

Azure AD- client manual

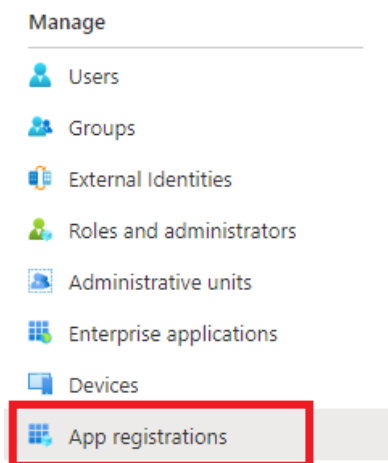
Setting up Azure AD app for integration with Axxerion

Disclaimer

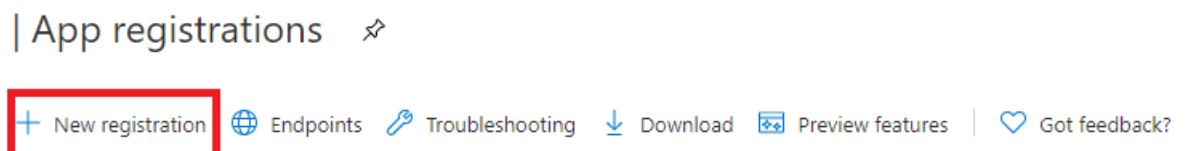
In the below manual, the necessary steps for setting up the Axxerion app in an Azure environment are given. In some of these steps, granting access and adding rights are described. We do not take responsibility for performing these actions, as the person performing these accounts should be aware about their company policies in regards to these actions. If for whatever reason these actions are not allowed due to company policy, please contact the Spacewell representative to discuss the options.

Necessary steps for setting up the Axxerion app

1. Login to Azure portal as administrator
2. Click on “App registrations” in the menu on the left



3. Click on “New registration”



4. In the “Register an application” screen, fill in a name and leave everything else as it is (in the below image, the yellow box should read the name of your organization). Click on “Register”.

Register an application

* Name

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

Axxerion

Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory 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.

Web e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applicat](#)

By proceeding, you agree to the [Microsoft Platform Policies](#)

Register

5. Click on “View API permissions”

Call APIs



Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

[View API permissions](#)

6. Click on “Add permission”

[+ Add a permission](#) ☒ Grant admin consent for [redacted]

API / Permissions name	Type	Description
▼ Microsoft Graph (1)		
User.Read	Delegated	Sign in and read user profile

7. Select “Microsoft Graph”

Select an API

Microsoft APIs APIs my organization uses My APIs

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



Office 365 Management APIs

Retrieve information about user, admin, system, and policy actions and events from Office 365 and Azure AD activity logs

8. Select “Application permissions”

Request API permissions



[← All APIs](#)



Microsoft Graph

<https://graph.microsoft.com/> [Docs](#) [↗](#)

What type of permissions does your application require?

Delegated permissions

Your application needs to access the API as the signed-in user.

Application permissions

Your application runs as a background service or daemon without a signed-in user.

9. Search for “user” and in the results, open the header “user”. Select the “User.Read.All permission” and press on “Add permissions”

Select permissions

Permission
> IdentityRiskyUser
> IdentityUserFlow
> TeamsAppInstallation
> TeamsTab
> UserAuthenticationMethod
> UserNotification
> UserShiftPreferences
> User

☒	User.Read.All ⓘ Read all users' full profiles	Yes
☐	User.ReadWrite.All ⓘ Read and write all users' full profiles	Yes

Add permissions

Discard

10. Search for “group” and in the results, open the header “Group”. Select the “Group.Read.All permission” and press on “Add permissions”

11. Press on “Grant admin consent for...” to allow the necessary access. Make sure the permissions look like those below.

Axxerion | API permissions

Search (Ctrl+/) Refresh Got feedback?

Overview Quickstart Integration assistant

Manage

- Branding & properties
- Authentication
- Certificates & secrets
- Token configuration
- API permissions**
- Expose an API
- App roles
- Owners
- Roles and administrators | Preview

The “Admin consent required” column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value organizations where this app will be used. [Learn more](#)

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](#)

+ Add a permission **✓ Grant admin consent for Axxerion WebService**

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (3)				
Group.Read.All	Application	Read all groups	Yes	✓ Granted for Axxerion WebService ...
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for Axxerion WebService ...
User.Read.All	Application	Read all users' full profiles	Yes	✓ Granted for Axxerion WebService ...

12. In the menu, navigate to “Certificates & secrets” and click on “New client secret”

Manage

- Branding
- Authentication
- Certificates & secrets**
- Token configuration
- API permissions
- Expose an API
- App roles | Preview
- Owners
- Roles and administrators | Preview
- Manifest

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

Upload certificate

Thumbprint	Start date	Expires	ID
No certificates have been added for this application.			

Client secrets

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

+ New client secret

Description	Expires	Value	ID
No client secrets have been created for this application.			

13. Fill in the necessary details (we advise that the secret never expires, but leave that decision up to the client/supplier – if a different value is used, we need to receive a new secret in time to make sure the connection keeps working)

Add a client secret

Description

Axxerion webservice

Expires

☐ In 1 year

☐ In 2 years

☒ Never

Add Cancel

14. Copy the value that has been created and make sure to somewhere as you cannot retrieve it afterwards. If this goes wrong, remove the client secret and perform the above actions again.

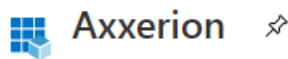
Client secrets

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

+ New client secret

Description	Expires	Value	ID
Axxerion webservice	12/31/2299	mFk36-wcL~P~jjOu2UL1_l63Ti-5hlo~2e	8773e7bb-19f9-4995-a490-568a55eb0135

15. Share with Axxerion the following values:
- The client secret that was retrieved in step 13
 - Application (client) ID (see below)
 - Directory (tenant) ID (see below)



Search (Ctrl+/)

Overview

Quickstart

Integration assistant

Manage

Delete Endpoints Preview features

Essentials

Display name : Axxerion

Application (client) ID : be57543d-601a-4d30-af09-a5dea5ee7504

Directory (tenant) ID : 9189ea4d-94ef-4b72-be2a-6f6b71e70db1

Object ID : 2d880c2e-f716-40f1-8ad9-1d493cd3ebae

16. Navigate to groups

Manage

- Users
- Groups
- External Identities

17. Open the group which Axxerion should have access to and copy the objectId. Share this with Axxerion

AX

Axxerion
Axxerion groep

Membership type

Assigned

Source

Cloud

Type

Security

Object Id

acd2bd3b-1890-41a1-a8de-9fc4f3867a45

Creation date

5/11/2020, 8:11:02 PM