Microsoft Entra ID SSO for Portainer

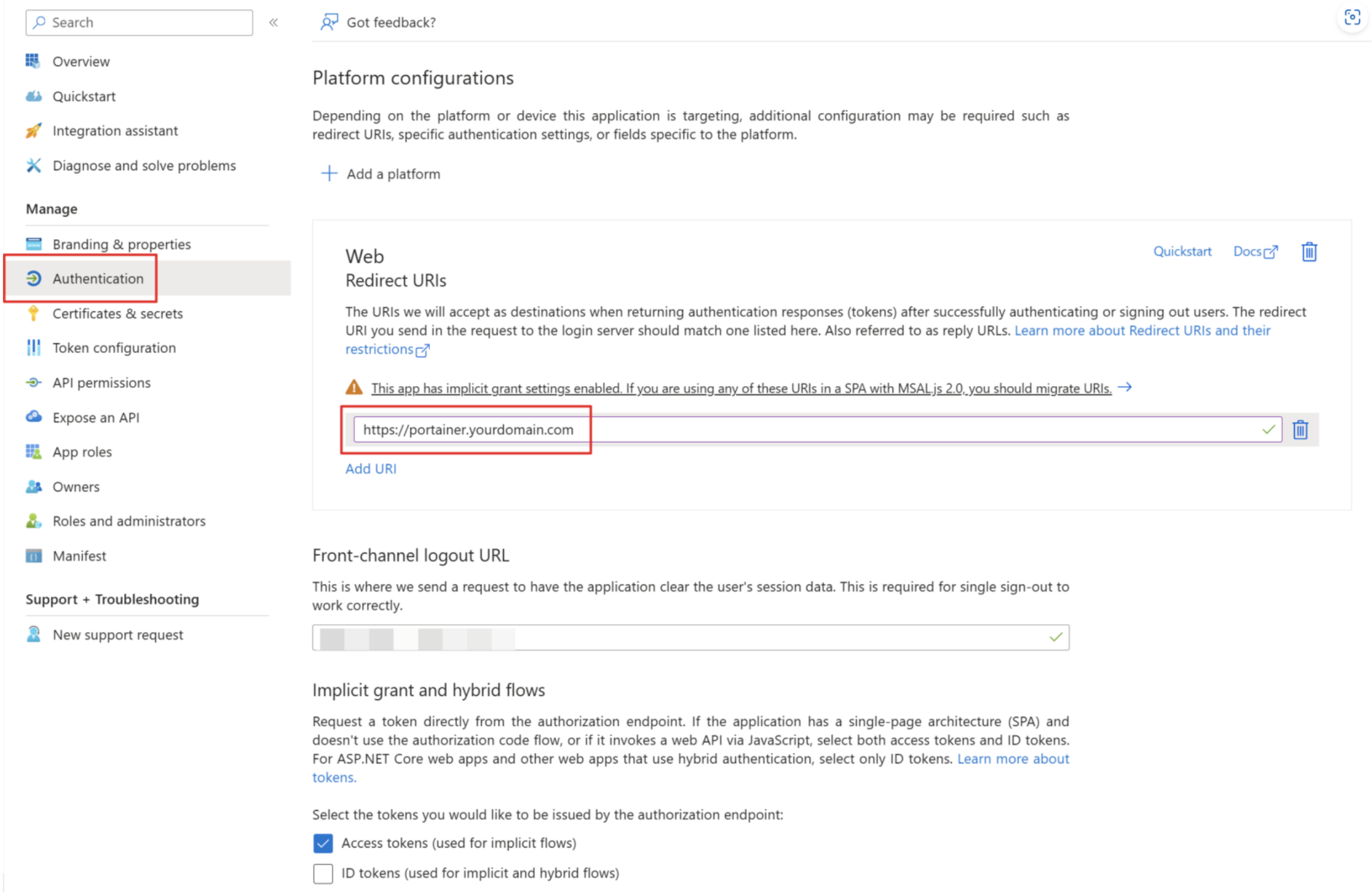
> **Prerequisites:** Ability to create an app registration with delegated standard rights. Portainer should be installed and administrator access to the web interface should be available.

This guide will take you through the various stages of installing SSO using Microsoft Entra ID. Unfortunately, it is not possible to add the SSO functions in the Portainer Community Edition (CE). Accordingly, we have to purchase a free license of the Business Edition (BE) and upgrade the Portainer instance. You will then be able to manage the logins and authorizations for the web interface via Microsoft Entra ID. Functions such as automatic user provisioning or default permissions are also supported by Portainer.

Create App Registration

First, an app registration including client secret must be created in Microsoft Entra ID. All settings can be left at the default values. Important settings are the Redirect URIs under the Authentication tab. Set these URIs to your external or internal domain on which Portainer is available. These URIs will be used for Microsoft Entra ID to know where to redirect the user in case of successful logins.

- **Authentication Type:** Web
- **Redirect URIs:** https://portainer.yourdomain.com



Platform configurations

Depending on the platform or device this application is targeting, additional configuration may be required such as redirect URIs, specific authentication settings, or fields specific to the platform.

Web

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating or signing out users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. Learn more about Redirect URIs and their restrictions

This app has implicit grant settings enabled. If you are using any of these URIs in a SPA with MSAL.js 2.0, you should migrate URIs. →

https://portainer.yourdomain.com

Front-channel logout URL

This is where we send a request to have the application clear the user's session data. This is required for single sign-out to work correctly.

[redacted]

Implicit grant and hybrid flows

Request a token directly from the authorization endpoint. If the application has a single-page architecture (SPA) and doesn't use the authorization code flow, or if it invokes a web API via JavaScript, select both access tokens and ID tokens. For ASP.NET Core web apps and other web apps that use hybrid authentication, select only ID tokens. Learn more about tokens.

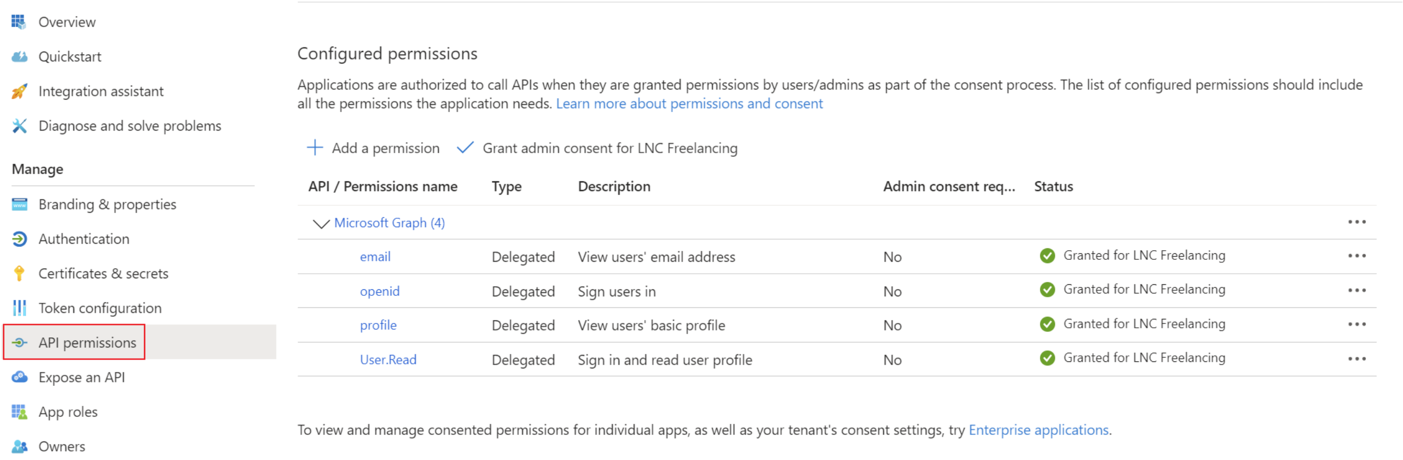
Select the tokens you would like to be issued by the authorization endpoint:

☑ Access tokens (used for implicit flows)

□ ID tokens (used for implicit and hybrid flows)

Add the corresponding permissions for OpenID Connect as delegated permissions and grant admin consent for your tenant.

- **Permissions:** Delegated OpenId permissions (email, offline_access, openid, profile)



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)				
email	Delegated	View users' email address	No	✓ Granted for LNC Freelancing
openid	Delegated	Sign users in	No	✓ Granted for LNC Freelancing
profile	Delegated	View users' basic profile	No	✓ Granted for LNC Freelancing
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for LNC Freelancing

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

Create a client secret for the application and save the tenant ID, application ID and client secret in your password manager. You can find instructions for this information here: Get app details and grant permissions to app registration

Acquire Portainer License

To activate the SSO functionality, the Portainer Comunity Edition must be replaced by a Business Edition. Don't worry, it costs nothing. At least for a Homelab environment and installation with less than 3 environments. For business customer licensing in Portainer Business Edition is based on the number of nodes you are managing.

Create an account and follow the instructions on this page: Take 3 - Get your first 3 nodes free (portainer.io)

Upgrade Portainer instance

After you have received the license key via email you can start upgrading your Portainer CE to a Portainer BE instance. Make sure to create a backup before you upgrade the instance.

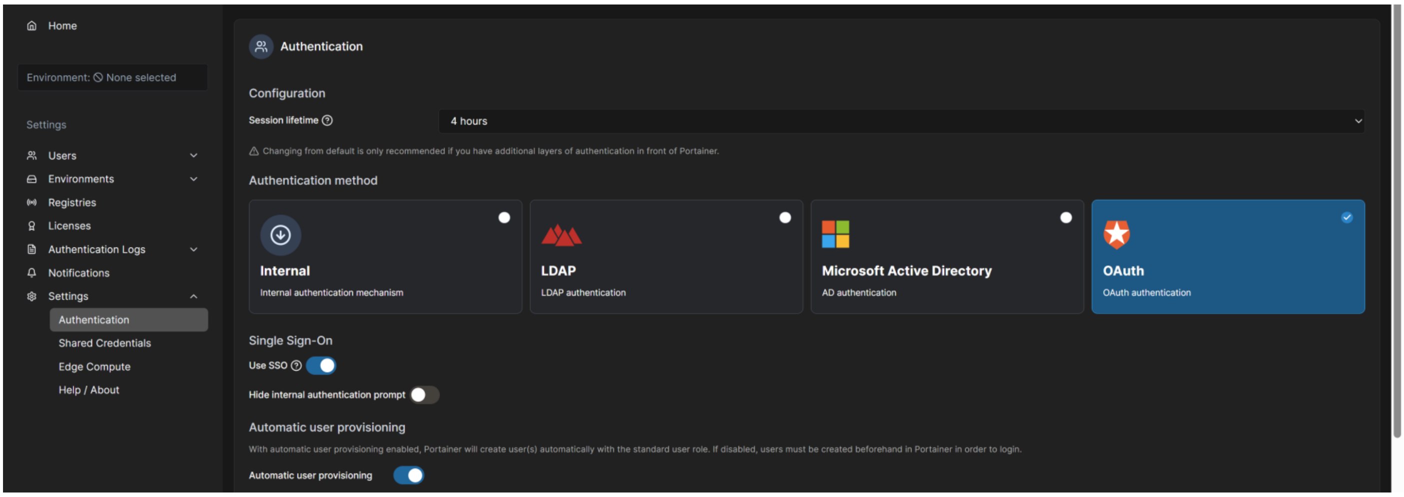
When running portainer inside a single docker container, it is no simpler than changing the image from "portainer/portainer-ce" to "portainer/portainer-ee" and restart the stack or containers. You can find more in depth guides and version specific upgrade manuals here: Docker Standalone | Portainer Documentation

Check activation status

After installing the license, this can be checked in the web interface. To do this, navigate to “Licenses” and check whether your license is installed and the limitation to 3 nodes is displayed. Then you are ready to add external authentication providers such as Microsoft Entra ID. Go to the next step.

Setup Microsoft Entra ID login provider

For the setup, log into the web interface with an administrator account. You can then select the “OAuth” option under Settings -> Authentication -> Authentication method. The following settings enable automatic user provision or a default group. Configure this as it suits you. **Attention:** It is recommended that the option “hide internal authentication prompt” is not activated so that the values can still be adjusted in the event of a misconfiguration of the OAuth provider settings. If this is activated, you can lock yourself out and have to rebuild the Portainer instance.



Authentication

Configuration

Session lifetime: 4 hours

⚠ Changing from default is only recommended if you have additional layers of authentication in front of Portainer.

Authentication method

- ○ **Internal** – Internal authentication mechanism
- ○ **LDAP** – LDAP authentication
- ○ **Microsoft Active Directory** – AD authentication
- ● **OAuth** – OAuth authentication

Single Sign-On

Use SSO: on

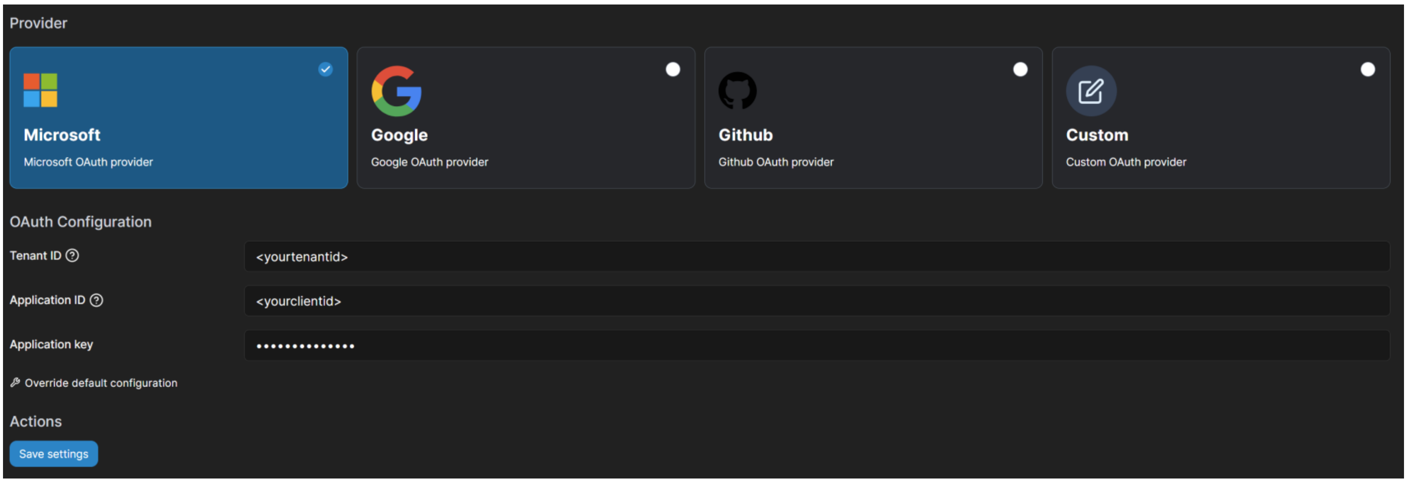
Hide internal authentication prompt: off

Automatic user provisioning

With automatic user provisioning enabled, Portainer will create user(s) automatically with the standard user role. If disabled, users must be created beforehand in Portainer in order to login.

Automatic user provisioning: on

You can then select the "Microsoft OAuth provider" under “Provider” and fill in the TenantID, ClientID and Client Secret options with the corresponding values from the app registration. After saving the settings, you can control who can connect to the web interface of the Portainer instance via the app registration members.



Provider

- ● **Microsoft** – Microsoft OAuth provider
- ○ **Google** – Google OAuth provider
- ○ **Github** – Github OAuth provider
- ○ **Custom** – Custom OAuth provider

OAuth Configuration

Tenant ID: <yourtenantid>

Application ID: <yourclientid>

Application key: ••••••••••••••

Revision #7
Created 19 July 2024 14:45:44
Updated 24 July 2024 10:23:29 by Luca Noah Caprez