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 does not require that an application owner is administrator in your Office365 domain. However, your Office365 administrator must authorize the application to access user mailbox.**

- Sign into 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.

**Register an application**

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

My test application

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (emailarchitect.net 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, Xb

☐ Personal Microsoft accounts only

To access user in other organizations or Microsoft personal account, you need to verify publisher later

## 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 account (hotmail.com, outlook.com), please select Multitenant type, and you must verify publisher.

Because we just need to support Office365 user in our organization, select **Accounts in this organizational directory only (single tenant)**.

Do not select **Personal Microsoft accounts only**, because there is no way to access Microsoft personal account in background service.

**If you do not verify publisher for a multitenant application, your application will not request an access token successfully.**

## 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 my organization uses > Office 365 Exchange Online > Application Permission > Other permission > full\_access\_as\_app, IMAP.AccessAsApp and POP.AccessAsApp

### Request API permissions

Select an API

Microsoft APIs

**APIs my organization uses**

My APIs

Apps in your directory that expose APIs are shown below

🔍 Start typing an API name or Application ID

Here is the permissions list:

## Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. [Learn more about permissions and consent](#)

[+](#) Add a permission [✓](#) Grant admin consent for appmail

API / Permissions name	Type	Description
▼ Microsoft Graph (3)		
Mail.ReadWrite	Application	Read and write mail in all mailboxes
Mail.Send	Application	Send mail as any user
User.Read	Delegated	Sign in and read user profile
▼ Office 365 Exchange Online (3)		
full_access_as_app	Application	Use Exchange Web Services with full access to all mailboxes
IMAP.AccessAsApp	Application	IMAP.AccessAsApp
POP.AccessAsApp	Application	POP.AccessAsApp

## Client ID and Client Secrets

Create a client secret for the application. Click Certificates and secrets > Client secrets and add a new client secret.

### Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

[+](#) New client secret

Description	Expires	Value	ID
Verified Test	12/31/2299	SOZ*****	1f867467-7

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

**Please store the client secret value, because it is hidden when you view it the next time.**

## Branding and Verify Publisher

Only do this setup if you are using multitenant.

Click Branding; you can edit your company logo, URL, and application name. If your application supports multitenant (access users in all Office 365 and Microsoft personal accounts), you must complete the publisher verification.

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

You can review publisher verification. After publisher verification is completed, your branding is like this:

**Oauth Multitenant Test (EmailArchitect) | Branding**

Search (Ctrl+/) Save Discard Got feedback?

Overview Quickstart Integration assistant

Manage

Branding

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles | Preview

Owners

Roles and administrators | Preview

Manifest

Support + Troubleshooting

Name \* Oauth Multitenant Test (EmailArchitect)

Logo None provided

Upload new logo Select a file

Home page URL https://www.emailarchitect.net

Terms of service URL e.g. https://myapp.com/termsofservice

Privacy statement URL e.g. https://myapp.com/privacystatement

Publisher domain emailarchitect.net

This domain will appear on the application's consent screen. [Learn more about publisher domain](#)

**Publisher verification**

Associate a verified Microsoft Partner Center (MPN) account with your application. A verified account is required for applications that access users in all Office 365 and Microsoft personal accounts. [Learn more](#)

MPN ID Your MPN ID

Publisher display name AdminSystem Software Limited

**You must complete the publisher verification for multitenant application, otherwise, your application will not request an access token correctly.**

## Client ID and Tenant

Click Overview to find your client ID and tenant.

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Got a second? We would love your feedback on Microsoft identity platform (pre

^ Essentials

Display name : [Office365 Service Account Test \(AdminSystem\)](#)

Application (client) ID : [REDACTED]

Object ID : [REDACTED]

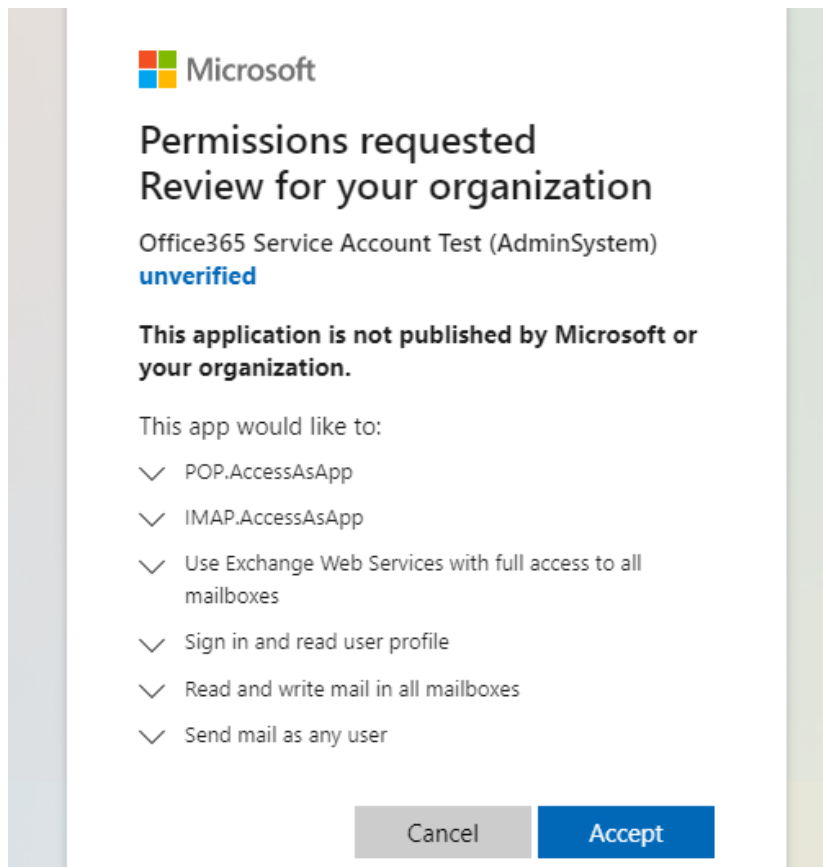
Directory (tenant) ID : [REDACTED]

Supported account types : [All Microsoft account users](#)

## Grant Admin Consent

To use your application to access user mailboxes in an Office365 domain, you should get administrator consent by an Office365 domain administrator.

- If you created the application and you are the Office365 administrator:  
In API Permission > "Click grant admin consent for ..." to grant consent to the application.
- If you created the application and you are not the Office365 administrator:  
Send the link to an Office365 administrator, please change client\_id to yours  
  
[https://login.microsoftonline.com/common/adminconsent?client\\_id=\[REDACTED\]&state=12345&redirect\\_uri=https://login.microsoftonline.com/common/oauth2/nativeclient](https://login.microsoftonline.com/common/adminconsent?client_id=[REDACTED]&state=12345&redirect_uri=https://login.microsoftonline.com/common/oauth2/nativeclient)
- Administrators can open the above link in a web browser. If the administrator agrees with the permissions the application requires, the administrator grants consent. If not, click cancel or close the window.



- Administrators can change/cancel the permissions by using the Sign into the Azure Portal > Select Azure Active Directory, then Enterprise applications.
- After an administrator has granted consent, the web browser will redirect to the following URL. Send the tenant value to the application developer.

[https://login.microsoftonline.com/common/oauth2/nativeclient?admin\\_consent=True&tenant=\[REDACTED\]&state=12345](https://login.microsoftonline.com/common/oauth2/nativeclient?admin_consent=True&tenant=[REDACTED]&state=12345)

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

## Grant Consent on Behalf of a Specific User

Instead of granting consent for an entire organization, an administrator 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](#).



## Limit User Access to an Application

User access to applications can still be limited, even when tenant-wide administrator 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](#).

## Configuring the Service Provider Device for the Publisher

Service Provider Devices

External Device: MAPI

Description: EMAIL

Protocol: Email

Address:

ID: AUTH\_0365

☐ Dial-up Service

Port Settings: 587

Account Name: [redacted]@boldgroup.com **User**

Password: [masked] **Client Secret**

Timeout 1: 0

Timeout 2: 0

Timeout 3: 0

Timeout 4: 0

Timeout 5: 0

Parameter 1: 0 Value 1: Client ID 093fe

Parameter 2: 0 Value 2: Tenant ID 106

Parameter 3: 0 Value 3:

- The ID will need to be set as AUTH\_0365.
- The Account Name will be the email that has an active inbox that you want to monitor.
- Password will be the client secret from your app registration.
- Value 1: will be the client Id from your app registration.
- Value 2: will be the tenant Id from your app registration.

**① When using Office365 you can only send emails from the Office365 domain. Office365 will not allow the "Send From" to be an address that is not within the domain.**