

O365 Client Credentials flow - Email Authentication

Contents

- [Contents](#)
- [Introduction](#)
- [For Azure administrators](#)
 - [Application creation](#)
 - [Application registration and permissions](#)
 - [Application secret](#)
- [For ECS organization administrators/implementers](#)
 - [Configure a mailbox](#)
- [Migrating from On behalf Of to Client Credentials / Hybrid application](#)

Introduction

As Microsoft has started to turn of Basic authentication for email protocols for tenants in favor of Modern Authentication (OAuth2) we have implemented one form of this for ECS email using an interactive flow, see also: [O365 On Behalf Of - Email Authentication](#)

This was done, because it was the quickest way for us to prevent users being unable to use email anymore, but it is not the optimal solution. We now have implemented the client credentials flow, which requires no user interaction and uses Microsoft Graph for email handling instead of STMP and IMAP.

The benefit for administrators/users is that there only needs to be one client secret provided in the Azure App which then needs to be configured in ECS once and then **all the tenant's mailboxes** can be used in ECS by just adding them to the ECS configuration. If there needs to be a restriction on allowed mailboxes, this can be controlled entirely by the Azure admin. For more information about this, please see: [Limiting application permissions to specific Exchange Online mailboxes](#)

Also, the expiration is controllable on Azure side and can be set to a maximum of two years (at the time of writing), meaning that when it works, the ECS configuration only needs to be updated once every two years when a new secret needs to be issued. In comparison, the interactive flow needs a quarterly update, involving user interaction and (most likely) a MFA input - per mailbox!



All mailboxes are accessible

We emphasize again that all the tenant's mailboxes are available by default, you probably want to limit access in Azure to only the mailboxes you actually need in ECS.

For more information about this, please see: [Limiting application permissions to specific Exchange Online mailboxes](#)

For Azure administrators

In [Azure Active Directory](#) you need to have an application that supports mail using MS Graph with a client secret available.

Application creation

If you do not have the application set up yet, create one (else, proceed to add permissions or adding secrets as needed):

Azure Active Directory admin center

Dashboard > Enterprise applications

Enterprise applications | All applications

Evolve IP - ECS Development - Azure Active Directory

+ New application Refresh Download (Export) Preview info Columns Preview features Got feedback

Overview

View, filter, and search applications in your organization that are set up to use your Azure AD tenant as their identity Provider.

The list of applications that are maintained by your organization are in application registrations.

Manage

Search by application name or object ID Application type -- Enterprise Applications Application ID starts with Add filters

10 applications found

Name	Object ID	Application ID	Homepage URL	Created on
MT Demo too (Server mail)	32f33cf1-1f46-4985-a8d3-2f53aeca2d8	fc335209-6aff-4aea-88b0-ec05e70be76		12/6/2022
TestReg	46a8f99b-5d7d-4810-9f81-bcf3346f53e	a83452c5-21d3-4e72-9d15-0f0c55e84e42	https://account.activedirectory.windowsazure.com/444/...	2/15/2023
ECS Test app	86bab196-178e-465a-a36f-9f696531e62a	9e0f26ef-7683-49e5-8899-e0b650ca8539		6/15/2022
Postman	882a3dc4-f524-459d-8309-98f6e546bce	ef3df4e3-d894-429f-bb0d-b7bb55969c25		6/21/2022
Tutorial Sample App	965b172a-6fa9-4a99-8321-7b0dd8debe8	6731de76-14a6-49aa-97bc-6eba6914391e		10/3/2022
MT Demo	ae67b24f-09f0-40dd-a88c-7277e549b5f	91e54e2d-eb7e-456f-ab89-16316e207741		12/6/2022
Graph Explorer	b5ab5260-64ea-4396-b14f-69046c467b28	de6bc8b5-d9f9-48b1-a6ad-b740da725064	https://developer.microsoft.com/graph/graph-explorer	6/14/2022
ECS O365 Mail Test On Behalf Of	ca0e91c8-be94-44f2-a108-a8dce194b20f	9098be66-bcb4-49fa-b992-9b1f117efc27	https://account.activedirectory.windowsazure.com/444/...	10/3/2022
O365 On Behalf Of	d3e63072-35a1-4eed-b724-fc10080c9d21	7e598f54-35a3-42c3-a9fc-1aa006761acf	https://account.activedirectory.windowsazure.com/444/...	11/10/2022
ECS O365 Mail Test - App	de9db444-1e58-4e2a-a799-6044cda05040	0bc6009-a764-4d32-8d83-e024aaed8ba	https://account.activedirectory.windowsazure.com/444/...	10/3/2022

1. Go to "Enterprise applications"
2. Select "+ New application"

Azure Active Directory admin center

Dashboard > Enterprise applications | All applications >

Browse Azure AD Gallery

+ Create your own application Got feedback?

The Azure AD App Gallery is a catalog of thousands of apps that make it easy to deploy and configure single sign-on (SSO) and automated user management. If you have an app that you want to share with other organizations to discover and use, you can file a request using the process described in [this article](#).

Search application Single Sign-on : All User Account Management : All Categories : All

Cloud platforms

Amazon Web Services (AWS)

Google Cloud Platform

On

1. Then select "+ Create your own application"

Create your own application ✕

Got feedback?

If you are developing your own application, using Application Proxy, or want to integrate an application that is not in the gallery, you can create your own application here.

What's the name of your app?

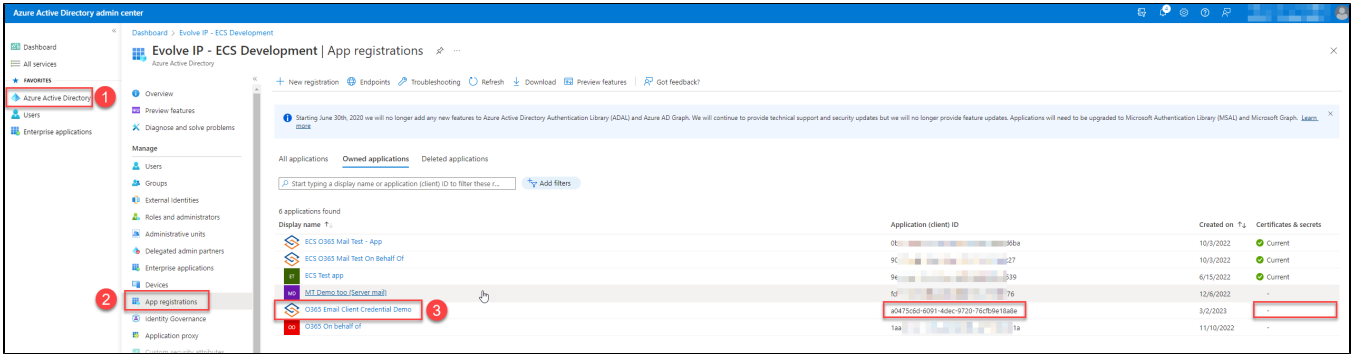
What are you looking to do with your application?

Configure Application Proxy for secure remote access to an on-premises application
 Register an application to integrate with Azure AD (App you're developing)
 Integrate any other application you don't find in the gallery (Non-gallery)

Give it a logical name so you or others can easily identify it in the future and select the correct application type, for us it is "Integrate any other...".

Application registration and permissions

Now go to the application registration page.

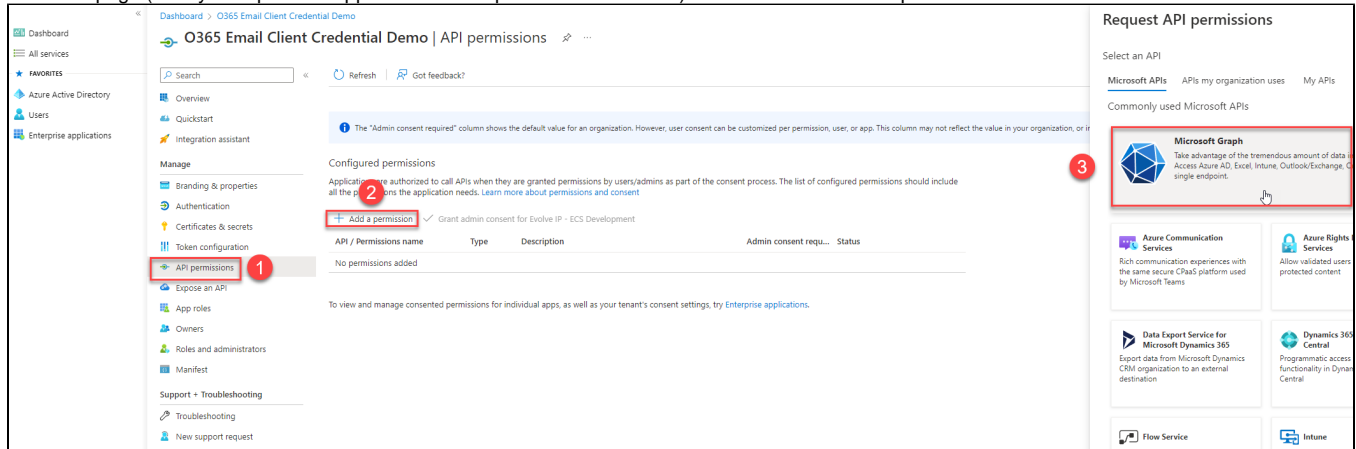


1. Select "Azure Active Directory" in the main menu
2. Select "App registrations" in the sub menu
3. As you can see here, the application does not have a secret yet. (far right column)
Now select the application by clicking the name, this will bring you to the application overview page.

Application id

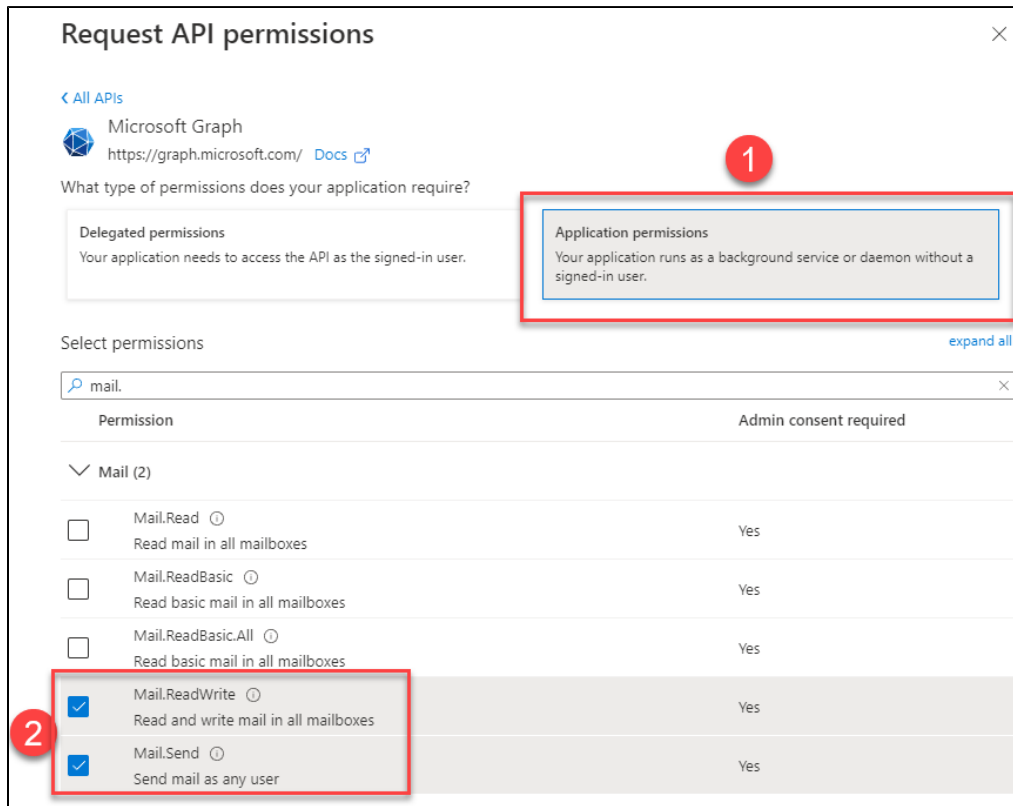
Also take note of the application id, this is part of the configuration we need. You can copy it now for later use. or from the application overview page which might be easier since it has copy buttons next to the id.
In our example this is "a0475c6d-6091-4dec-9720-76cfb9e18a8e"

From this page (after you copied the application id and pasted it somewhere) we will add the needed permissions:



1. Select "API permissions"
2. Select "+ Add a permission"
3. Select "Microsoft Graph"

On the API sidebar:



1. Select "Application permissions"

Search
For the next step note that you can do a partially search for "mail." in the search box as show in the screenshot.

2. Select both:
 - a. Mail.ReadWrite
 - b. Mail.Send

Dashboard > O365 Email Client Credential Demo

O365 Email Client Credential Demo | API permissions

Search Refresh Got feedback?

You are editing permission(s) to your application, users will have to consent even if they've already done so previously.

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value.

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission ✓ Grant admin consent for Evolve IP - ECS Development

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (2)				
Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	⚠ Not granted for Evolve I...
Mail.Send	Application	Send mail as any user	Yes	⚠ Not granted for Evolve I...

To view and manage consented permissions for individual apps, as well as your tenant's consent settings, try [Enterprise applications](#).

When you confirm your permissions you should see they are added, but not have not been granted access yet. You can grant access here by clicking on the "Grant admin consent for..." title.

Application secret

We now are ready to create a secret we can use in ECS to get access to the mailboxes.

Expiration

If the secret is about to expire or is already expired, return here to create a new secret. You will of course need to update ECS configuration afterwards as well in order to use the new key.

Azure Active Directory admin center

Dashboard > O365 Email Client Credential Demo

O365 Email Client Credential Demo | Certificates & secrets

Search Got feedback?

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (0) Federated credentials (0)

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	Secret ID
No client secrets have been created for this application.			

1. In the submenu select "Certificates and secrets"
2. Select "+ New client secret"

On the client secret sidebar:

Add a client secret

Description: ECS access for Dev Org

Expires: Custom

Start: Recommended: 180 days (6 months)

End: 90 days (3 months)

365 days (12 months)

545 days (18 months)

730 days (24 months)

Custom

- Give it a logical description so you or others can easily identify it in the future
- Select the desired expiration, using the "custom " option you can also set a start date if desired

When the secret is generated you will see its value on the page.

! Only chance to copy

Be aware that this is the **only** chance to copy the secret's value, if you navigate away and return it is no longer visible or copyable and you will need to generate a new secret!

Azure Active Directory admin center

Dashboard > O365 Email Client Credential Demo

O365 Email Client Credential Demo | Certificates & secrets

Search

Got feedback?

Got a second to give us some feedback? →

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (1)** Federated credentials (0)

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	Secret ID
ECS access for Dev Org	3/1/2025	bmg...rbww...5570aa7...ac4e	

Client secret values cannot be viewed, except for immediately after creation. Be sure to save the secret when created before leaving the page.

Make sure to copy the secret

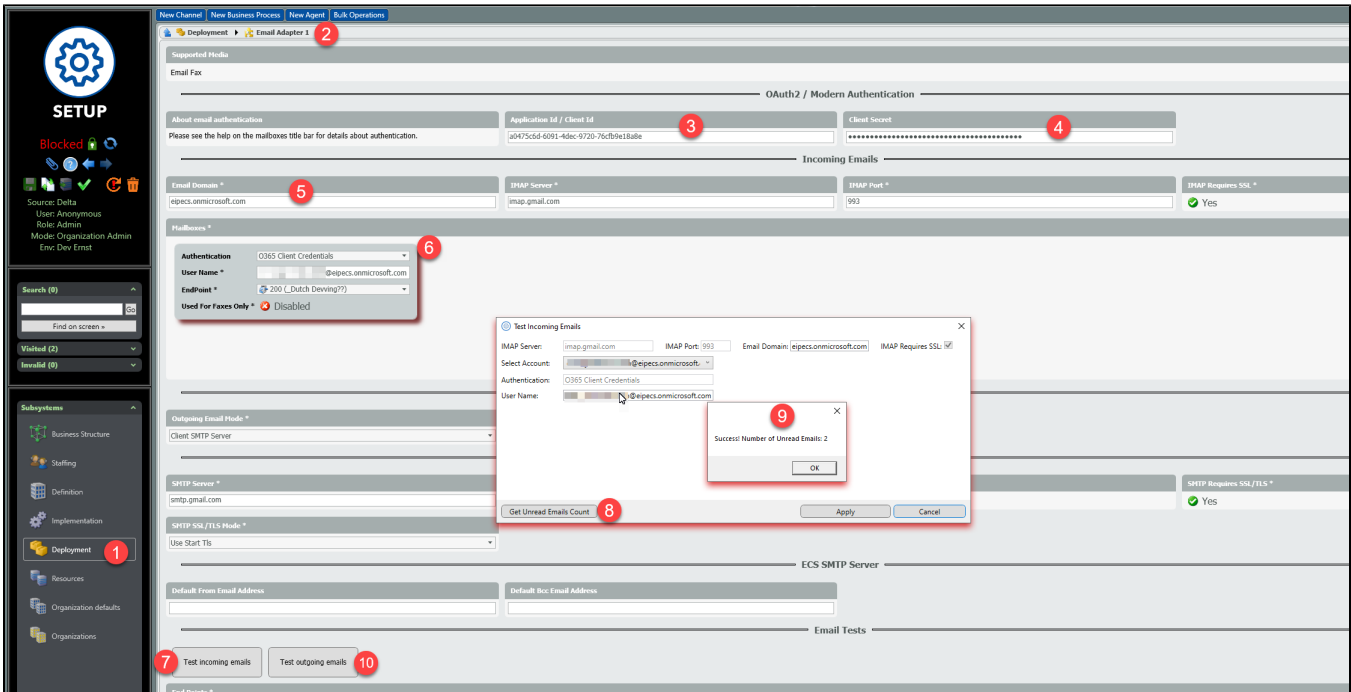
- Use the copy button to copy the secret's **value** - the ID is not relevant.
- Paste it somewhere for later reference

For ECS organization administrators/implementers

When the Azure part is set up correctly, the mailboxes can be configured.

Configure a mailbox

This step is much simpler than the interactive flow, open the organization's setup and then navigate to the email adapter:



1. Select "Deployment"
2. Select the email adapter (selection itself not shown in the screenshot, only the result, but it is under deployment - software services - email adapter)
3. Fill out the application id of your Azure application, in our example it is "a0475c6d-6091-4dec-9720-76cfb9e18a8e"
4. Paste the secret of your Azure application. In case you forgot to copy it earlier, you will need to create a new secret.
5. Fill out the domain, this is used as tenant identifier for logging in in Azure, which should normally match the domain of your user.

Other users

Note that you can have other users, but in that case they need to have been imported into your Azure Active Directory

6. Create a mailbox using "O365 Client Credentials" and the username of the mailbox you want to use
7. Select the "Test incoming emails" button
8. Select "Get unread emails count"
9. If configured correctly, you should see this window.
Note that it can show 0 (zero), that depends on actual unread emails in the mailbox, but that will still mean the configuration works
10. Optionally you can also send out a test email

If the configuration works you can optionally add more mailboxes and then deploy the configuration.

Migrating from On behalf Of to Client Credentials / Hybrid application

If you already are using the interactive flow, it should be fairly easy to migrate/upgrade to the client credentials flow. Since the application is already present, you can add the permissions and secrets as described above and as long as you do not remove the original API permissions you are able to use both types in ECS.

A hybrid configuration on Azure would look something like this:

The screenshot shows the 'API permissions' page in the Azure portal. A warning banner at the top indicates that editing permissions requires user consent. The 'Configured permissions' section contains a table with the following data:

API / Permissions name	Type	Description	Admin consent requ...	Status
1 IMAP.AccessAsUser.All	Delegated	Read and write access to mailboxes via IMAP.	No	Granted for Evolve IP - E...
2 Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	Granted for Evolve IP - E...
3 Mail.Send	Application	Send mail as any user	Yes	Granted for Evolve IP - E...
4 offline_access	Delegated	Maintain access to data you have given it access to	No	Granted for Evolve IP - E...
5 SMTP.Send	Delegated	Send emails from mailboxes using SMTP AUTH.	No	Granted for Evolve IP - E...

- The permissions 1, 4 & 5 are set for the "on behalf of" interactive flow as can be seen by the Type "Delegated".
- Permissions 2 & 3 are for the client credentials flow, Type "Application".
- ⚠️ Note the warning above, on changes it may be necessary for users to give consent again, meaning acquiring a new token and following the popup guidelines

In ECS setup, a hybrid configuration would look like:

The screenshot shows the 'SETUP' screen for an 'Email Adapter 1'. It is divided into 'Supported Media' and 'Incoming Emails' sections. Under 'Incoming Emails', there are two mailbox configurations:

- Mailbox A:** Authentication is set to 'O365 On Behalf Of'. It shows a 'Token acquired, click to re-test' button and a 'Remove token (log out)' button.
- Mailbox B:** Authentication is set to 'O365 Client Credentials'. It shows a 'User Name' field and an 'EndPoint' dropdown.

Red circles 1 and 2 highlight the 'Application Id / Client Id' and 'Client Secret' fields in the 'OAuth2 / Modern Authentication' section.

- The application id (1) remains the same.
- The new client secret is added (2).
- Mailbox A is using a token, but note that if all settings are correct we could just change authentication type.
- Mailbox B is using the client credentials flow.

