ELO for Microsoft

Connect to OneDrive

Table of contents

Connect to OneDrive

Check out to OneDrive

The function *Check out to OneDrive* allows the user to check out Microsoft Office documents to Microsoft OneDrive and edit them there. For the user, this is similar to the previous local check out process. What's new is that the interface with Microsoft 365 allows for use of Microsoft Office Online and collaborative working.

You can find more information about using the function in the client in the user documentation:

- ELO Java Client
- ELO Web Client
- ELO Desktop Client

Requirements

General technical requirements

- ELO Server Setup version 21.3 and higher
- Connection to Microsoft 365 with modern authentication (OAuth 2.0)
- App registration in Microsoft Azure of an app for ELO for Microsoft 365
- Activated ELOauth plug-in for ELO Indexserver (ELOix)
- Activated O365sync plug-in for ELO Indexserver

> **Please note**
>
> The *Check out to OneDrive* function can only be used with the *OAuth* authentication method, as Microsoft OneDrive requires an OAuth2 token. The *SAML* method cannot be used for the *Check out to OneDrive* function.

For more information on the ELOauth plug-in and the OAuth 2.0 authentication method, refer to the ELOauth Plug-in documentation.

Requirements for user authentication

- All users require a Microsoft work or school account with Microsoft OneDrive for Business.
- All users must be configured with their Microsoft work or school account in the ELO Administration Console and logged on to the corresponding ELO client.

Register new app in Microsoft Azure

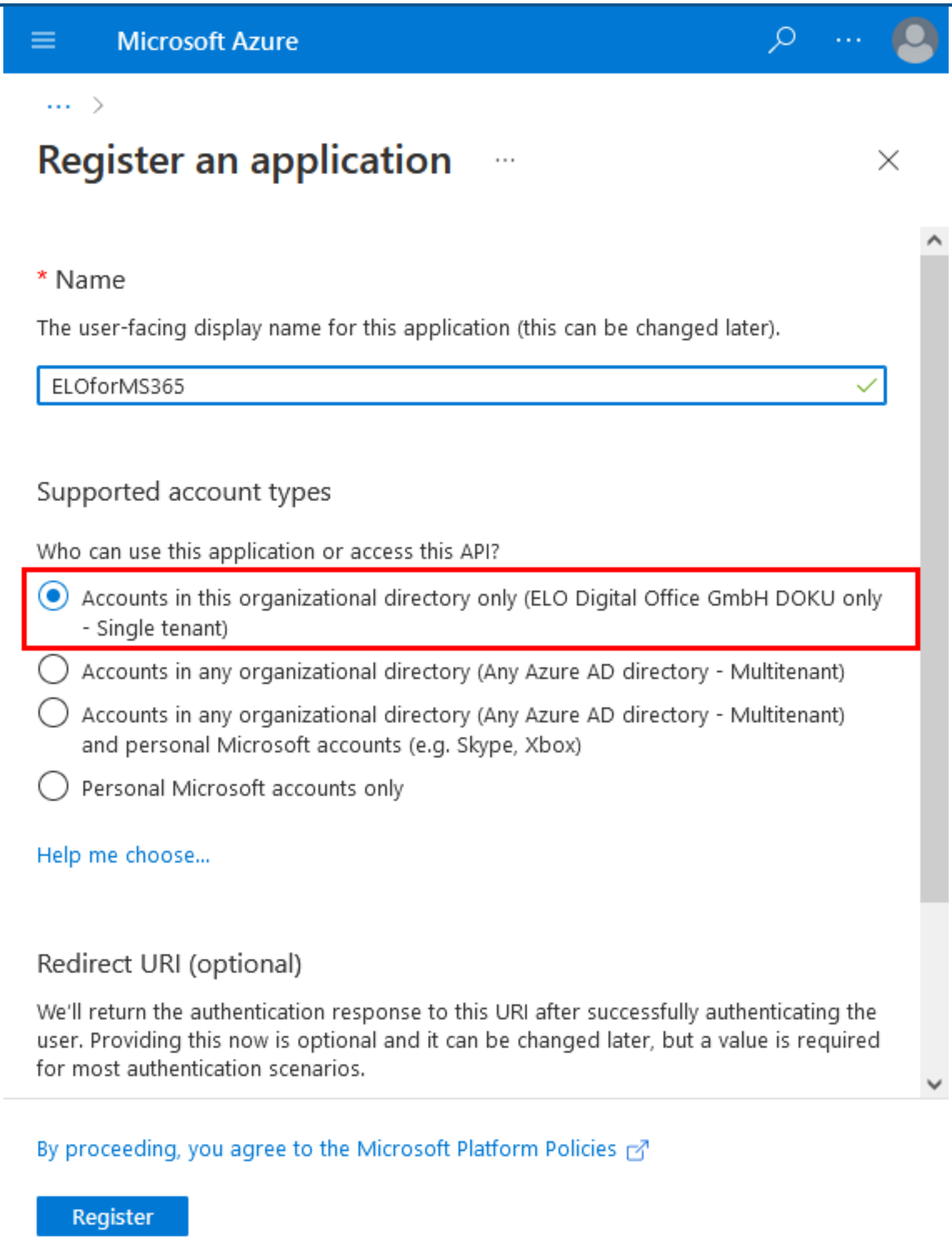
To connect to Microsoft 365 with modern authentication (OAuth 2.0), you need a new app in your Microsoft Azure environment. You will learn how to do this and what to consider during configuration in the following.

For more information on app registration, refer to the Microsoft documentation Register app or web API.

1. Sign in to the Microsoft Azure Portal at the following address: https://portal.azure.com/.

2. Select *Azure Active Directory*.

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



* Name

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

ELOforMS365

Supported account types

Who can use this application or access this API?

- ● Accounts in this organizational directory only (ELO Digital Office GmbH DOKU 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)

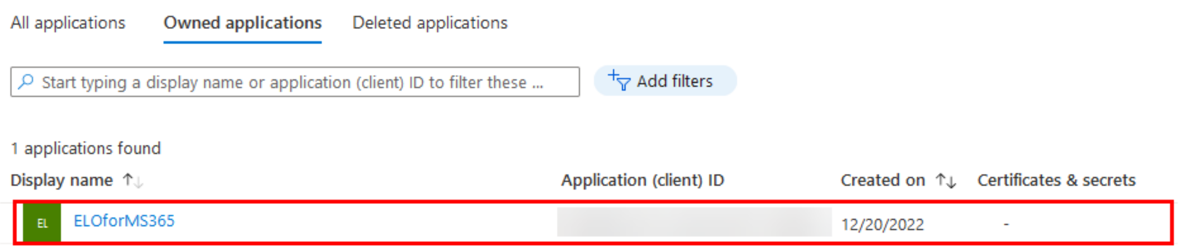
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

By proceeding, you agree to the Microsoft Platform Policies

The *Register an application* window opens.

4. Enter a name of your choice as the display name for the application. In this example, the application is called *ELOforMS365*.

5. Under *Supported account types*, select the option *Accounts in this organizational directory only ...*.


All applications
Owned applications
Deleted applications
Start typing a display name or application (client) ID to filter these ...
Add filters

1 applications found

Display name	Application (client) ID	Created on	Certificates & secrets
EL ELOforMS365	[redacted]	12/20/2022	-

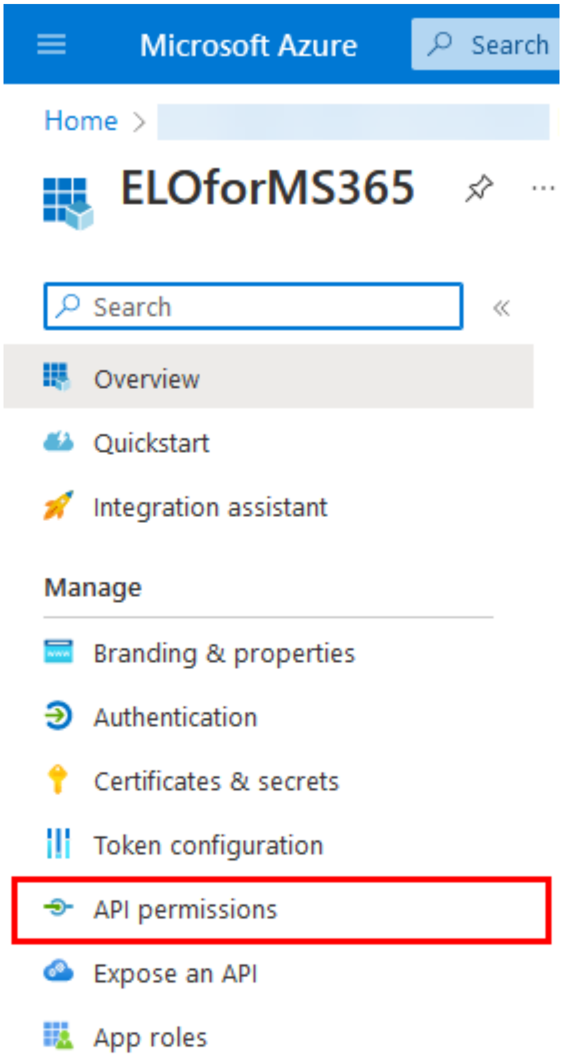
The new app is now registered as an application and listed in the Microsoft Azure overview.

Configure API permissions

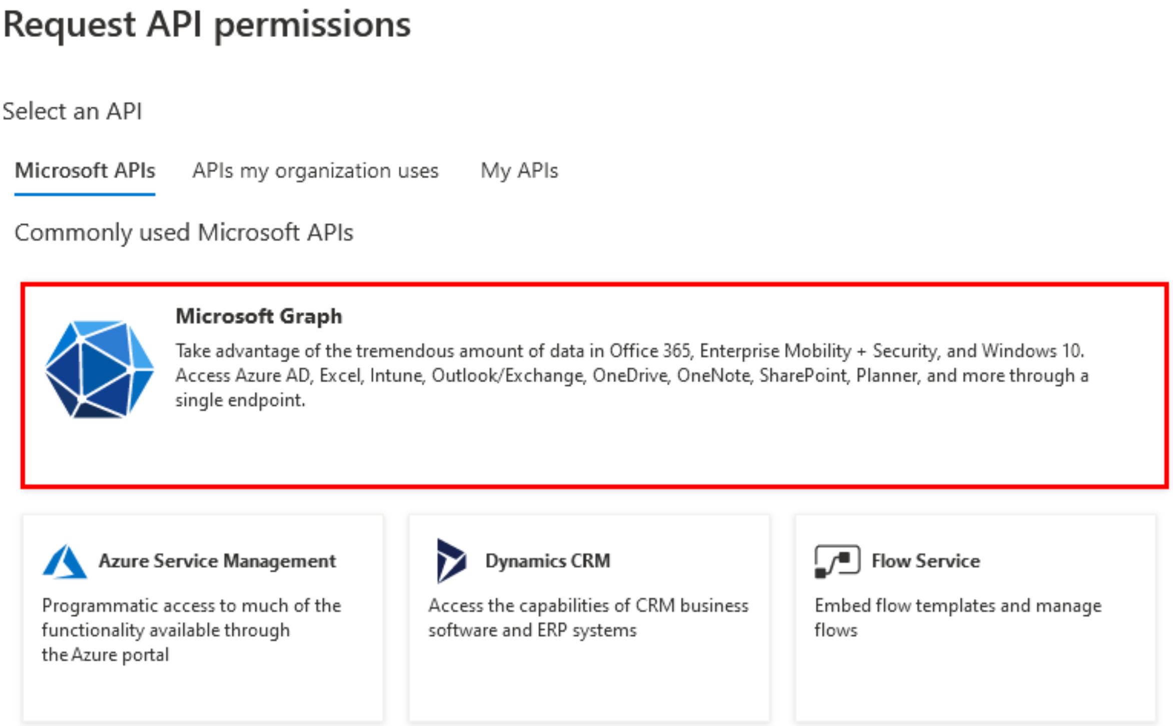
In this section, you will learn which API permissions are required and how to grant them.

1. In the overview, select the newly registered app.

   The overview window for your newly registered app opens.



2. Select the item *API permissions* in the sidebar.



Request API permissions

Select an API

Commonly used Microsoft APIs

Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

Azure Service Management
Programmatic access to much of the functionality available through the Azure portal

Dynamics CRM
Access the capabilities of CRM business software and ERP systems

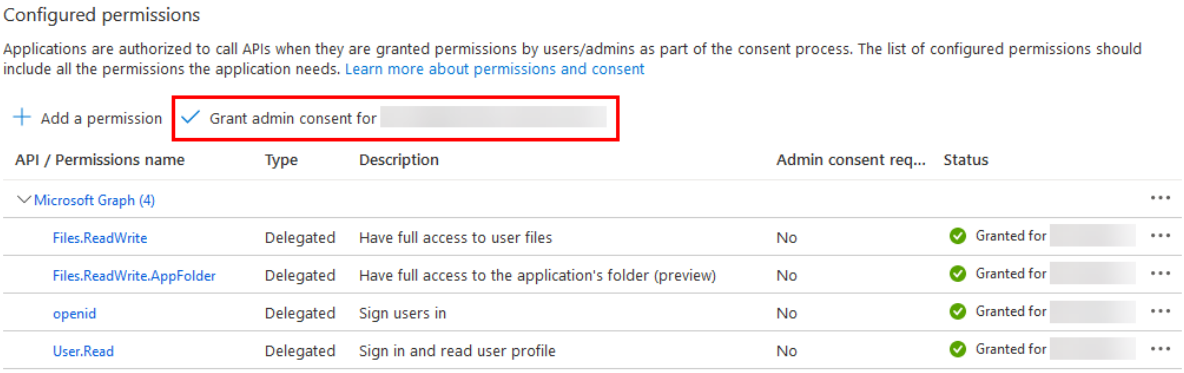
Flow Service
Embed flow templates and manage flows

3. Select *Add a permission* > *Microsoft APIs* tab > *Microsoft Graph*.

4. Select *Delegated permissions* in *Microsoft Graph*.

5. Add the following permissions:

- *Files.ReadWrite*
- *Files.ReadWrite.AppFolder*
- *openid*
- *User.Read*



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

API / Permissions name	Type	Description	Admin consent req...	Status
Microsoft Graph (4)				
Files.ReadWrite	Delegated	Have full access to user files	No	✓ Granted for [redacted]
Files.ReadWrite.AppFolder	Delegated	Have full access to the application's folder (preview)	No	✓ Granted for [redacted]
openid	Delegated	Sign users in	No	✓ Granted for [redacted]
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for [redacted]

1. Select *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 …*.

Configure platform

1. Select *Authentication* in the sidebar.
2. Under *Platform configurations*, click *Add* > *Web*.
3. Enter a redirect URI. Example:

```
https://<ELO>:<port>/ix-<repository>/plugin/de.elo.ix.plugin.auth/logincb/
```

1. Enter a URL for front-channel logout. Example:

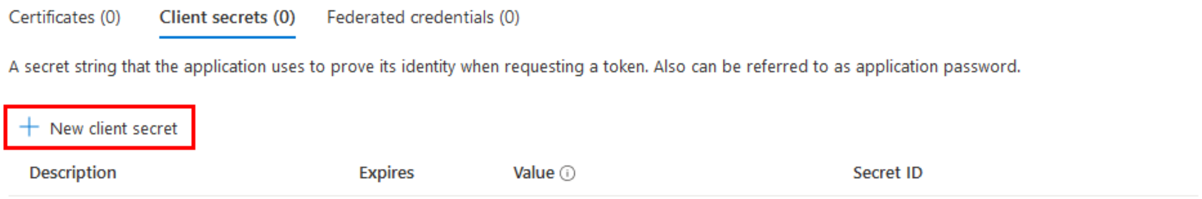
```
https://<ELO>:<port>/ix-<repository>/plugin/de.elo.ix.plugin.auth/logoutcb/
```

1. Click *Configure* to save your entries.

Configure client secrets

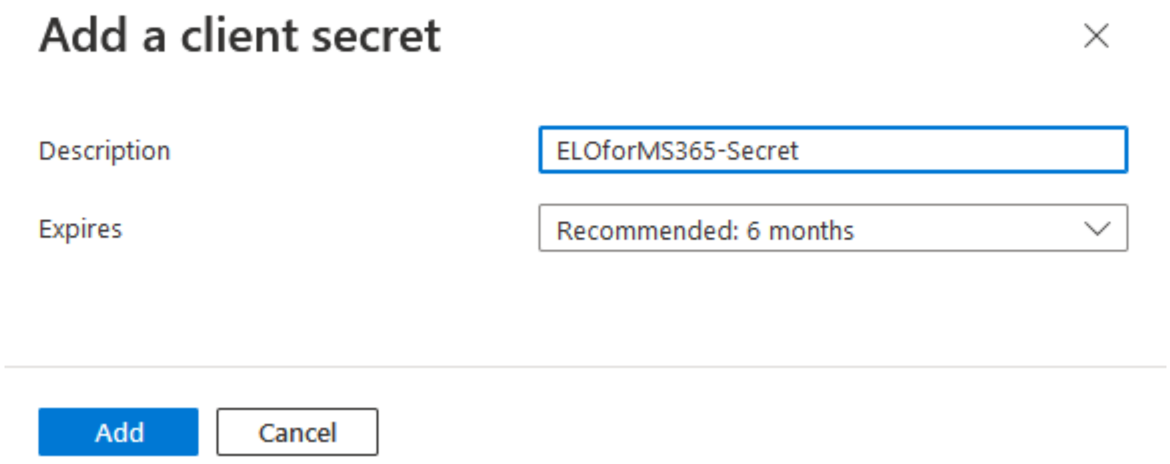
1. Select *Certificates & secrets* in the sidebar.

2. Select the *Client secrets* tab.



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

3. Click *New client secret*.



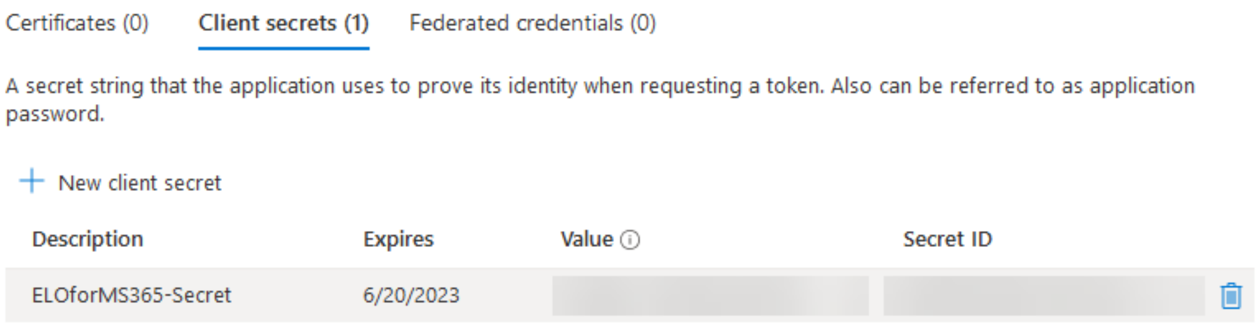
Add a client secret

Description: ELOforMS365-Secret

Expires: Recommended: 6 months

The *Add a client secret* window opens.

4. Enter a name for the client secret in the *Description* field and click *Add*.



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
ELOforMS365-Secret	6/20/2023	[redacted]	[redacted]

The newly created client secret is now configured for your app.

> **Please note**
>
> Write down the value of the secret immediately after you create it. This value is no longer shown in its entirety when you open the overview of secrets at a later point in time.
>
> You will need this value to configure the ELOauth plug-in (see the section Configure ELOauth plug-in).

Configure plug-ins

Logging on ELO with a Microsoft account requires two plug-ins for ELOix:

- *auth-plugin-<version number>.jar*
- *o365sync-plugin-<version number>.jar*

The ELO Server Setup installs these plug-ins to the directory *\<ELO>\prog\webapps\ix-plugins*.

To update these plug-ins, proceed as follows:

1. Download the plug-ins from the ELO SupportWeb.

2. Open the *\<ELO>\prog\serversetup2\repository* folder.

3. Store the files *auth-plugin-<version number>.jar* and *o365sync-plugin-<version number>.jar* in this folder.

4. Switch to the *\<ELO>\prog\serversetup2* folder.

5. Run the *setup.bat* file.

Configure ELOauth plug-in

The connection between ELO and the Microsoft Azure app is established via the *de.elo.ix.plugin.auth.json* file. You will learn how to configure this file in the following.

1. Open the *\<ELO>\config\ix-<repository>\ELO-SRV01-1* folder.

2.

Create the file *de.elo.ix.plugin.auth.json*.

Alternative: If the file already exists, open it.

3. Add the details of your newly registered app to the file.

   The following example shows a possible configuration of the JSON file:

```
{
  "<App name or config ID>": {
    "mapping": "mail",
    "api": "azure",
    "appKey": "<App ID>",
    "appSecret": "<Value of the client secret>",
    "azureTenant": "<Directory ID>",
    "fixedCallbackUrl": "https://<ELO>:<port>/"
  }
}
```

For more information on configuring the plug-in, refer to the ELOauth plug-in – existing implementations documentation.

1. Save the file and restart the ELO Indexserver.

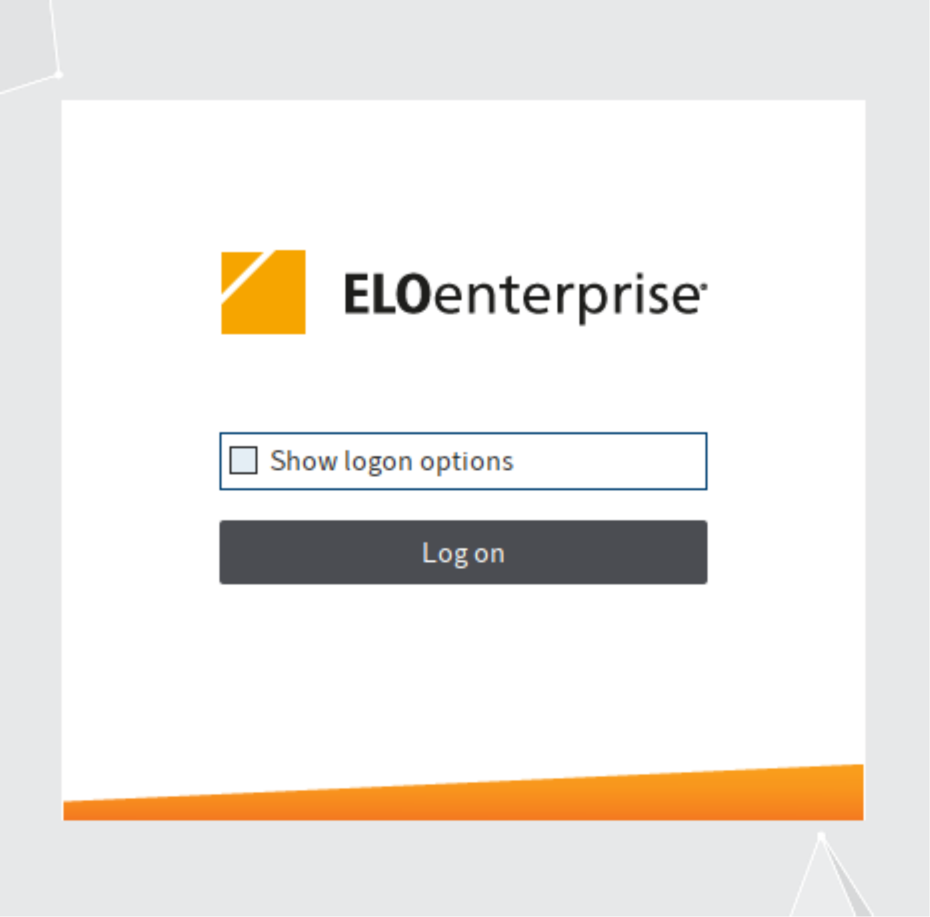
Connect to ELO clients

To enable logon via Microsoft 365, you will have to set up authentication via the ELOauth plug-in in the ELO clients.

ELO Java Client and ELO Desktop Client

1. Create a new profile in the ELO client.
2. In this profile, configure a URL following this pattern:

```
https://<ELO>:<port>/ix-<repository>/plugin/de.elo.ix.plugin.auth/login/?clientUrl=native&c
```



When this profile is selected, the fields for the user name and password are no longer shown in the ELO client logon dialog box. Users click *Log on* and then log on with their Microsoft account data in a new dialog box.

ELO Web Client

1. Open the ELO Web Client status page.
2. Click the *Change settings* link.

   The *ELO Web Client settings* page appears.
3. Select the option *useELOauth* from the drop-down menu.
4. In the *Value* field, enter the name for your registered app, which is also configured in the *de.elo.ix.plugin.auth.json* file.
5. Select *Save*.

When users open the ELO Web Client, they will automatically be redirected to the Microsoft logon page. There, they log on with their Microsoft account data.