

Configuring a SMTP OAuth Connection

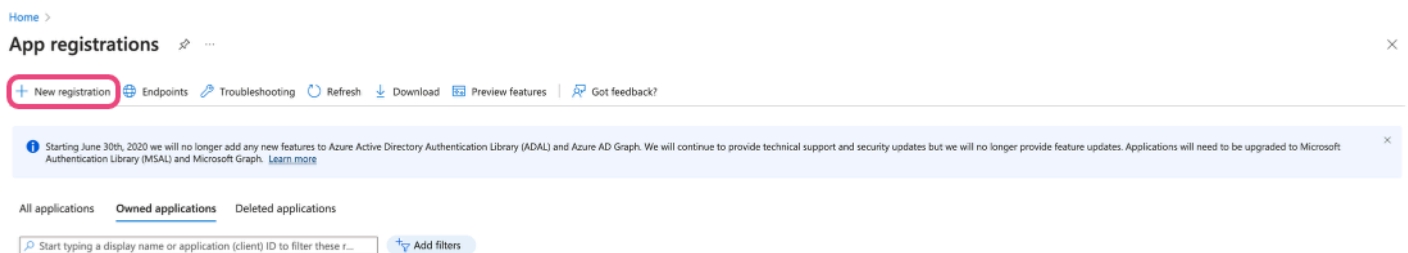
Configuring the App Registration



This step must be done by someone with access to the client's Azure configuration.

1. Go to Azure App registrations and Click “+New Registration”

Go to Azure App registrations at https://portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationsListBlade and click “+New Registration” to open the App registrations page.



2. Setup a Single Tenant Account



Name and redirect URI can be anything but a redirect URI must be entered.

Register an application ...

* Name

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

Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory only (onitplatform 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, Xbox)
- ☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Select a platform ▼

e.g. https://example.com/auth

3. Click “Add a certificate or a secret” to Add a New Secret

You should be taken to the main page for the app. Once on the homepage, find "Client credentials: Add a certificate or a secret" and open the link to add a new secret.

Home > App registrations >

Documentation

Search

Delete Endpoints Preview features

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

Manifest

Support + Troubleshooting

Troubleshooting

New support request

Essentials

Display name : Documentation

Application (client) ID : 351d5cb2-1c69-4081-9cc2-a769d71567cb

Object ID : 768a35ab-7d58-44b9-8185-05631f4c8862

Directory (tenant) ID : d9f2fc08-d549-4fcc-bae2-2266c314d3cd

Supported account types : My organization only

Client credentials : [Add a certificate or secret](#)

Redirect URIs : [1 web, 0 spa, 0 public client](#)

Application ID URI : [Add an Application ID URI](#)

Managed application in L : [Documentation](#)

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. [Learn more](#)

Get Started

Documentation

Build your application with the Microsoft identity platform

The Microsoft identity platform is an authentication service, open-source libraries, and application management tools. You can create modern, standards-based authentication solutions, access and protect APIs, and add sign-in for your users and customers. [Learn more](#)

Call APIs

Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

[View API permissions](#)

Sign in users in 5 minutes

Use our SDKs to sign in users and call APIs in a few steps. Use the quickstarts to start a web app, mobile app, SPA, or daemon app.

[View all quickstart guides](#)

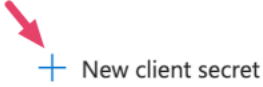
Configure for your organization

Assign users and groups, apply conditional access policies, configure single sign-on, and more in Enterprise applications.

[Go to Enterprise applications](#)

4. Click “+ New client secret”


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




Description	Expires	Value ⓘ	Secret ID
-------------	---------	---------	-----------



No client secrets have been created for this application.

a. Provide a brief description of where this secret will be used and select an expiration length.

 A longer expiration is better if the client does not want to update the secret frequently.

b. Once the secret is created, copy the value and secret ID.

 This will be your only chance to copy the value so keep it somewhere secure.

+ New client secret				
Description	Expires	Value ⓘ	Secret ID	
Documentation	4/16/2025	WvR8Q~dburnMyAJOayfq9DUm61D~M...	0667485f-2d81-4cef-97f8-723b3fec0642	 

5. Setup API Permissions

Home > Documentation

Documentation | API permissions

Refresh Got feedback?

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

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 [more](#)

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 onitplatform

API / Permissions name	Type	Description	Admin consent requ...	Status
▼ Microsoft Graph (1)				
User.Read	Delegated	Sign in and read user profile	No	...

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

- a. Click "+ Add a permission" and add the following:
 - Microsoft Graph > Delegated Permissions > SMTP.Send (<https://graph.microsoft.com/SMTP.Send>)
 - Microsoft Graph > Delegated Permissions > offline_access
- b. Grant admin consent by clicking "Grant admin consent for ...". After consent is successfully granted, a green check mark and "Granted for onitplatform" will appear under the Status column (shown below).

API / Permissions name	Type	Description	Admin consent requ...	Status	
Microsoft Graph (2)					
SMTP.Send	Delegated	Send emails from mailboxes using SMTP AUTH.	No	 Granted for onitplatform	...

6. Copy Important Information

Grab the following information for use later:


- a. Directory (tenant) ID
- b. Client Secret value
- c. Application (client) ID
- d. Redirect-URI

Generating the Refresh Token

For this stage it is best to follow the steps mentioned in Microsoft's documentation: <https://learn.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-auth-code-flow>.

Below is a copy of Microsoft's instructions.

1. Requesting an Authorization Code


 This step must be done using the account that is tied to the SMTP OAuth configuration.

- a. Copy the URL below and paste it into the browser.

https://login.microsoftonline.com/{tenant}/oauth2/v2.0/authorize?client_id={client_id}&response_type=code&redirect_uri=http%3A%2F%2Flocalhost%2Fmyapp%2F&response_mode=query&scope=https%3A%2F%2Fgraph.microsoft.com%2FSMTP.Send%2Foffline_access

- b. Add in the correct values from the previous step (6) for "tenant" and "client_id" (highlighted below) into the URL that you just pasted into the browser.

```
https://login.microsoftonline.com/{tenant}/oauth2/v2.0/authorize?client_id={client_id}&response_type=code&redirect_uri=http%3A%2F%2Flocalhost%2Fmyapp%2F&response_mode=query&scope=https%3A%2F%2Fgraph.microsoft.com%2Fmail.send&state=12345
```


-  tenant = Directory (tenant) ID. Valid tenant values are common, organizations, consumers, and tenant identifiers.
- client_id = Application (client) ID. An example of a client_id is: 6731de76-14a6-49ae-97bc-6eba6914391e.

When navigated to in the browser the user will be prompted to login. Upon successful completion of this step the user will be redirected and the new url will look similar to the image below.

c. Copy the “code” value from this url to use in the next step (2b).

```
http://localhost?code=AwABAAAAPM1KaPlrEqdFSBzjqfTGBCmLdgfSTLEMPGYuNHSUYBrq&state=12345
```

2. Request an Access Token with a Client Secret

-  You will need to use Postman or a similar tool for these steps. The client_id, scope, code, redirect_url, grant_type, and client_secret values will need to be added to the Body of the request.

a. In Postman, setup a new request as follows:

POST /{tenant}/oauth2/v2.0/token HTTP/1.1

Host: <https://login.microsoftonline.com>

Content-Type: application/x-www-form-urlencoded

client_id=6731de76-14a6-49ae-97bc-6eba6914391e

&scope= user.read offline_access

&code=AwABAAAAPM1KaPlrEqdFSBzjqfTGBCmLdgfSTLEMPGYuNHSUYBrq

&redirect_uri=http%3A%2F%2Flocalhost%2Fmyapp%2F

&grant_type=authorization_code

&client_secret=JqQX2PNo9bpM0uEihUPzyrh



client_secret is only required for web apps. This secret needs to be URL-Encoded.

b. Update the tenant, client_id, redirect_uri, and client_secret values with what you have setup in Azure. The “code” will be the value returned in the previous step (1c).

c. Copy the refresh_token from the successful request. We'll be using it to finish our setup in the next step (2l).

```
"access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Ij5HVEZ2ZEZEstZnl0aEV1Q...",
```

```
"token_type": "Bearer",
```

```
"expires_in": 3599,
```

```
"scope": "user.read offline_access",
```

```
"refresh_token": "AwABAAAAvPM1KaPlrEqdFSBzjqfTGAMxZGUTdM0t4B4..."
```

```
"id_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJub25lIn0.eyJhdWQiOiIyZDRkMTFhMi1mODE0LTQ2YTctOD...",
```

Setting up the SMTP OAuth Connection in Onit

Now that we have all of our information we can do the setup in Onit.

1. Navigate to Auth Providers and Create a New SMTP OAuth Provider

The screenshot shows the Onit application wizard interface. On the left, the 'Auth Provider' section is expanded, showing a list of providers including 'MS SMTP OAuth2'. The main panel displays the configuration form for 'MS SMTP OAuth2'. The form includes fields for Name, Display Name, Address, Port, Domain, User Name (Email), Tenant, Application (client) ID, Client secret, Scope, Redirect-URI, and Refresh Token (Required On Create). The 'Tenant' field is a dropdown menu.

2. Fill Out the Values as Follows:

- a. Name: Custom
- b. Display Name: Custom
- c. Address: smtp.office365.com
- d. Port: 587
- e. Domain: the domain of the exchange setup. For example, Onit's email domain is "onit"
- f. User Name (email): the email of the account that was used for the setup in the previous steps
- g. Tenant: Organizations
- h. Application (client) ID: Value from the Azure app
- i. Client Secret: Value from the Azure app
- j. Scope: <https://graph.microsoft.com/SMTP.Send>
- k. Redirect URI: Value from the Azure app
- l. Refresh Token: The refresh token value from the Postman response

3. Ensure that the "Enabled" Checkbox is Checked and then Save

That's all! Now you can test to see if emails are going out properly. Please note that the sent emails page cannot track successful delivery of emails when an SMTP provider is configured.