ELO XC

Installation

Table of contents

Installation

System requirements

ELO XC requires an ELO server that is currently supported. You will find the version number in the *ELOxcVersion.txt* version history.

ELO XC must be installed on one of these systems:

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

These Exchange servers are supported:

- Microsoft Exchange Server 2013 SP1
- Microsoft Exchange Server 2016
- Microsoft Exchange Server 2019
- Microsoft Exchange Online (Microsoft 365)

Local authentication (on-premises) is done using basic authentication for LDAP and EWS.

Authentication with Microsoft 365 (cloud) is done using an app registration (Modern Authentication OAuth 2.0). This also requires the following installations:

- Windows Management Framework 5.1
- Microsoft PowerShell 5 (possibly Microsoft PowerShell 7 as well)
- Exchange Online PowerShell V3 module

The integrated configuration interface ELO XC Manager is supported by the following browsers:

- Current version of Microsoft Edge
- Current version of Google Chrome
- Current version of Mozilla Firefox

Up to four certificates may be required on the ELO XC system, which must be available in the system's certificate store:

Source	Meaning/Requirement
Z1 ELO server (Tomcat certificate)	SSL connections to Indexserver
Z2 EWS (Exchange certificate)	SSL connection to EWS
Z3 ELO XC	Integrated web host for ELO XC Manager
Z4 ELO XC	App registration in Microsoft 365 tenant

Installation steps

You will need to perform the following steps to test and configure your environment. For documentation purposes, transfer the values of your environment into the highlighted variables in the notes template.

- ELO server system is S1
- Tomcat certificate is Z1
- ELO XC installation system is S2 (S1=S2 is possible)
- Successful validation of the HTTPS connection from S2 to the ELO server on S1
- Import from Z1 into certificate store of S2
- Set an ELO user XCADM (as main administrator) to access XC Manager
- Exchange server certificate is Z2
- Export from Z2 and import into S2 in the *Trusted Publishers* store
- Create certificate Z3 for ELO XC on S2 and save fingerprint TP1

Exchange local	**Exchange M365**
Import from Z2 into certificate store of S2 and check Autodiscover URL with Edge	Install PowerShell and Exchange Online PowerShell V3 3.0.0 (or higher) module in S2
Create ELO XC service account XCSRV in domain AD1, save account XCACC and password XCPWD	Create certificate Z4, export for registration, and save fingerprint TP2
Ensure XCSRV has read access to AD via LDAP	Create app registration for ELO XC in M365, save TNNT tenant and XCAPP app ID
Assign ApplicationImpersonation role for XCSRV in EAC	Authentication of registration with Z4
	API permissions to the registration for EWS and PowerShell
	Assignment of the Exchange Administrator role for the XCAPP service principal object (ServicePrincipal)

- Select the installation directory, the log directory for ELO XC in S2 and the HTTPS port XCPORT above 5000
- Register ELO XC via the command line (administrator):

```
eloxc.exe -install -port:< HTTPS port > -logdir:<log directory> -cert:<fingerprint-Z3>
```

- Start ELO XC via Windows Services and open it in Edge https://S2:XCPORT
- Log on to ELO XC Manager with XCADM

Updates

The content of the installation directory is replaced by the update package. The *ELOxc.xml* file must be retained.

The automatic update of the configuration takes place whenever an older configuration version than the current program version is found.

Connection settings

	Local	M365
Directory name	AD1	TNNT
Authentication	XCACC	XCAPP
Key	XCPWD	TP2

Notes template

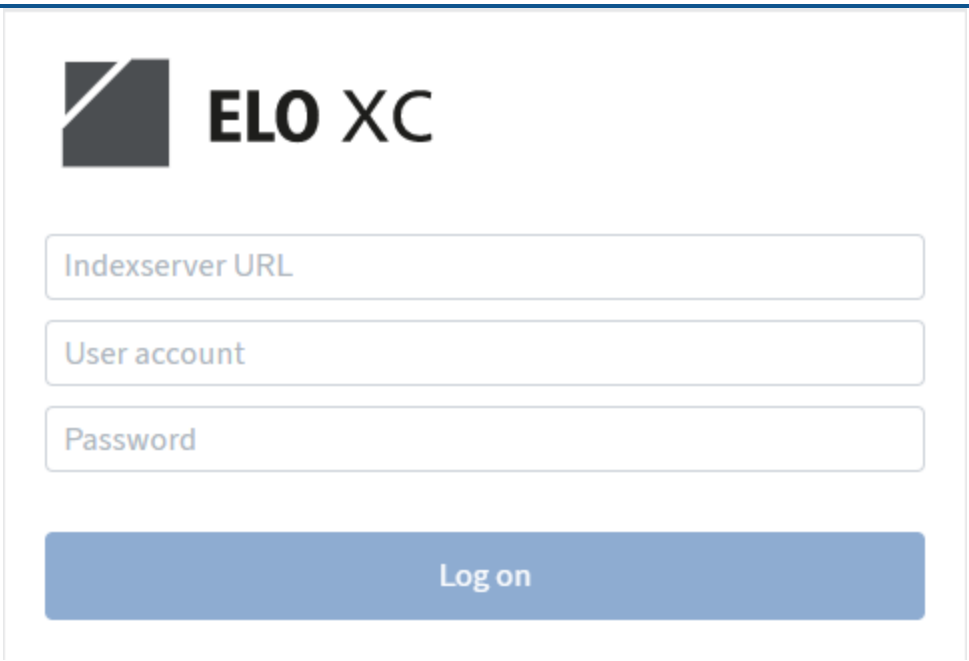
Checklist	Local	M365	Value
ELO user	XCADM	XCADM	
ELO XC system	S2	S2	
Manager port	XCPORT	XCPORT	
Manager certificate	TP1	TP1	
Directory name	AD1	TNNT	
Authentication	XCACC	XCAPP	
Key	XCPWD	TP2	

Execution

As a Windows service, ELO XC can be executed in the Local System context. The necessary authentication for the system environment and for ELO is provided in the configuration. It is recommended to run *cmd.exe* with administrative permissions when using a command prompt.

1. To access the ELO XC Manager user interface, open the browser and enter the following URL:

   ```
   https://<ELO XC host name>:<port>
   ```



You will see the ELO XC Manager logon screen.

2. Enter the address of the ELO Indexserver in the *Indexserver URL* field. You can select previously entered addresses from the drop-down menu.
3. Enter an ELO user in the *User account* field and the corresponding password in the field below.
4. Click *Log on*.

The ELO XC Manager home screen opens.

Installation details

ELO server certificate

If you are using Windows Server 2016, you may need to make adjustments in the registry to use the SSL server certificate. We recommend using at least Transport Layer Security (TLS) 1.2. For information on how to configure the TLS version for Windows systems, refer to the Microsoft documentation.

The Tomcat *server.xml* file should look like this:

```
<Connector

    SSLEnabled="true"

    ciphers="TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,

        TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256,

        TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,

        TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,

        TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,

        TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,

        TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,

        TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,

        TLS_AES_256_GCM_SHA384,

        TLS_CHACHA20_POLY1305_SHA256,

        TLS_AES_128_GCM_SHA256"

    clientAuth="false"

    keystoreFile="C:\keystore.jks"

    keystorePass="aaa"

    maxHttpHeaderSize="65536"
```

```
    maxThreads="1000"

    port="9093"

    protocol="org.apache.coyote.http11.Http11NioProtocol"

    scheme="https"

    secure="true"

    sslEnabledProtocols="+TLSv1.2,+TLSv1.3"

/>
```

The *keystoreFile*, *keystorePass*, and *port* properties must be adapted to the respective server installation. Make sure that you enter the cipher suites *TLS*ECDHE*RSA*WITH*AES*256*GCM*SHA384* and *TLS*ECDHE*RSA*WITH*AES*128*GCM*SHA256*. If you make changes in the *server.xml*, you have to restart the ELO server.

If you have activated the correct TLS version in the system, configured the *server.xml* as described, and ELO XC still cannot establish an SSL connection to the Indexserver, you can try removing the *ciphers* property from *server.xml*.

If this also fails, you can still try to enter TLS version 1.1 as well:

```
sslEnabledProtocols="+TLSv1.1,+TLSv1.2,+TLSv1.3"
```

PowerShell cmdlets

> **Please note**
>
> The following PowerShell cmdlets illustrate what needs to be done to establish a successful PowerShell connection to Microsoft 365 Exchange Online. You can find the complete documentation and help for troubleshooting issues on the Microsoft website. In particular, the examples used here may not be as complete or up-to-date as the current software.

Certificates

When generating a self-signed certificate, you should copy the fingerprint from PowerShell and save it separately for use in other cmdlets or the ELO XC configuration.

This is how you create a certificate:

```
New-SelfSignedCertificate -DnsName "dns.name" -CertStoreLocation "cert:\LocalMachine\My" -N
```

The *DnsName* parameter must not contain any wildcards. The ELO XC host certificate (Z3) requires the local *fully qualified domain name (FQDN)*, whereas the app registration certificate (Z4) requires the Microsoft 365 tenant name. We recommend that you always save the fingerprint of new certificates.

```
PS C:\Windows\system32> New-SelfSignedCertificate -DnsName "srvtxcdev16vm.xc.local" -CertStor

   PSParentPath: Microsoft.PowerShell.Security\Certificate::LocalMachine\My

Thumbprint                                Subject
----------                                -------
0C202737F8B809FAFA532C61E878F8DEEB385787  CN=srvtxcdev16vm.xc.local

PS C:\Windows\system32> New-SelfSignedCertificate -DnsName "xcdev.onmicrosoft.com" -CertSto

   PSParentPath: Microsoft.PowerShell.Security\Certificate::LocalMachine\My

Thumbprint                                Subject
----------                                -------
2E5B4D92DFBE8CFB49CF57106534EC50B12E6974  CN=xcdev.onmicrosoft.com
```

In this example, `0C202737F8B809FAFA532C61E878F8DEEB385787` was used for Z3 and `2E5B4D92DFBE8CFB49CF57106534EC50B12E6974` was used for Z4. Since Z4 is required for app registration in Microsoft 365, you need to export it as CER file without private key:

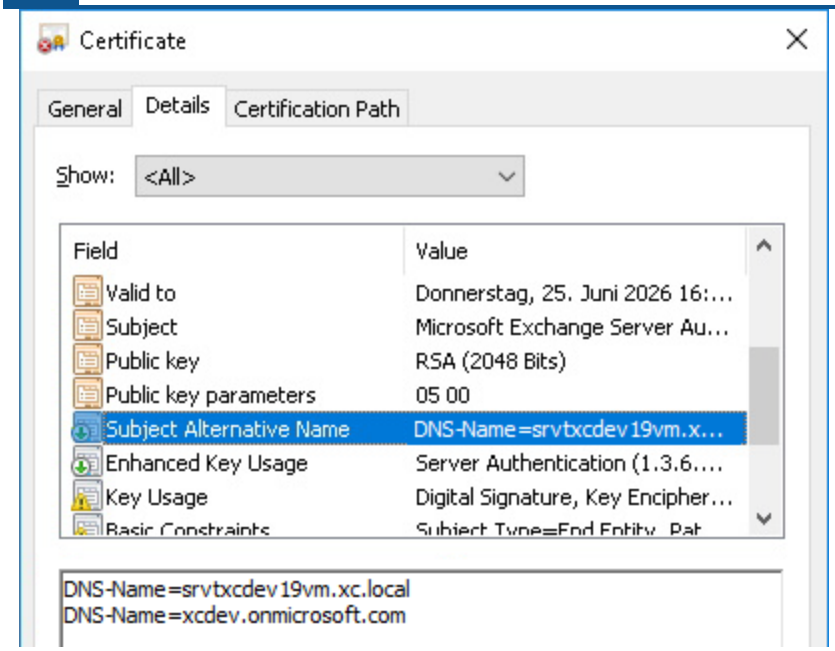
```
$cert = Get-ChildItem -Path cert:\LocalMachine\My\2E5B4D92DFBE8CFB49CF57106534EC50B12E6974
$cert | Export-Certificate -FilePath C:\XcAppReg.cer
```

```
PS C:\Windows\system32> $cert = Get-ChildItem -Path cert:\LocalMachine\My\2E5B4D92DFBE8CFB49CF57106534EC50B12E6974
PS C:\Windows\system32> $cert | Export-Certificate -FilePath C:\XcAppReg.cer

    Verzeichnis: C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        25.05.2022     15:47            832 XcAppReg.cer
```

If you want to use the same certificate for the ELO XC web host and the app registration (Z3=Z4), generate the certificate with two values for *DnsName*.

```
PS C:\Windows\system32> New-SelfSignedCertificate -DnsName "srvtxcdev19vm.xc.local", "xcdev.onmicrosoft.com" -Cert

   PSParentPath: Microsoft.PowerShell.Security\Certificate::LocalMachine\My

Thumbprint                                Subject
----------                                -------
9428F14E2E9A50DEC74D8EFD9D95DB5C3602F38D  CN=srvtxcdev19vm.xc.local
```

In this case, you would need a new fingerprint (here: `9428F14E2E9A50DEC74D8EFD9D95DB5C3602F38D`). You can check the DNS name in the local certificate store:

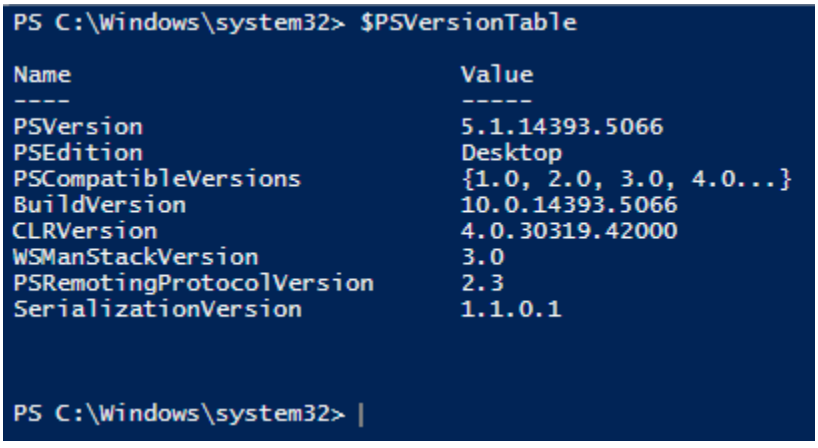


Install module for Exchange Online

The PowerShell examples assume that the user has administrative privileges. Run PowerShell as administrator. Use at least PowerShell version 5.1. If necessary, update the local PowerShell version.

This is how to check the local PowerShell version:

```
$PSVersionTable
```



Most cmdlets require external calls. In particular, the connection to Exchange Online must be enabled for remote PowerShell access. To install missing modules and the *PackageProvider*, you may be required to change the TSL version:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

To allow remote PowerShell access, you need to enable this function:

```
Enable-PSRemoting
```

You should only run signed external cmdlets and scripts. You need to confirm the start of the execution manually:

```
Set-ExecutionPolicy RemoteSigned
```

```
PS C:\Windows\system32> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
PS C:\Windows\system32> Enable-PSRemoting
PS C:\Windows\system32> Set-ExecutionPolicy RemoteSigned
PS C:\Windows\system32>
```

First, you can check whether the Exchange Online module (*ExchangeOnlineManagement* module) is available locally:

```
Get-Module
```

```
PS C:\Windows\system32> Get-Module

ModuleType Version    Name                                Export
---------- -------    ----                                ------
Binary     1.0.0.0    CimCmdlets                          {Export
Script     1.0.0.0    ISE                                 {Get-I
Manifest   3.1.0.0    Microsoft.PowerShell.Management     {Add-C
Manifest   3.0.0.0    Microsoft.PowerShell.Security       {Conve
Manifest   3.1.0.0    Microsoft.PowerShell.Utility        {Add-M
Manifest   3.0.0.0    Microsoft.WSMan.Management          {Conne
Manifest   2.0.0.0    NetSecurity                         {Get-D
```

If it is missing, you must download and install it. First install the *PackageProvider* for *NuGet* and import it. Afterwards, you need the *PowerShellGet* module.

This is how to install and import the *PackageProvider*:

```
Install-PackageProvider -Name NuGet -Force

Import-PackageProvider -Name NuGet
```

```
PS C:\Windows\system32> Install-PackageProvider -Name NuGet -Force -Scope AllUsers

Name                           Version          Source           Summary
----                           -------          ------           -------
nuget                          2.8.5.208        https://onege... NuGet provider for


PS C:\Windows\system32> Import-PackageProvider -Name NuGet
```

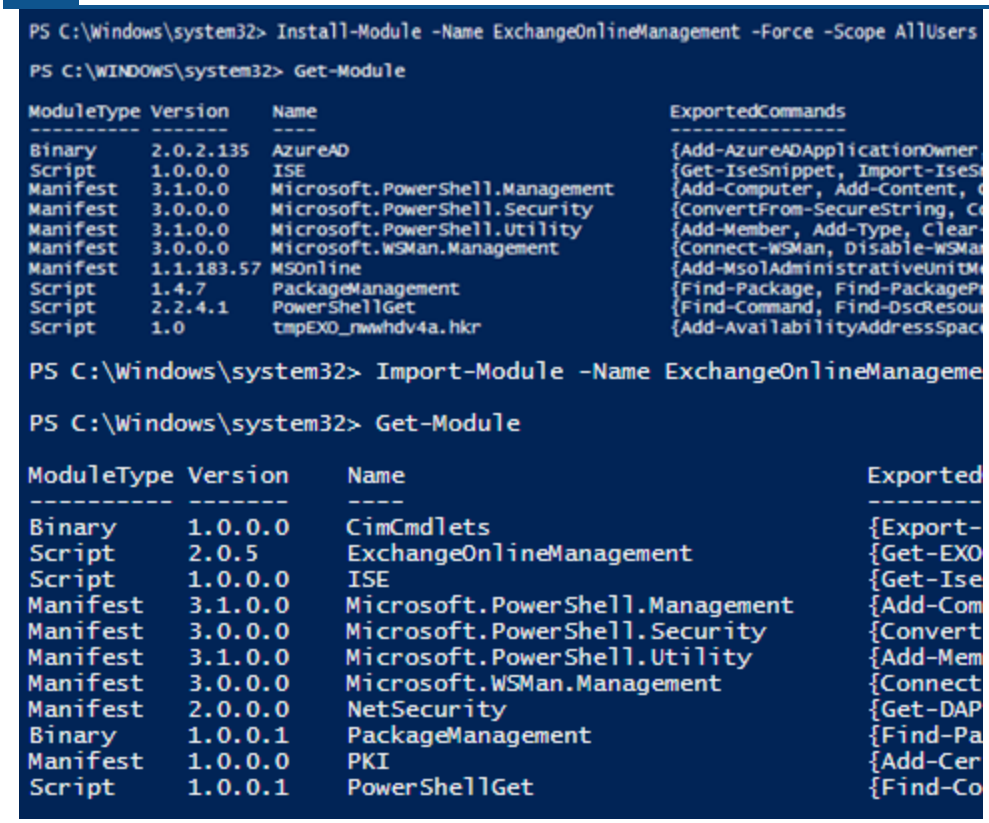
This is how to install the *PowerShellGet* module:

```
Install-Module PowershellGet -Force
```

```
PS C:\Windows\system32> Install-Module PowershellGet -Force -Scope AllUsers

PS C:\Windows\system32> Get-Module

ModuleType Version    Name                                ExportedCommands
---------- -------    ----                                ----------------
Binary     1.0.0.0    CimCmdlets                          {Export-BinaryMiLog,
Script     1.0.0.0    ISE                                 {Get-IseSnippet, Impo
Manifest   3.1.0.0    Microsoft.PowerShell.Management     {Add-Computer, Add-Co
Manifest   3.0.0.0    Microsoft.PowerShell.Security       {ConvertFrom-SecureSt
Manifest   3.1.0.0    Microsoft.PowerShell.Utility        {Add-Member, Add-Type
Manifest   3.0.0.0    Microsoft.WSMan.Management          {Connect-WSMan, Disab
Manifest   2.0.0.0    NetSecurity                         {Get-DAPolicyChange,
```

Now you can install and import the module for Exchange Online:

```
Install-Module -Name ExchangeOnlineManagement -Scope AllUsers

Import-Module -Name ExchangeOnlineManagement
```



> **Please note**
>
> The module for Exchange Online must be installed for all users, because the ELO XC service is usually registered with its own account or the system account, but the module is installed with a different account.

Test module for Exchange Online

If the module for Exchange Online is installed, ELO XC imports it automatically at runtime. If the import is successful, the connection to Exchange Online can be established. Once you have set up the app registration, you can test the PowerShell connection manually. Use the app ID and the tenant from the registration:

```
Connect-ExchangeOnline -CertificateThumbPrint 2E5B4D92DFBE8CFB49CF57106534EC50B12E6974 -Org
```

```
PS C:\Windows\system32> Connect-ExchangeOnline -CertificateThumbPrint 2E5B4D92DFBE8CFB49CF57106534EC50B12E6974 -Organiza

------------------------------------------------------------------------------
The module allows access to all existing remote PowerShell (V1) cmdlets in addition to the 9 new, faster, and more relia
mdlets.

|------------------------------------------------------------------------------|
|    Old Cmdlets                    |    New/Reliable/Faster Cmdlets           |
|------------------------------------------------------------------------------|
|    Get-CASMailbox                 |    Get-EXOCASMailbox                     |
|    Get-Mailbox                    |    Get-EXOMailbox                        |
|    Get-MailboxFolderPermission    |    Get-EXOMailboxFolderPermission        |
|    Get-MailboxFolderStatistics    |    Get-EXOMailboxFolderStatistics        |
|    Get-MailboxPermission          |    Get-EXOMailboxPermission              |
|    Get-MailboxStatistics          |    Get-EXOMailboxStatistics              |
|    Get-MobileDeviceStatistics     |    Get-EXOMobileDeviceStatistics         |
|    Get-Recipient                  |    Get-EXORecipient                      |
|    Get-RecipientPermission        |    Get-EXORecipientPermission            |
|------------------------------------------------------------------------------|

To get additional information, run: Get-Help Connect-ExchangeOnline or check https://aka.ms/exops-docs

Send your product improvement suggestions and feedback to exocmdletpreview@service.microsoft.com. For issues related to
odule, contact Microsoft support. Don't use the feedback alias for problems or support issues.
------------------------------------------------------------------------------


PS C:\Windows\system32> |
```

Once the connection is successfully established, you can test the two required *Get-Mailbox* and *Get-MailboxPermission* cmdlets:

```
Get-Mailbox -RecipientTypeDetails 'sharedmailbox' * | Get-MailboxPermission | where {$_.use
```

```
PS C:\Windows\system32> Get-Mailbox -RecipientTypeDetails 'sharedmailbox' * | Get-MailboxPermission | wher

Identity  User
--------  ----
XC Shared xcadmin@xcdev.onmicrosoft.com
XC Shared xc1@xcdev.onmicrosoft.com
XC Shared xc2@xcdev.onmicrosoft.com
```

You should disconnect active PowerShell connections at the end as follows:

```
Disconnect-ExchangeOnline
```

```
PS C:\Windows\system32>  Disconnect-ExchangeOnline
Removed the PSSession ExchangeOnlineInternalSession_1 connected to outlook.office365.com
Disconnected successfully !
```

To prevent ELO XC from accessing specific mailboxes through the PowerShell connection, you can use *New-ApplicationAccessPolicy*. This requires an active PowerShell connection:

```
New-ApplicationAccessPolicy -AppId cbfafa12-36c7-4586-a9c7-f94e923f1d26 -PolicyScopeGroupId
```

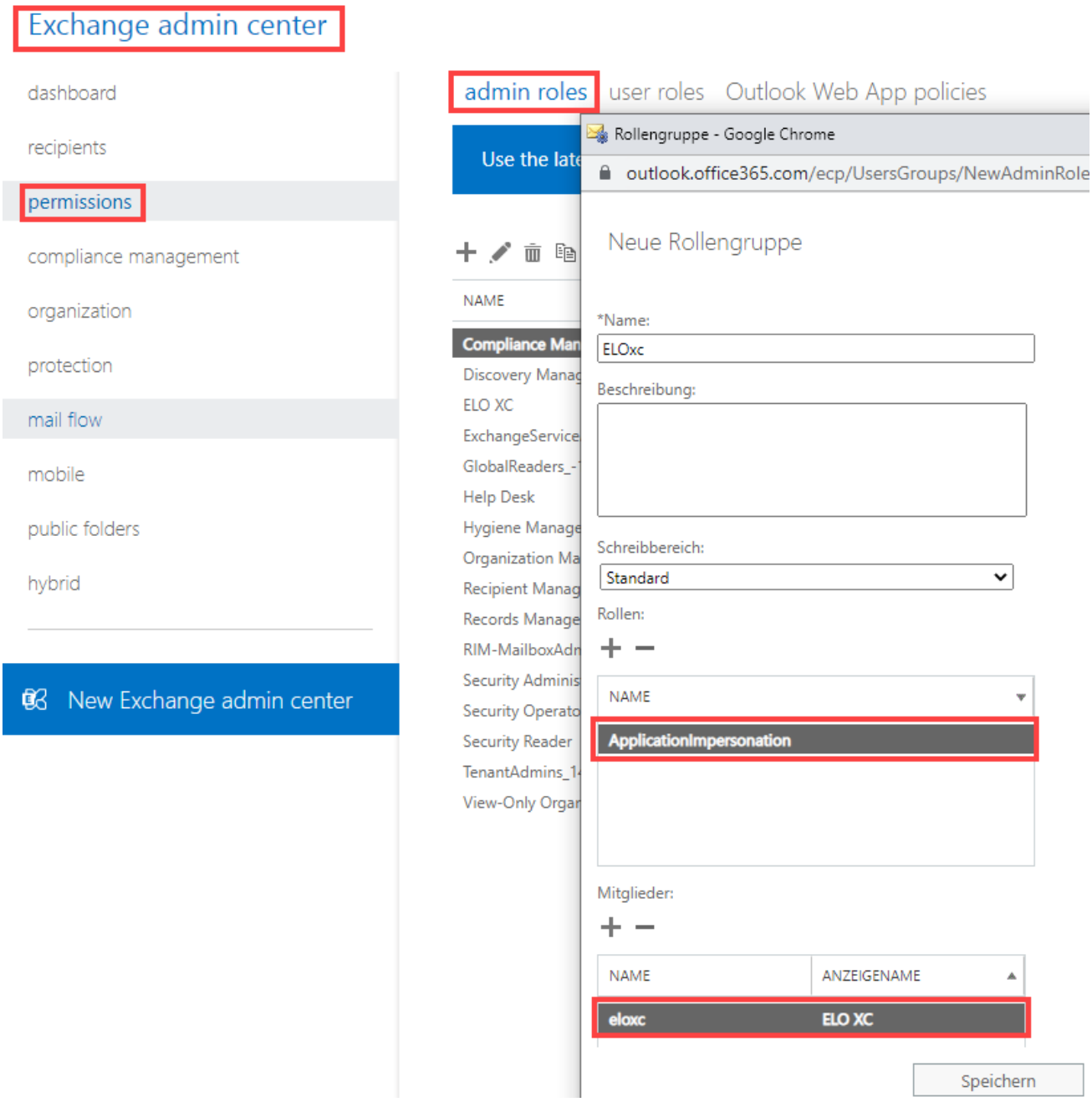
```
PS C:\WINDOWS\system32> New-ApplicationAccessPolicy -AppId cbfafa12-36c7-4586-a9c7-f94e923f1d26 -PolicyScopeGroupId xc2@xcdev.onmicrosoft.com -AccessRight DenyAccess


RunspaceId       : 78acdaba-4602-45db-9e66-451ba59de8c4
ScopeName        : XC User 2
ScopeIdentity    : xc2
Identity         : e1516898-cda4-47cd-ab77-e2f7e7c75745\cbfafa12-36c7-4586-a9c7-f94e923f1d26:S-1-5-21-2103643036-1067027473-1901050440-30458400;8cd87a58-7cd1-46fc-a7
                   -fd963006829a
AppId            : cbfafa12-36c7-4586-a9c7-f94e923f1d26
ScopeIdentityRaw : S-1-5-21-2103643036-1067027473-1901050440-30458400;8cd87a58-7cd1-46fc-a7c7-fd963006829a
Description      : Restrict this app to members of distribution group EvenUsers.
AccessRight      : DenyAccess
ShardType        : All
IsValid          : True
ObjectState      : Unchanged
```

Please note

After a successful test, the Exchange Online module may fail to load in some environments. If you check the MainCatalog log, you will see a message that the *Connect-ExchangeOnline* cmdlet cannot be found. In this case, you should install Microsoft PowerShell 7 in addition to Microsoft PowerShell 5. Install the module there again.

EAC: ApplicationImpersonation

In an on-premises installation, ELO XC requires the ApplicationImpersonation role. This role allows the service account (*XCACC*) to impersonate access to mailboxes. The role needs to be assigned in EAC (Exchange Admin Center).



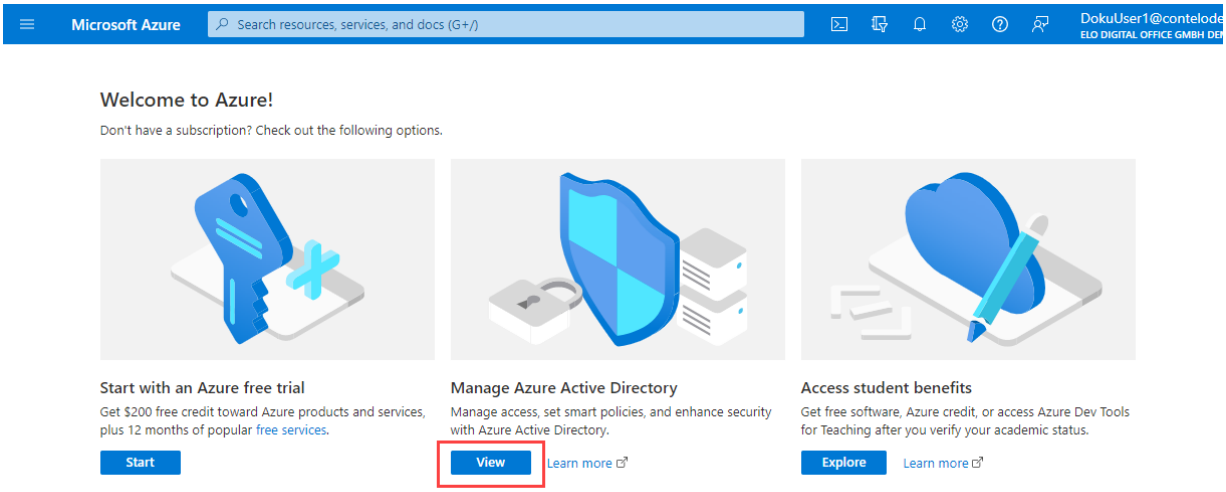
App registration

To allow ELO XC to connect to Microsoft 365 with Modern Authentication (OAuth 2.0), an app registration in the tenant (*TNNT*) is required, which administratively acts as the service principal of ELO XC and is assigned the *Exchange administrator* role. The app registration has an ID that is used to recognize that external authentication has taken place. The key is a locally generated certificate whose public key (CER file) must be added to the app registration. The API permissions of the app registration determine which functions are available to ELO XC in Microsoft 365.

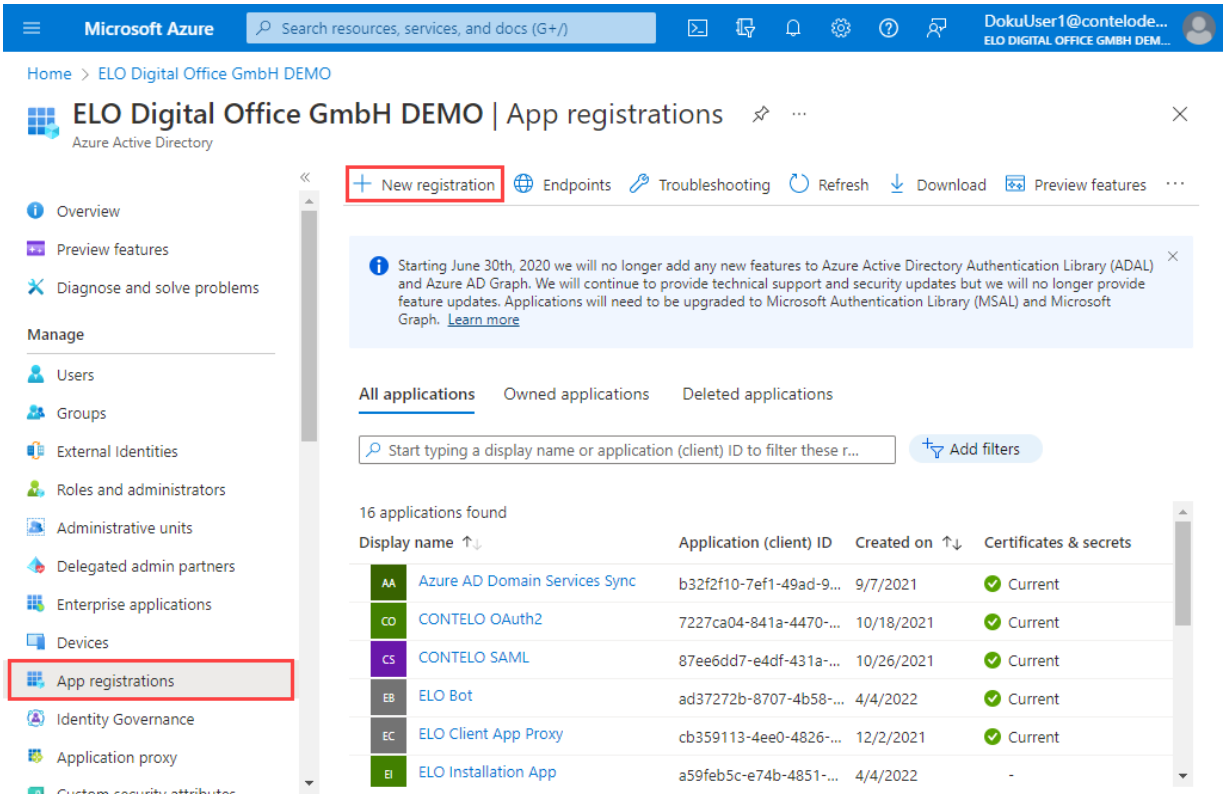
Create

1.

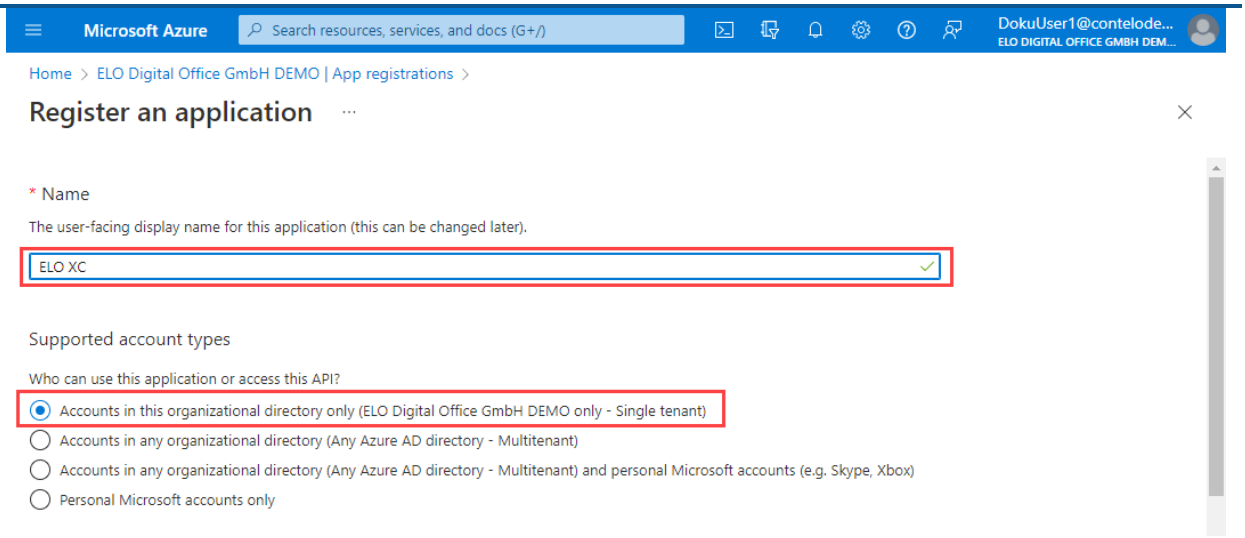
Sign in to the Microsoft Azure Portal: https://portal.azure.com/.



2. Select *Manage Azure Active Directory* > *View*.

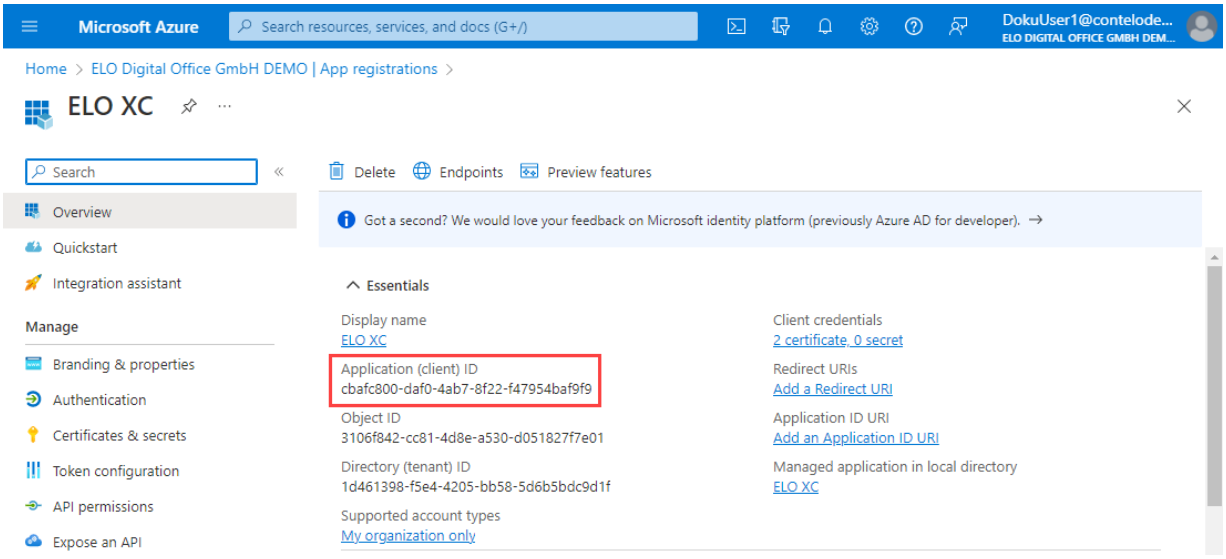


3. Select *App registrations* > *New registration* tab.



The *Register an application* window opens.

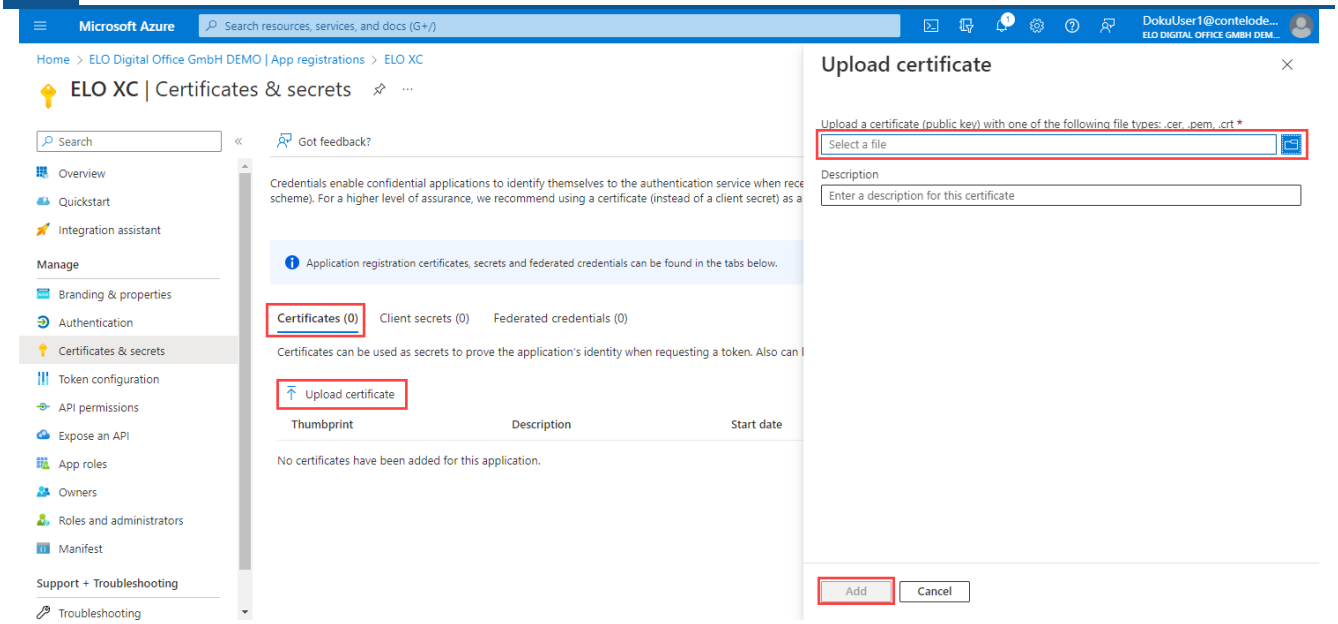
4. Enter *ELO XC* as the display name for the application.
5. Under *Supported account types*, select *Accounts in this organizational directory only ...*.



ELO XC is now registered as an application and listed in the Microsoft Azure overview. Save the application ID (*XCAPP*).

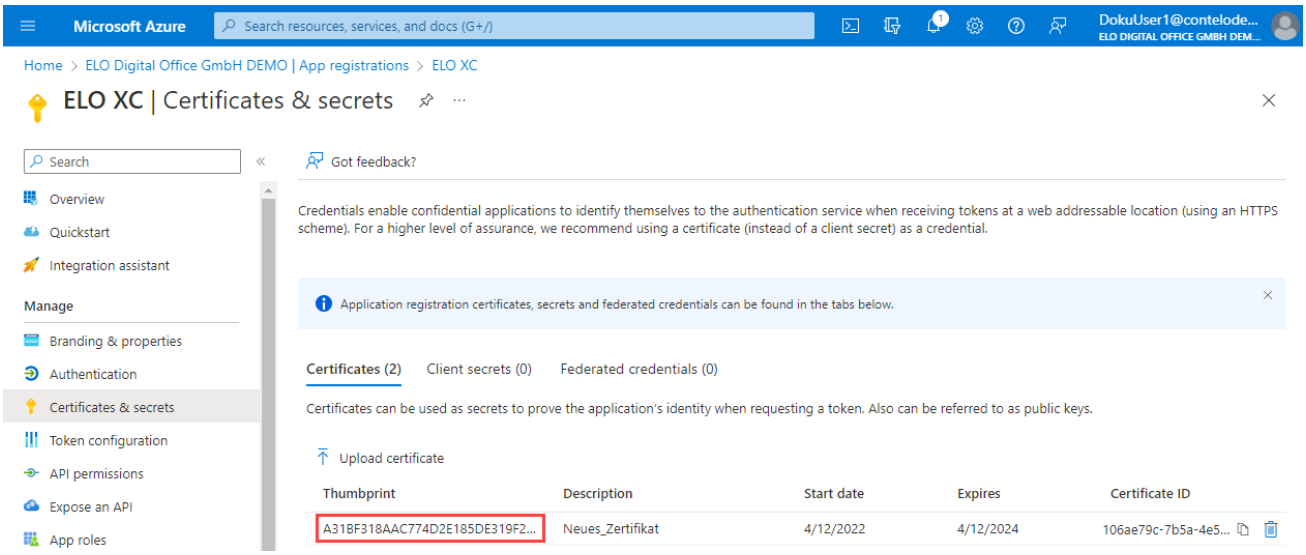
Upload certificate

App registration in Microsoft Azure requires a certificate file. Refer to the *Self-signed certificates* section of the PowerShell documentation to learn how to create them in PowerShell.



1. Select *Certificates & secrets* in the sidebar.
2. Click *Upload certificate*.
3. Select a locally stored certificate and click *Add*.

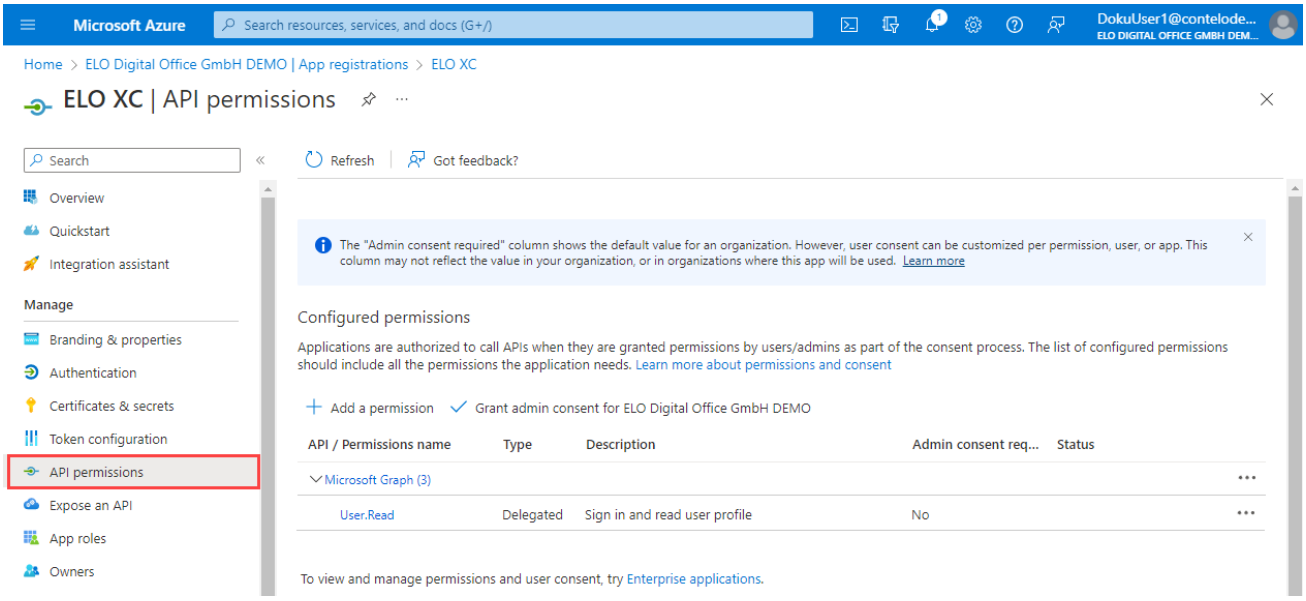
> **Please note**
>
> App registrations only allow public certificate keys, which CER files are. PFX files contain public and private keys, which is why they cannot be used.



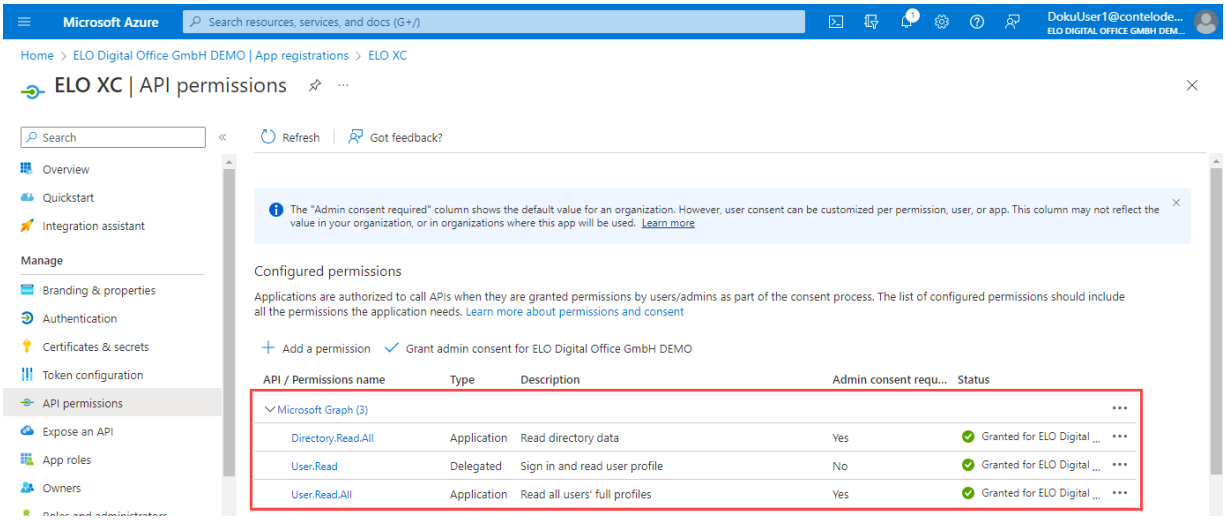
You will find the uploaded certificates in the Microsoft Azure overview of ELO XC certificates and secrets. At this point, you can also check the fingerprint again, which must match Z4.

API permissions

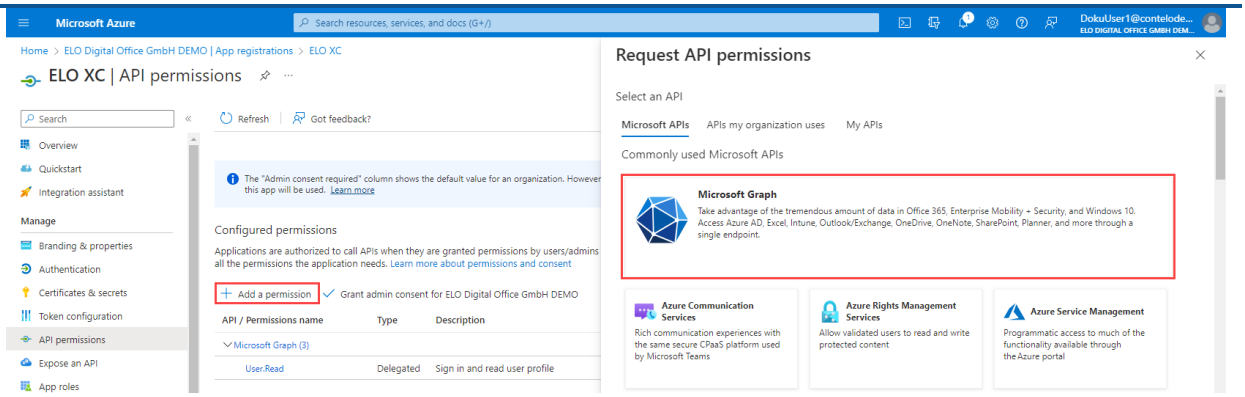
In this section, you will learn which API permissions are required and how to grant them.



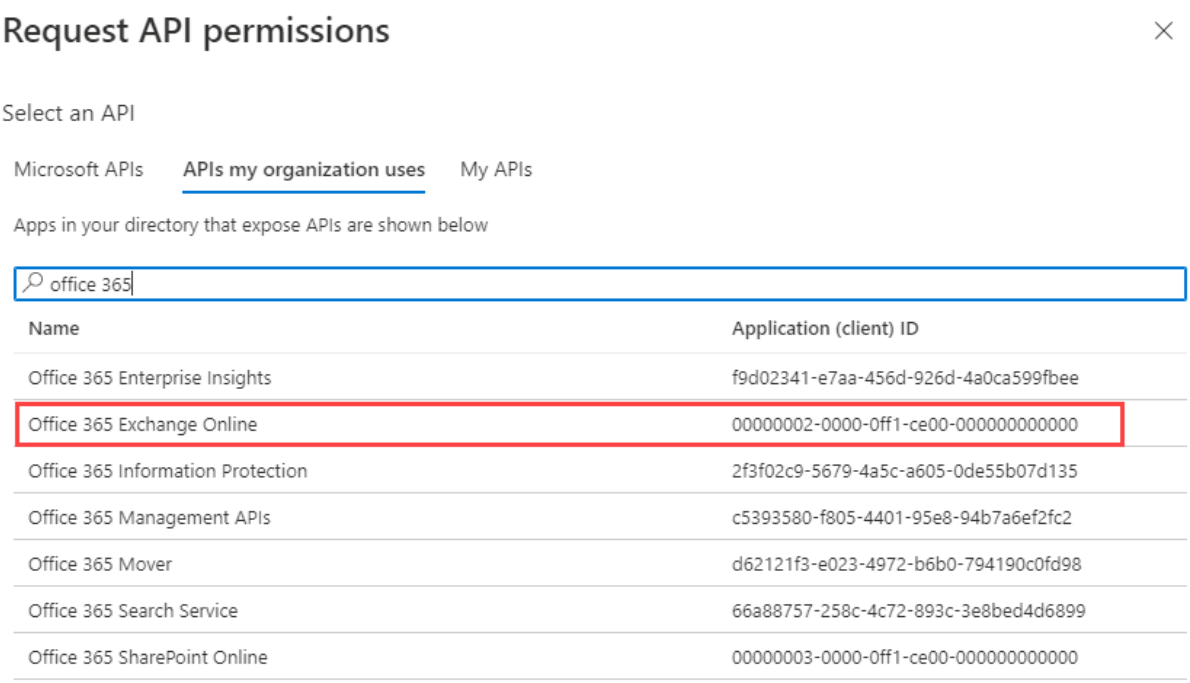
1. Select *API permissions* in the sidebar.



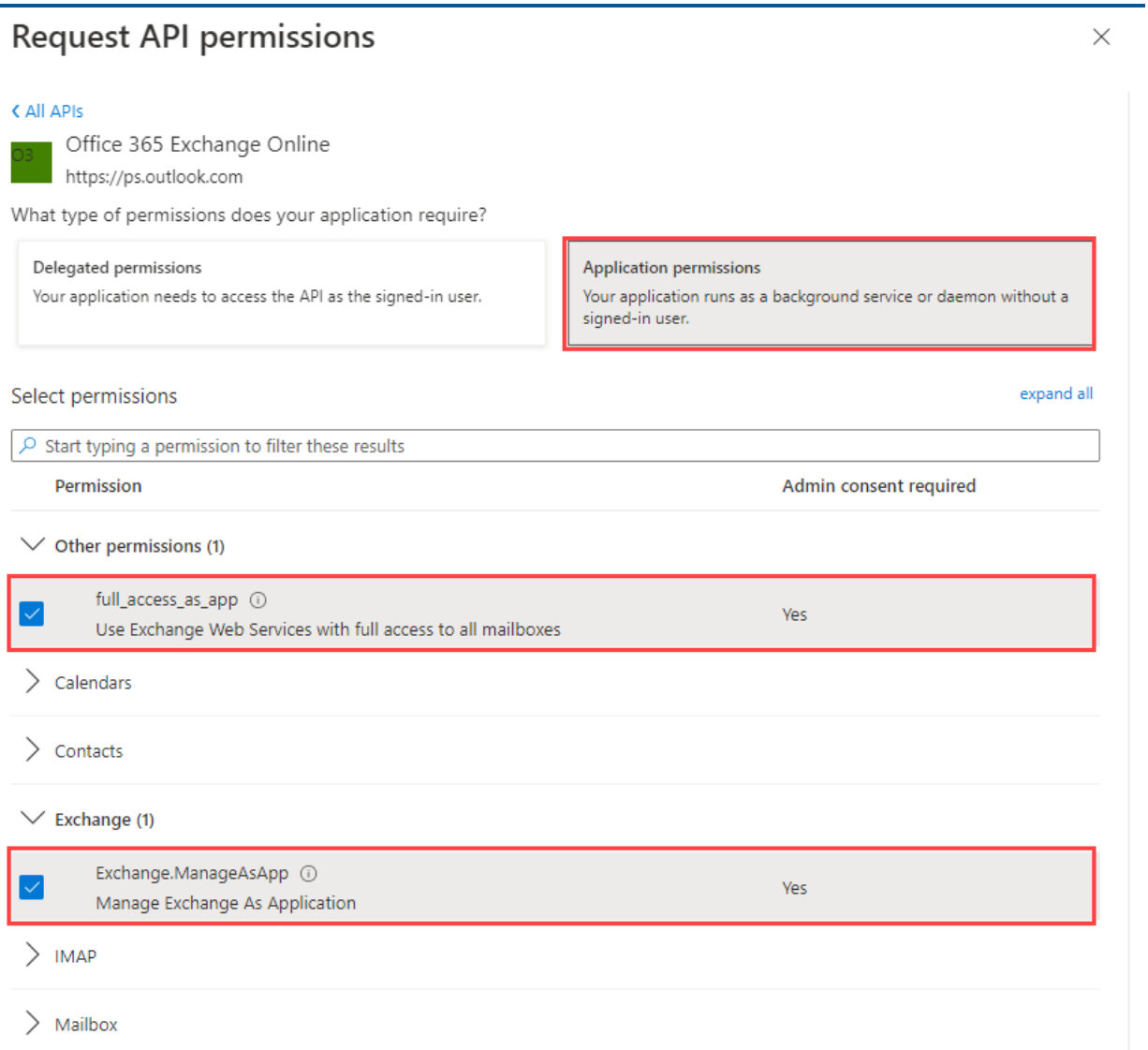
2. Right-click the *User.Read* permission of the Microsoft Graph API to open the context menu. Select *Revoke admin consent* first and then *Remove permission* to remove the Microsoft Graph API.



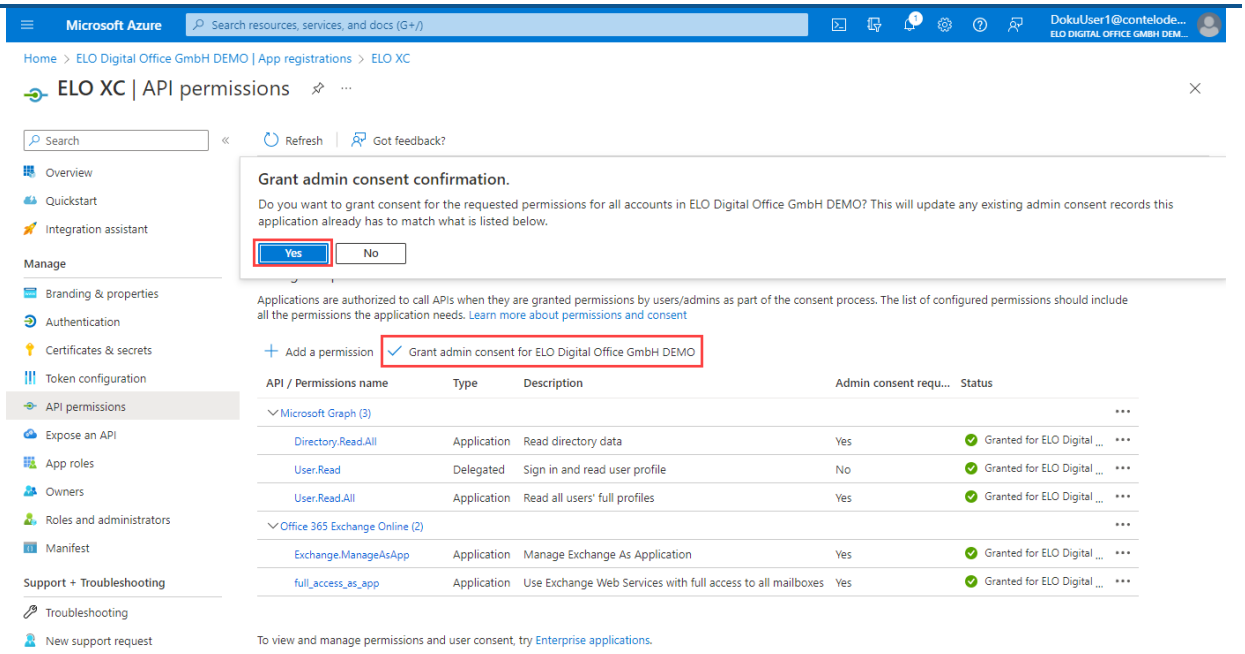
3. Click *Add a permission*.



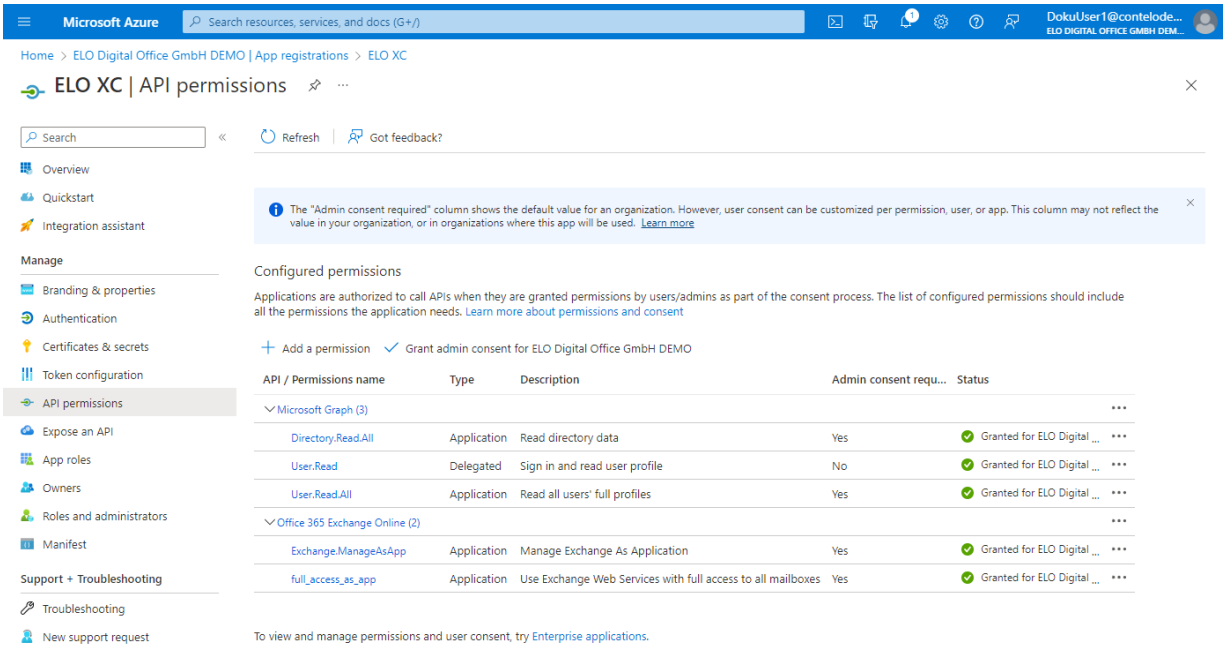
4. In the *APIs my organization uses* tab, select the *Office 365 Exchange Online* application.



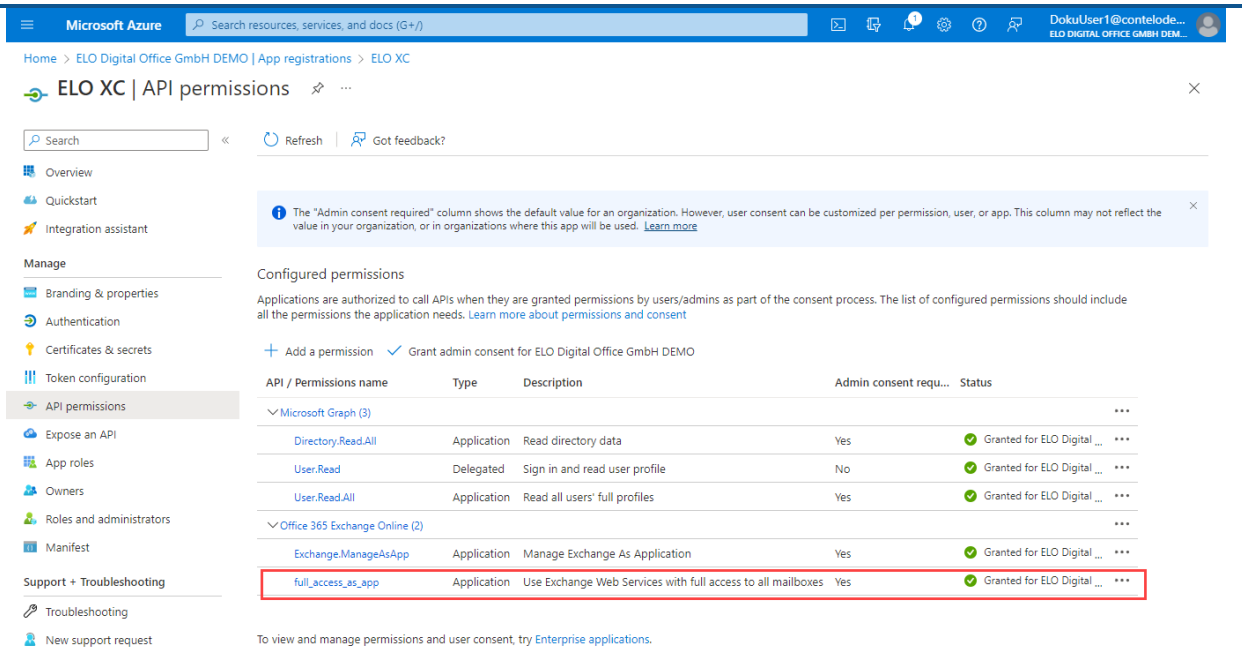
5. Click *Application permissions* and select the following application permissions: *full*access*as_app*, *Exchange.ManageAsApp*.



6. Click *Grant admin consent for ...* and click *Yes* to confirm.



Once admin consent is granted, the status in the API permissions overview changes to *Granted for ...*.



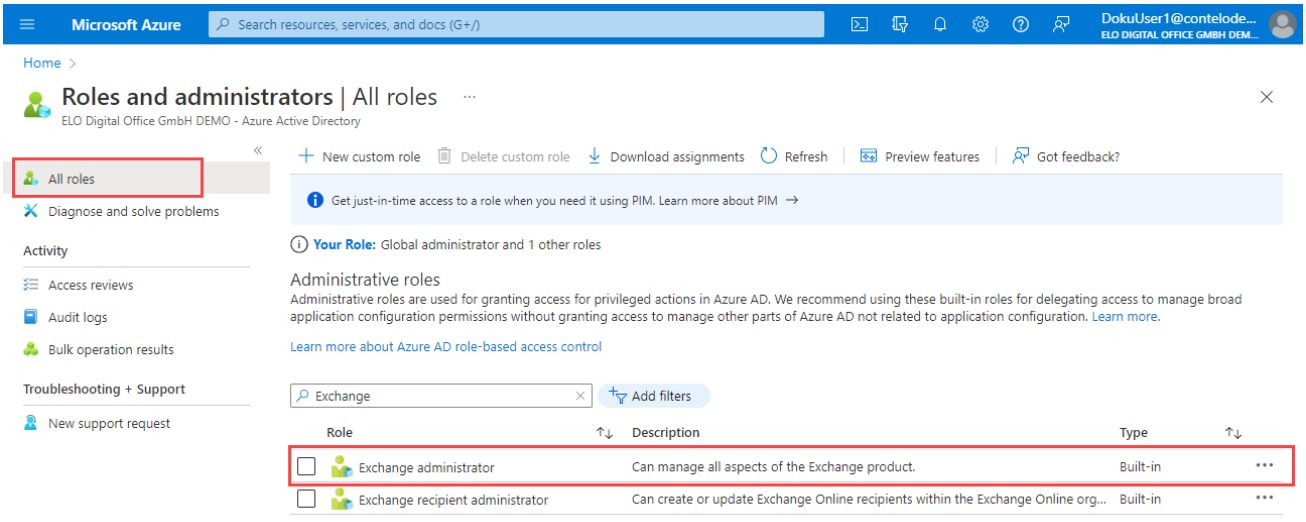
The *full*access*as_app* permission allows full access to all mailboxes.

To limit access to specific *Exchange Online mailboxes*, you need to generate *application access policies*.

The process is described here: https://learn.microsoft.com/en-us/graph/auth-limit-mailbox-access.

'Exchange administrator' role

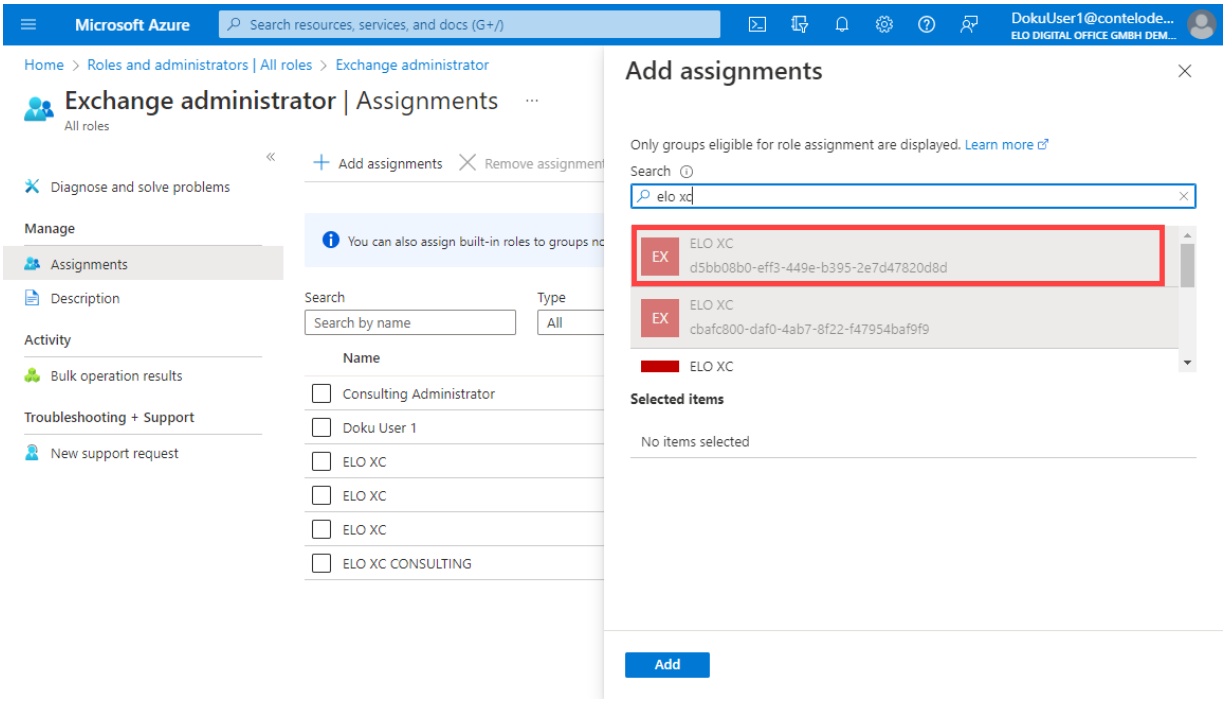
A ServicePrincipal is automatically created during the app registration. With the API permissions, ELO XC is authorized to get proxy access to mailboxes and read their permission settings to find delegates in shared mailboxes. To establish a connection with the `Connect-ExchangeOnline` cmdlet, the service principal needs the *Exchange administrator* role.



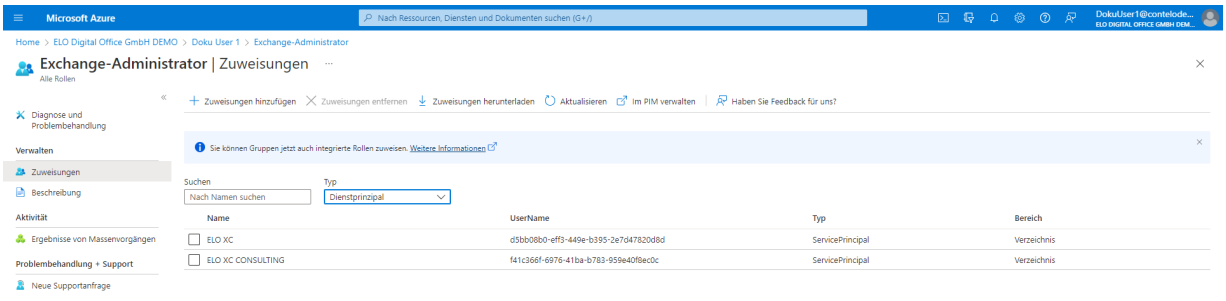
1.

In the Microsoft Azure Active Directory sidebar, select the *Roles and administrators* tab.

2. Select the *Exchange administrator* role.



3. Assign *ELO XC* to the created app registration.



ELO XC has now been assigned the role of Exchange administrator.

Updating the main version

Update paths

- ELO XC updates the configuration automatically. This includes changes to the configuration formatted in XML due to dropped, new or renamed items or properties.
  - In versions prior to ELOxc 12, this requires a separate command prompt.
  - From ELOxc 12 onwards, this is done automatically every time the program is started.
  - Up to and including ELOxc 20, all configuration versions down to and including ELOxc 9 are recognized.
  - As of ELO XC 21, the oldest allowed configuration version is ELOxc 11.
- When switching the main version, however, changes can occur that cannot be implemented automatically but require manual intervention. It is always advisable to check things manually in this case.

> **Warning**
>
> To allow you to easily perform a rollback in the event of unexpected issues, we recommend creating backup copies of the instance configurations in the *Manager* repository area before version migration.

Upgrade ELOxc 9 to ELOxc 10

1. Stop ELOxc and the Tomcat that the console is running on.
2. Go to the directory *...\server\webapps* and delete the *xc-console.war* file.
3. In the ELOxc installation directory, run the *ELOxcTools.exe* as administrator and delete the ELOxc service.
4. Uninstall *ELOxc for Microsoft EWS 9* via the Control Panel.
5. Delete the remaining files in the installation directory.
6. In the registry, rename the *HKLM\Software\ELO Digital\ELOxc9* key to `EL0xc`.
7. Install *ELOxc for Microsoft EWS 10*.
8. Copy the console from the installation directory to the *...\prog\webapps* directory.
9. Start the Tomcat.
10. Run *ELOxcTools.exe* to register the service again and insert the new license key.
11. Use the ELO client (Administrator) to remove the `<Licenses>…</Licenses>` XML element from the SORDS extra text in the *//Administration//ELOxc Base//Workspace* and *//Administration//ELOxc Base//Service* paths.
12.

Run `ELOxc.exe -update` from the command line (Administrator).

Upgrade ELOxc 10 to ELOxc 11

1. Install .NET Framework 4.7.2
2. Stop ELOxc and the Tomcat that the console is running on.
3. Delete the *xc-console.war* file in *...\server\webapps* and the associated folders in *...\servers\Tomcat\webapps* and *...\servers\Tomcat\work\catalina\localhost*.
4. In the ELOxc installation directory, run the *ELOxcTools.exe* as administrator and delete the ELOxc service.
5. Uninstall *ELOxc for Microsoft EWS 10* via the Control Panel. Delete the remaining files in the installation directory.
6. Copy all files in the ELOxc 11 package to the installation directory.
7. Copy the console from the installation directory to the *...\prog\webapps* directory.
8. Start the Tomcat.
9. Run *ELOxcTools.exe* to register and start the service. Insert the new license key.
10. Run `ELOxc.exe -update` from the command line (Administrator).
11. After a few moments, check the update log file to see if it contains the line *update finished and succeeded*.
12. Since the *Keywording* action was removed in ELOxc 11 and replaced by *templates*, all instance configurations have to be adjusted manually.

Keywording with ELOxc 11

In ELOxc version 11, the *Keywording* action was removed. Instead, keywording is stored as a template, which makes it easier to work with ELOxc EWS in the long term.

You can quickly and easily convert existing keywording into templates.

1. Go to the *Keywording* action that is currently displayed and copy the XML contents (from line 2, the first line is created when you create the template with names).
2. Switch to *Templates > Keywording* and add the keywording with the desired name.
3. Then edit the XML file and insert the old XML lines.
4. Finally, you only need to change the last line in your XML file: Replace `</KeywordingDef>` with `</TemplateKeywordingDef>` and remove the slash `/` from the first line.

Old *Keywording* action XML:

```
<KeywordingDef>
<Mask Name="EMail" EloSearchKey="ELOOUTL3" />
```

```
<ObjType TypeDefault="261" TypeWithAttachments="261" />
<OKey Name="ELOOUTL1">
<Match FieldName="EloSender" Id="1" Pattern="*" />
</OKey>
<OKey Name="ELOOUTL2">
<Match FieldName="EloRecipients" Id="2" Pattern="*" />
</OKey>
</KeywordingDef>
```

New *Keywording* template:

```
<TemplateKeywordingDef Name="test"/>
```

End result:

```
<TemplateKeywordingDef Name="test">
<Mask Name="EMail" EloSearchKey="ELOOUTL3" />
<ObjType TypeDefault="261" TypeWithAttachments="261" />
<OKey Name="ELOOUTL1">
<Match FieldName="EloSender" Id="1" Pattern="*" />
</OKey>
<OKey Name="ELOOUTL2">
<Match FieldName="EloRecipients" Id="2" Pattern="*" />
</OKey>
</TemplateKeywordingDef>
```

Upgrade ELOxc 11 to ELOxc 12

1. Stop ELOxc and the Tomcat that the console is running on.
2. Delete the WAR file and the console directories.
3. Remove the old service registration with `eloxc.exe -uninstall`.
4. Delete all ELOxc 11 files other than *ELOxc.xml* from the installation directory.
5. Extract all ELOxc 12 files to the installation directory.
6. Restart Tomcat.
7. Register ELOxc 12 with `eloxc.exe -install -logdir:<logdir> -port:<port> -cert:<certificate>`
8. Start ELOxc 12.
9. The configuration updates are performed automatically the first time you start the program.
10. If you use LDAP filters in action trees, you must transfer the filters to the instance connection settings and use the name assigned there as the parameter value in the action trees.

Upgrade ELOxc 12 to ELOxc 20

1. Set all instances to *idle* processing mode so that when ELOxc 20 is started for the first time, the program will not start processing without prior validation.
2. The catalog type *ews* no longer exists in ELOxc 20. Replace it with a PowerShell or Azure catalog and test the affected instances.
3. Stop ELOxc 12.
4. Remove the old service registration with `eloxc.exe -uninstall`.
5. Delete all ELOxc 12 files other than *ELOxc.xml* from the installation directory.
6. Extract all ELOxc 20 files to the installation directory.
7. Register ELOxc 20 with `eloxc.exe -install -logdir:<logdir> -port:<port> -cert:<certificate>`
8. Start ELOxc 20.
9. The configuration updates are performed automatically the first time you start the program.
10. Validate the instances again. Check for possible validation errors, especially:
    a. Consolidate deleted collective folders
    b. Consolidate path types
    c. Consolidate stubbing
11. Publish the validated instance configurations.
12. Restart ELOxc 20.

Incorrect validation of instances

Validate each instance. In some cases, error messages are expected that you need to correct manually in the instance configuration:

Removal of collective folders

Due to the removal of collective folders (when extracting attachments) and the changes in associated options for entering metadata, the pseudo-property *EloSplitFolderGuid* needs to be removed manually if used. You can simulate the mechanism for using collective folders by making simple configuration additions in *ArcPathDef*. However, a GUID for these folders is no longer available.

Introduction of path types

By introducing path types, all paths are marked as main paths, which violates the rule that there can only be one main path for each *ArcPathDef* action. All paths starting from the second one must be manually configured as paths for logical references as this is how they were used in ELOxc 12.

Stubbing

If you skeletonize messages (*StubbingDef* and *TemplateStubDef*), you have to change the action and the template, because the variables in *TemplateStubDef* have changed significantly. In addition, a template now requires you to specify both template types: HTML and text. This enables you to use a template for both message body types.

Upgrade ELOxc 20 to ELO XC 21

- The minimum version for automatic updates is ELOxc 11. Older versions must be migrated to an ELOxc 11 version or newer.
- In ELO XC 21, the instance configurations distributed to SORDS extra text are combined in an XML file. We therefore recommend that you create a backup of the ELOxc 20 configuration structure.
- PowerShell catalogs can no longer be used with basic authentication, but require the Exchange Online PowerShell V3 module and an app registration.
- Change the *Cert* property in the *ELOxc.xml* file. The property value must be the certificate fingerprint. A *TP=* prefix or even *CN=* is no longer needed and must be removed.

1. Stop ELOxc 20.
2. If necessary, remove the old service registration with `eloxc.exe -uninstall`.
3. Delete all ELOxc 20 files other than *ELOxc.xml* from the installation directory.
4. Extract all ELO XC 21 files to the installation directory.
5. Register ELO XC 21 with `eloxc.exe -install -logdir:<logdir> -port:<port> -cert:<certificate>`
6. Start ELOxc 20.
7. Check the main log of each instance for configuration errors.
8. Log on to ELO XC Manager and change/validate the instance configurations if necessary.