

How to set up IMAP and POP3 authentication using OAuth 2.0.

Starting in October 2022, Microsoft began to [disable basic authentication](#) in Exchange Online for POP3, IMAP and other protocols. In order for applications to continue to access Microsoft 365 (formerly Office 365) via these protocols, they must switch to modern authentication (OAuth 2.0).

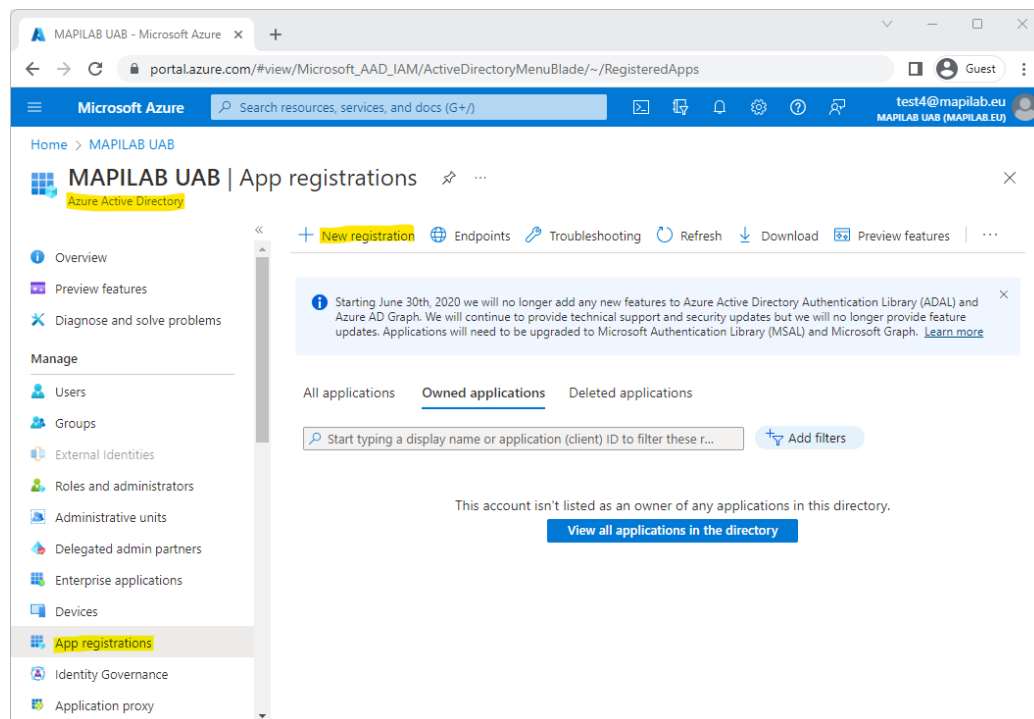
Before the application can access mailboxes, it must be registered and configured in Azure Active Directory and Microsoft 365. Below we show what needs to be done to access mailboxes via POP3 and IMAP using modern authentication.

1. Register an application.

Log in to [Azure Portal](#) or [Azure Active Directory admin center](#).

On the left navigation panel, click **Azure Active Directory**.

Navigate to **App registration** and click **New registration**.



Give your application a name and click **Register**.

Register an application

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

HarePoint Workflow Extensions ✓

Supported account types
Who can use this application or access this API?

Accounts in this organizational directory only (MAPILAB UAB 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

By proceeding, you agree to the Microsoft Platform Policies

Register

Once your application is registered, note the **Application (client) ID** and **Directory (tenant) ID**.

HarePoint Workflow Extensions

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

Got a second? We would love your feedback on Microsoft identity platform (previously Azure AD for developer).

Essentials

Display name
HarePoint Workflow Extensions

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

Application (client) ID
f40f8a8e-79757db581c9

Redirect URIs
[Add a Redirect URI](#)

Object ID
b42464d3-aa1632e42942

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

Directory (tenant) ID
d98f9541-0b6358d220ac

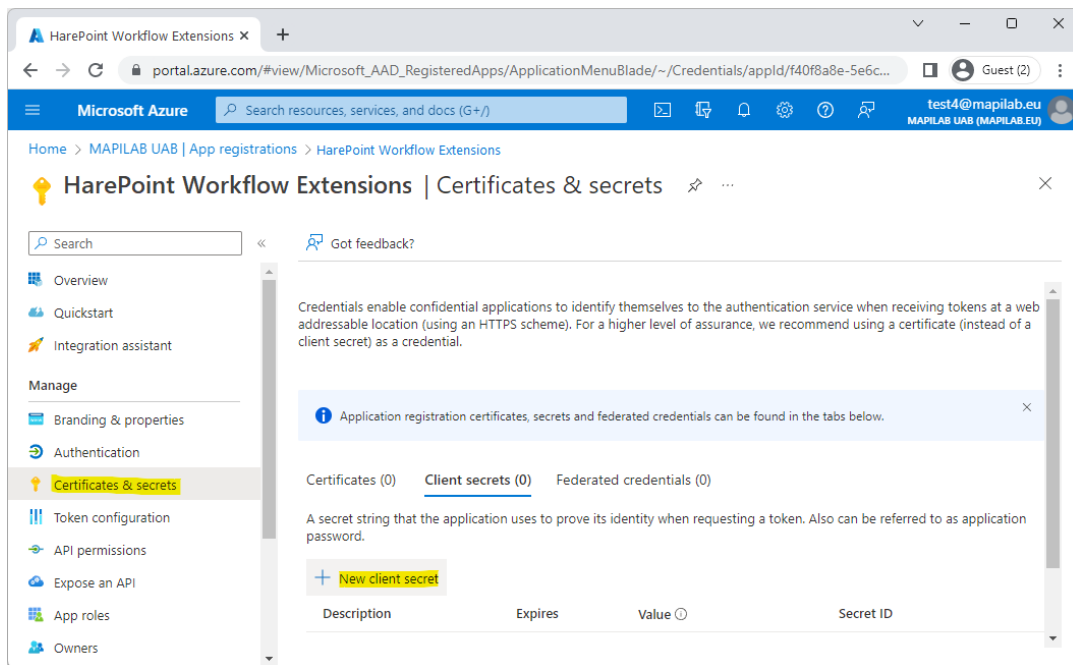
Managed application in local directory
[HarePoint Workflow Extensions](#)

Supported account types
[My organization only](#)

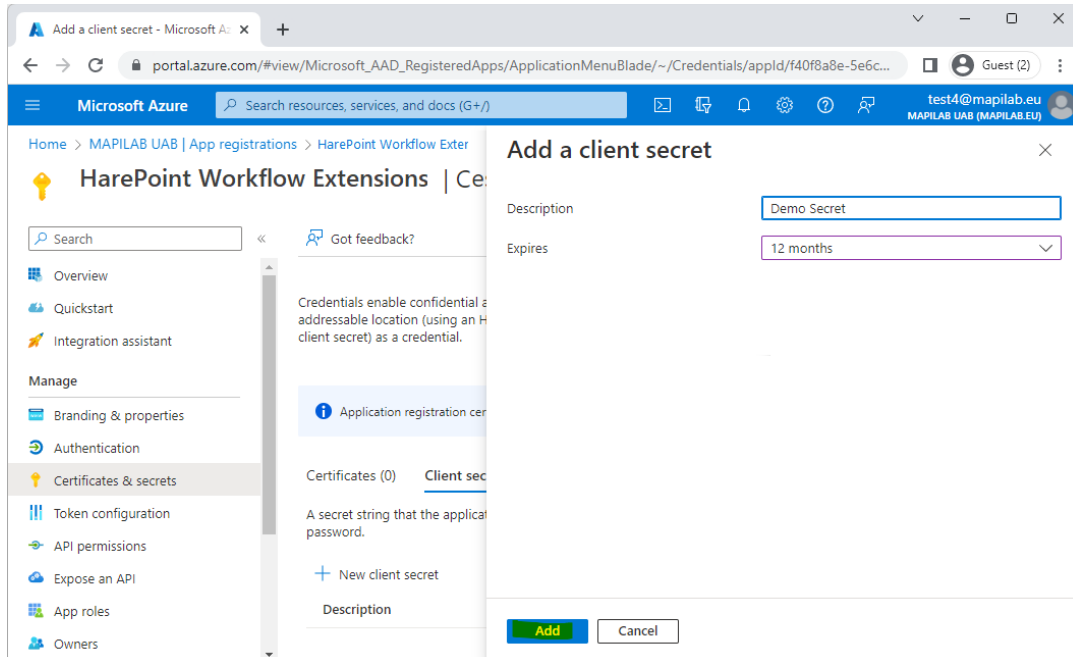
Get Started Documentation

2. Create client secret

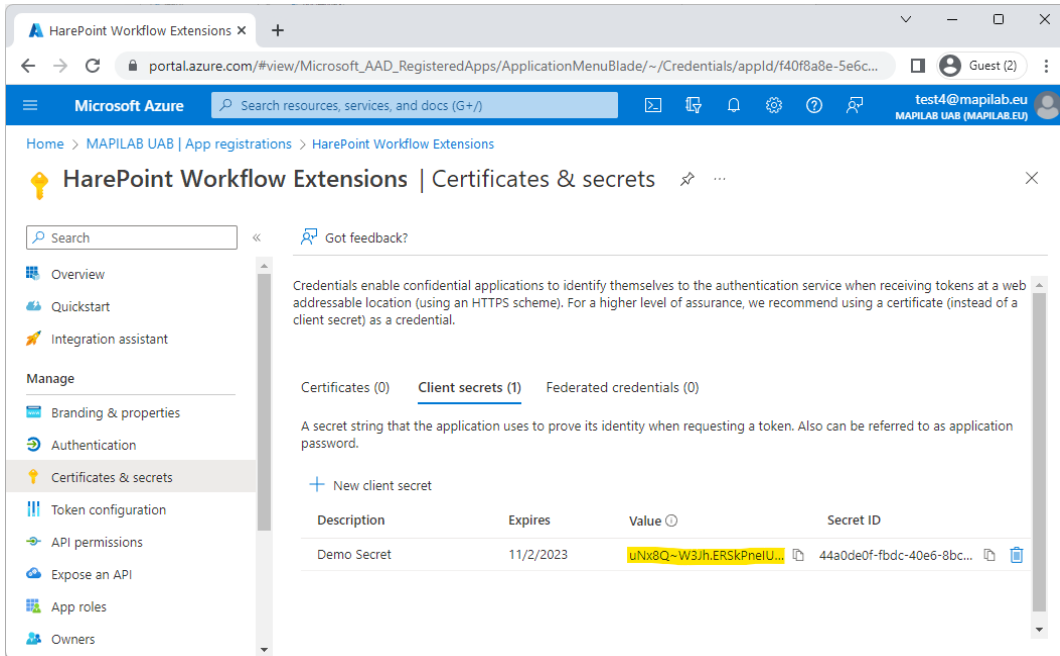
In the left menu, select **Certificates & secrets** and then click **New client secret**.



Give the secret a name, specify an expiry period and click **Add**.

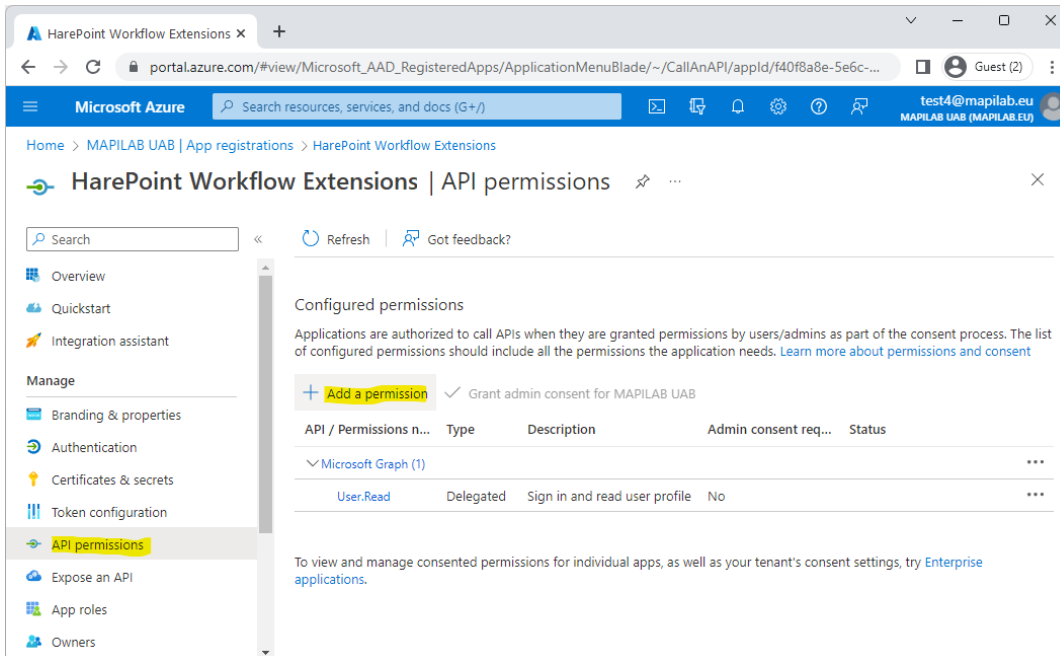


Immediately copy and save the Value (not the Secret ID) of the secret. Client secret values cannot be viewed, except for immediately after creation.

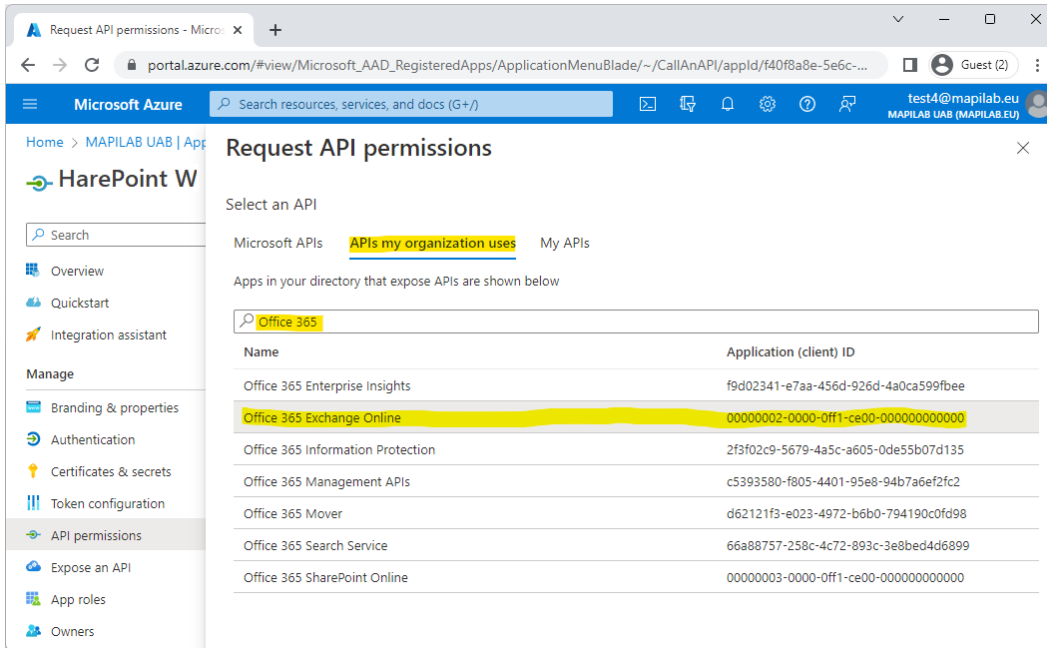


3. Configure app API permissions

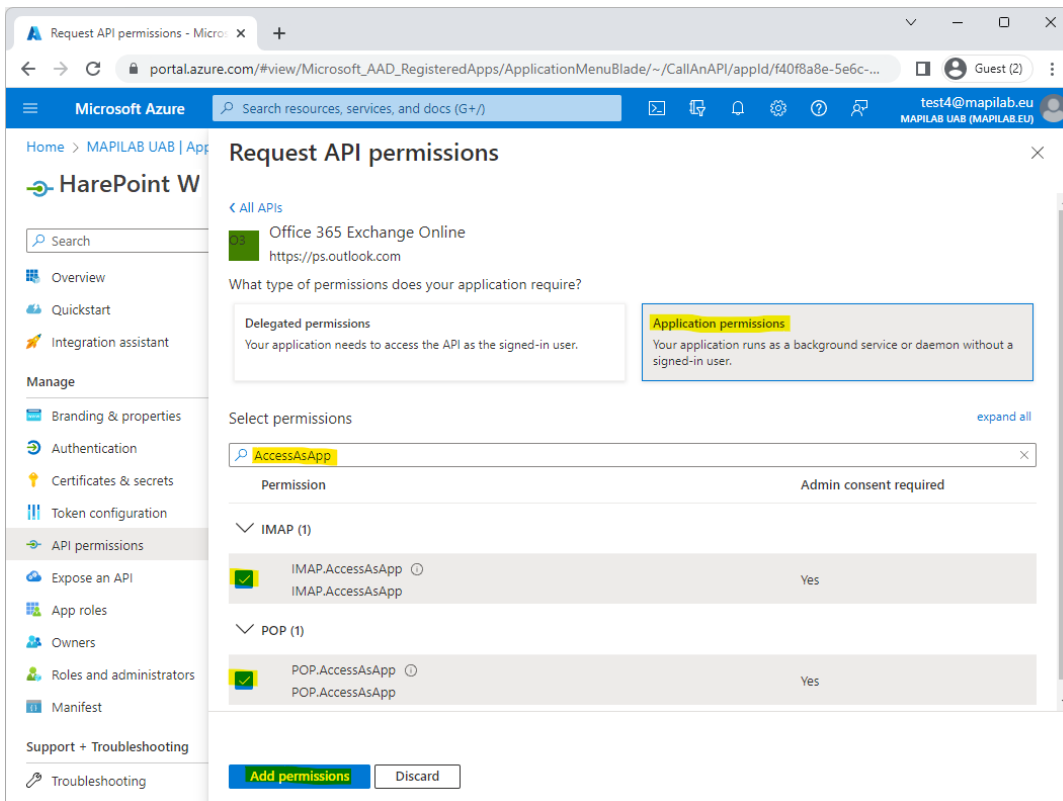
In the left menu, select **API permissions** and then click **Add a permission**.



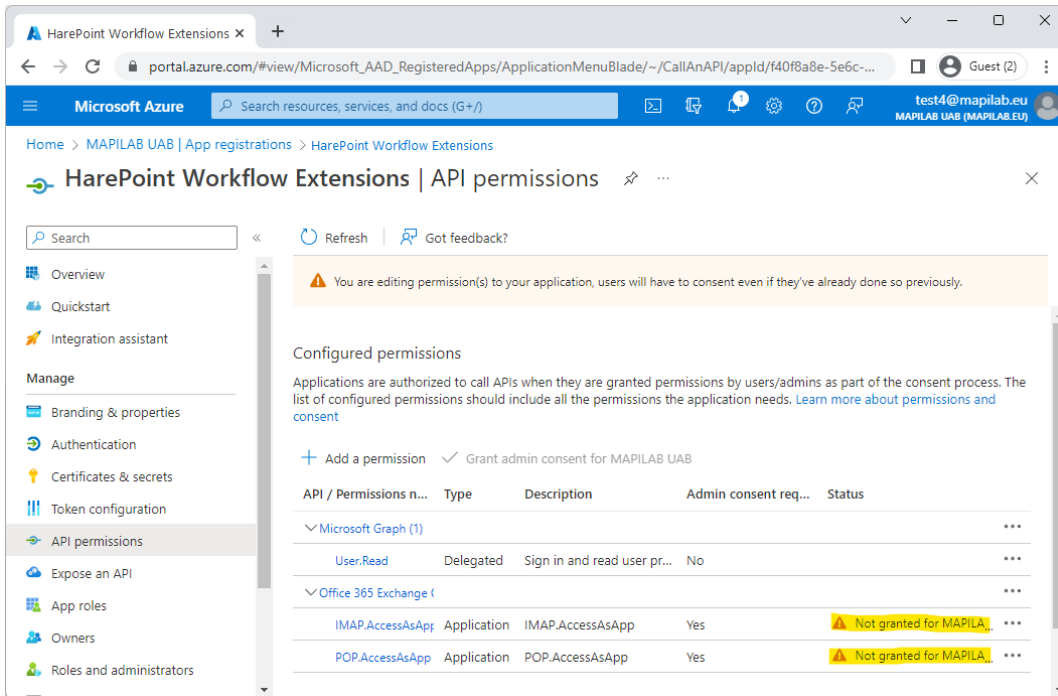
Switch to **APIs my organization uses** tab, type **Office 365** in the search box and click **Office 365 Exchange Online** entry.



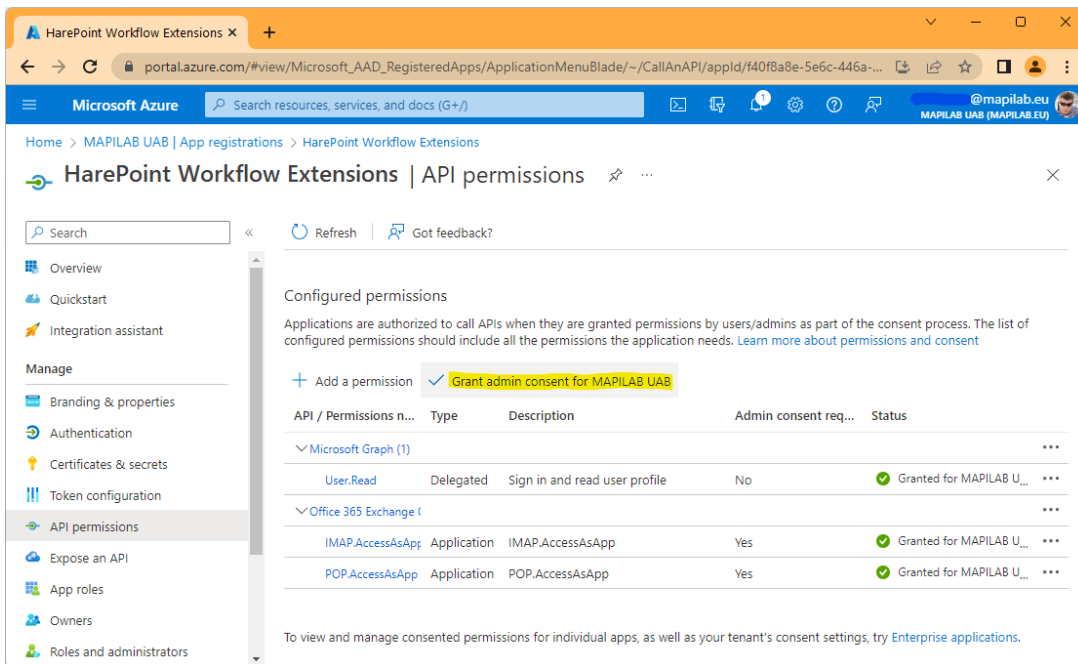
Select **Application permissions**, type **AccessAsApp** in the search box, check **IMAP.AccessAsApp** and **POP.AccessAsApp** and click **Add permissions**.



The permissions you have just added must be approved by your organization's administrator.



In the example shown in the screenshot above, the application and permissions were created by a user without an administrative role. In this case, he/she should ask the administrator to **grant consent** to your application for the specified permissions.



Optionally, you can remove the *delegated* **User.Read** permission, which is not needed for this application.

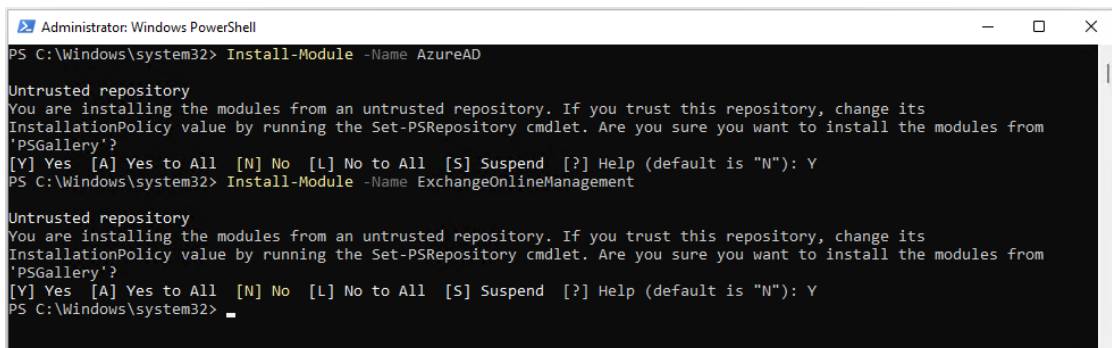
4. Add mailbox access permissions

Assigning mailbox access permissions is done using PowerShell.

Open your **PowerShell as Administrator**.

If you have not yet installed the **AzureAD** and **ExchangeOnlineManagement** modules, install them.

```
Install-Module -Name AzureAD
Install-Module -Name ExchangeOnlineManagement
```



```
Administrator: Windows PowerShell
PS C:\Windows\system32> Install-Module -Name AzureAD

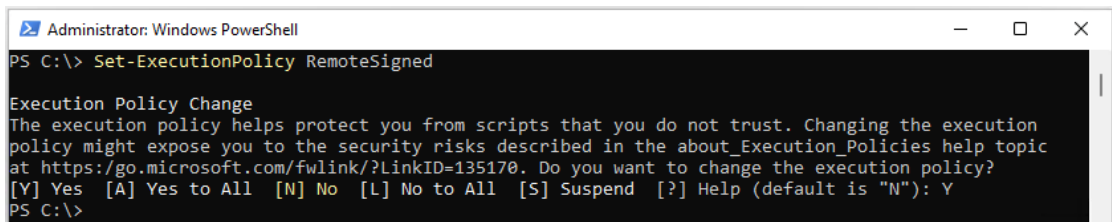
Untrusted repository
You are installing the modules from an untrusted repository. If you trust this repository, change its
InstallationPolicy value by running the Set-PSRepository cmdlet. Are you sure you want to install the modules from
'PSGallery'?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"): Y
PS C:\Windows\system32> Install-Module -Name ExchangeOnlineManagement

Untrusted repository
You are installing the modules from an untrusted repository. If you trust this repository, change its
InstallationPolicy value by running the Set-PSRepository cmdlet. Are you sure you want to install the modules from
'PSGallery'?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"): Y
PS C:\Windows\system32> _
```

(Wondering why these modules are installed from untrusted repository? See [this Azure-PowerShell issue](#))

In addition, you may need to set an execution policy.

```
Set-ExecutionPolicy RemoteSigned
```



```
Administrator: Windows PowerShell
PS C:\> Set-ExecutionPolicy RemoteSigned

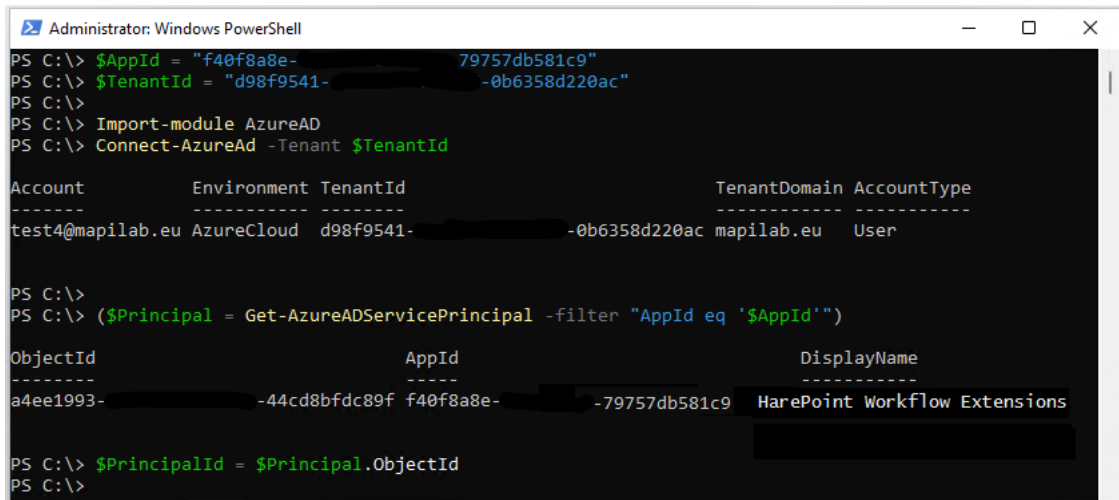
Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution
policy might expose you to the security risks described in the about_Execution_Policies help topic
at https://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"): Y
PS C:\>
```

Get the **service principal ID** associated with your application. You will be prompted to log in to your Azure account. You can use a non-administrator account at this step.

```
$AppId = "YOUR_APP_ID_HERE"
$TenantId = "YOUR_TENANT_ID_HERE"

Import-Module AzureAD
Connect-AzureAd -Tenant $TenantId

($Principal = Get-AzureADServicePrincipal -filter "AppId eq '$AppId'")
$PrincipalId = $Principal.ObjectId
```



```
Administrator: Windows PowerShell
PS C:\> $AppId = "f40f8a8e-79757db581c9"
PS C:\> $TenantId = "d98f9541-0b6358d220ac"
PS C:\>
PS C:\> Import-Module AzureAD
PS C:\> Connect-AzureAd -Tenant $TenantId

Account          Environment TenantId          TenantDomain AccountType
-----
test4@mapilab.eu AzureCloud d98f9541-0b6358d220ac mapilab.eu User

PS C:\>
PS C:\> ($Principal = Get-AzureADServicePrincipal -filter "AppId eq '$AppId'")

ObjectId          AppId          DisplayName
-----
a4ee1993-44cd8bfdc89f f40f8a8e-79757db581c9 HarePoint Workflow Extensions

PS C:\> $PrincipalId = $Principal.ObjectId
PS C:\>
```

Create the service principal for your application. You will be prompted to log in to your Exchange Online account. You must use an administrator account at this point.

```
$DisplayName = "Principal for HarePoint"

Import-Module ExchangeOnlineManagement
Connect-ExchangeOnline -Organization $TenantId

New-ServicePrincipal -AppId $AppId -ServiceId $PrincipalId -DisplayName $DisplayName
```



```

Windows PowerShell
PS C:\> $DisplayName = "Principal for HarePoint"
PS C:\>
PS C:\> Import-module ExchangeOnlineManagement
PS C:\> Connect-ExchangeOnline -Organization $TenantId

-----
This V3 EXO PowerShell module contains new REST API backed Exchange Online cmdlets which doesn't require WinRM for Client-Server communication. You can now run these cmdlets after turning off WinRM Basic Auth in your client machine thus making it more secure.

Unlike the EXO* prefixed cmdlets, the cmdlets in this module support full functional parity with the RPS (V1) cmdlets.

V3 cmdlets in the downloaded module are resilient to transient failures, handling retries and throttling errors inherently.

However, REST backed EOP and SCC cmdlets are not available yet. To use those, you will need to enable WinRM Basic Auth.

For more information check https://aka.ms/exov3-module
-----
PS C:\>
PS C:\> New-ServicePrincipal -AppId $AppId -ServiceId $PrincipalId -DisplayName $DisplayName

DisplayName                ServiceId                AppId
-----
Principal for HarePoint    a4ee1993-                -44cd8bfdc89f    f40f8a8e-                -79757db581c9
PS C:\>

```

Add **FullAccess** permissions to all mailboxes that you want to access from your application.

```

Add-MailboxPermission -User $PrincipalId -AccessRights FullAccess -Identity "test4@mapilab.eu"
Add-MailboxPermission -User $PrincipalId -AccessRights FullAccess -Identity "test5@mapilab.eu"

```

```

Windows PowerShell
PS C:\> Add-MailboxPermission -User $PrincipalId -AccessRights FullAccess -Identity "test5@mapilab.eu"

Identity      User                AccessRights                IsInherited Deny
-----
test5         S-1-5-21-89837200... {FullAccess}                False        False
PS C:\>

```

You have now registered your application to access Office 365 mailboxes via IMAP and POP3 and received its **Application (client) ID**, **Client secret** and **Directory (tenant) ID**.

These strings will be used to authenticate to Microsoft 365 via OAuth 2.0 and obtain an OAuth token. This token is then used to authenticate to Exchange Online using IMAP and POP3.

5. Configuring the HarePoint Workflow Extensions action to download mail from Exchange Online

In order to use OAuth 2.0 authentication for the selected mailbox, you must specify a login in the form **email@clientID@tenantID**.

The login will look like this:

```
test4@mapilab.eu@f40f8a8e-1234-5678-9abc-79757db581c9@d98f9541-1234-5678-9abc-0b6358d220ac
```

Specify **client secret** as a password.

Use **SSL** as the connection type.

Receive e-mail from outlook.office365.com, secure connection type **SSL**, login test4@mapilab.eu@f40f8a8e-1234-5678-9abc-79757db581c9@d98f9541-1234-5678-9abc-0b6358d220ac and password [uNx8Q~W4Jh.E](#). Store e-mail sender in [Variable: variable1](#), recipient in [Variable: variable2](#), subject in

Microsoft 365 IMAP/POP3 server: **outlook.office365.com**

IMAP port: **993**

POP3 port: **995**

Always **connect using SSL/TLS** (implicit).