**How to Use the New Exchange Online V2 PowerShell Module for Unattended Automation Scripts**

Since MFA requires users’ interaction to create Exchange session, so in order to have **non-interactive PowerShell scripts** using Modern authentication Microsoft released **Exchange Online PowerShell V2** with introduction of this new feature:

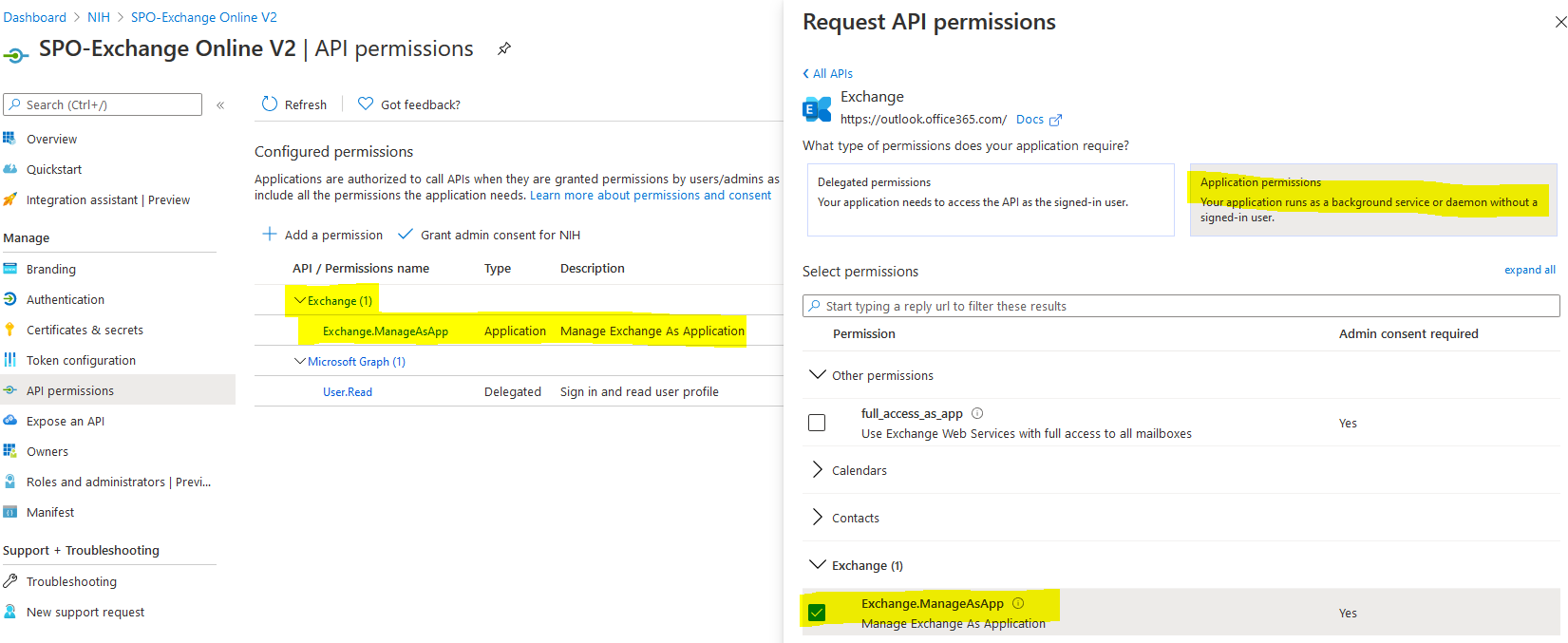
* Exclude service accounts with Conditional Access policy
* No need to store credential in the local file
* Easily automate script scheduler with Modern Auth

To automate Exchange Online PowerShell login, you need to install [EXO V2 PowerShell Module version 2.0.3](https://www.powershellgallery.com/packages/ExchangeOnlineManagement/2.0.3) or later version in which servers you run the script. This unattended script authentication uses Azure AD app, certificates and Modern auth.

Below are the steps:

* **Creating an Azure AD application with API Permissions**
  + Register the application in Azure AD at <https://portal.azure.com>
  + Assign permission to access Exchange Online

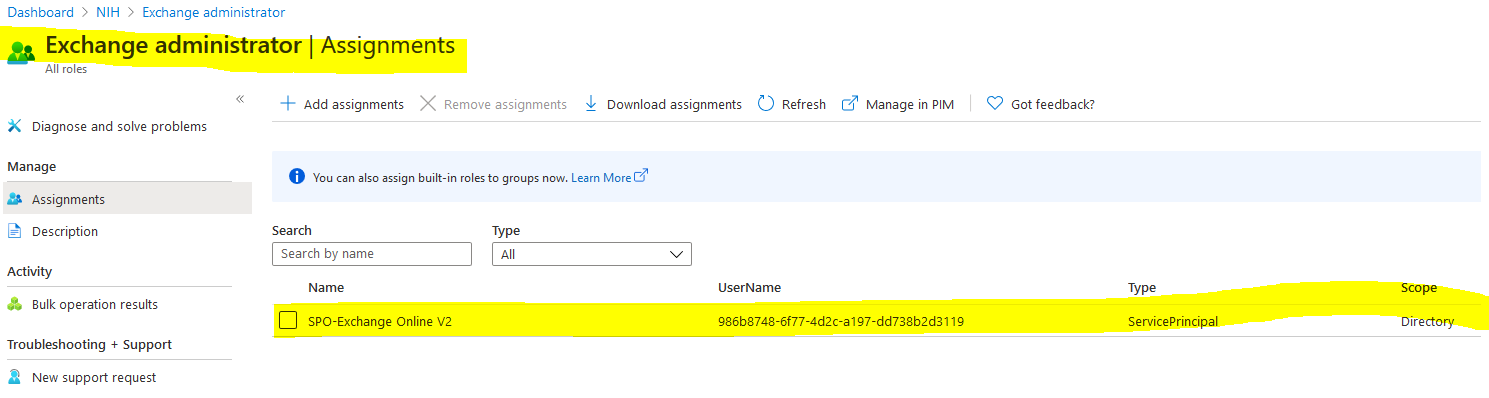
It needs to have the Application permission Exchange.ManageAsApp



* **Assigning an Azure AD Role to the Application**

The valid supported roles for Exchange Online V2 are these below.

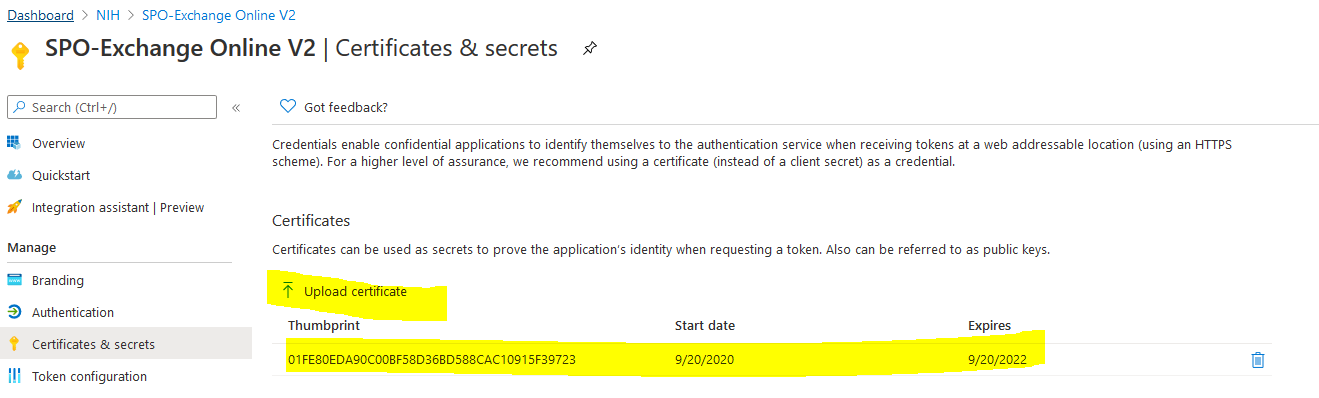
* + Company administrator
  + Compliance administrator
  + Security reader
  + Security administrator
  + Helpdesk administrator
  + Exchange Service Administrator
  + Global Reader



* **Generating and attach a Self-Signed Certificate to the Application**
  + Refer to the code D:\Scripting\O365DevOps\AdminScripts\UnitTests\Create-SelfSignedCertificate.ps1
  + Run the script below will generate a self-signed cert

.\Create-SelfSignedCertificate.ps1 -CommonName "SPO-ExchangeOnlinePowerShellV2" -StartDate 2020-09-21 -EndDate 2022-09-21

* + Attach this cert to the Azure AD app by upload self-signed cert



* **Connecting to Exchange Online PowerShell**

There are two methods to utilize the certificate credentials:

* + Using the local certificate file (.pfx)
  + Using the thumbprint of the certificate installed in **the current user’s personal certificate store**.

**Authenticating Using Local PFX Certificate**

To connect to Exchange Online PowerShell using a local certificate to authenticate, you must have the following information:

* The Directory ID or verified domain of your Azure AD tenant.
* The AppId of the application that you registered previously.
* The full file path of the self-signed PFX certificate.
* The password of the self-sign PFX certificate.

**Authenticating Using Certificate Thumbprint**

This authentication method can be considered more secure than using the local certificate with a password.

In this method, you will need to **import the certificate to the Personal certificate store**. You **only need to use the thumbprint** to identify which certificate to use for authentication.

Note that you **only need to do this step once for the current user**.

## Set the certificate file path (.pfx)

$CertificateFilePath = 'D:\Scripting\O365DevOps\AdminScripts\UnitTests\SPO-ExchangeOnlinePowerShellV2.pfx\'

## Get the PFX password

$mypwd = Get-Credential -UserName 'Enter password below' -Message 'Enter password below'

## Import the PFX certificate to the current user's personal certificate store.

Import-PfxCertificate -FilePath $CertificateFilePath -CertStoreLocation Cert:\\CurrentUser\\My -Password $mypwd.Password

* **Running Exchange Online PowerShell Scripts with App-Only Authentication**

$TenantName = "citspdev.onmicrosoft.com"

$appId = "986b8748-6f77-4d2c-a197-dd738b2d3119"

$thumprint = '01FE80EDA90C00BF58D36BD588CAC10915F39723'

Connect-ExchangeOnline -AppId $appId -Organization $TenantName -CertificateThumbprint $thumprint