Cloud B2B
solutions

Purchase Orders Online

OpenID Connect Module

Single Sign-On with Azure AD

Integration Guide

Overview

The Cloud B2B OpenID Connect module enables PO System users to login with their Azure Active Directory credentials, using a single sign-on authentication process via a special Logon URL endpoint or by publishing an internal App via the Azure Portal for use in the enterprise.

OpenID authentication gives the enterprise greater control over Users online identities whilst minimising the password security risks and the frustration associated with maintaining multiple Usernames and Passwords.

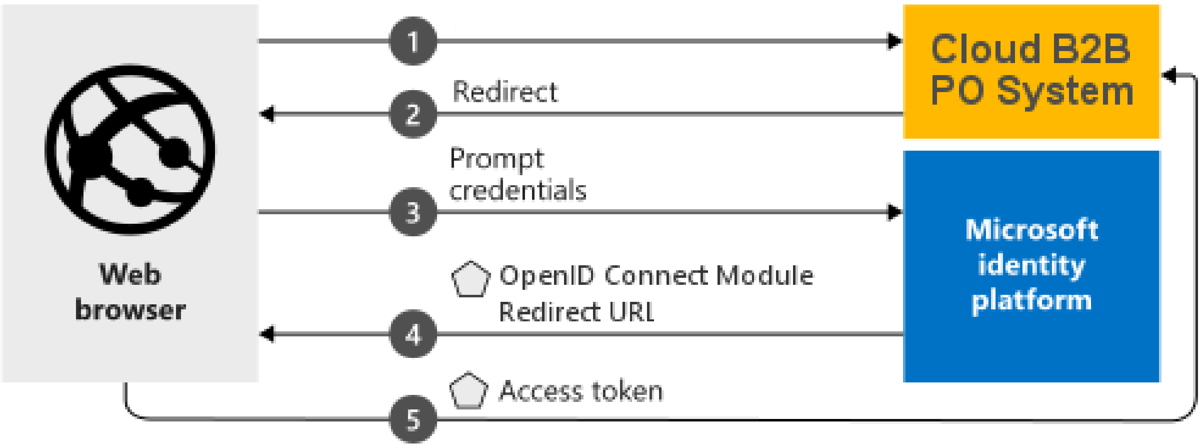
Full user management is handled via the PO System UI, where new Users will be added and Users roles and permissions set. Users that are removed from the Azure Active Directory will also need to be removed from the PO System by an Administrator.

The Cloud B2B OpenID Connect module authenticates Users against the PO System and against the pre-configured App ID and Tennant ID specific to the Azure Active Directory and App instance.

The module matches to the PO System User Account based on the first part of the Users logon name (e.g. example.user@thecompany.com the username would be ‘example.user’) and email address associated with their Azure Active Directory account. If a PO System User Account doesn’t exist, the User will be redirected to the standard PO System logon screen and an error message displayed.

Systems that are configured for OpenID Connect authentication will have the PO System self-service password reset facility and the ability for Users to change their PO System password disabled (account status, lockout and password resets will be handled by Azure Active Directory).

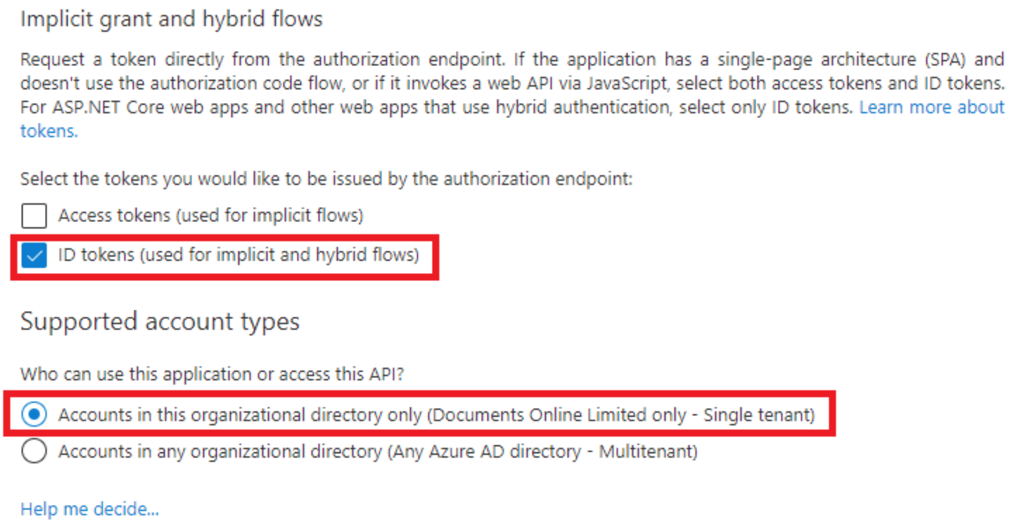
Authentication flow



Setting up the Azure App (completed by client)

Step 1

1. Sign in to the Azure portal.
2. If you have access to multiple tenants, use the **Directory + subscription** filter in the top menu to switch to the tenant in which you want to register the application.
3. Search for and select **Azure Active Directory**.
4. Under **Manage**, select **App registrations** > **New registration**.
5. For **Name**, enter a name for your application. For example, enter **PO System**. Users of your app will see this name when publishing the App to the enterprise.
6. Set the **Redirect URL** type to **Web** and value to the URL provided by Cloud B2B (your custom OpenID Connect URL Endpoint).
7. Select **Register**.
8. Under **Manage**, select **Authentication**.
9. Under the **Implicit grant and hybrid flows** section, select **ID tokens**.
10. Under the **Supported account types** section, select **Accounts in this organizational directory only** option (see below screenshot).
11. Click **Save**.



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)

Supported account types

Who can use this application or access this API?

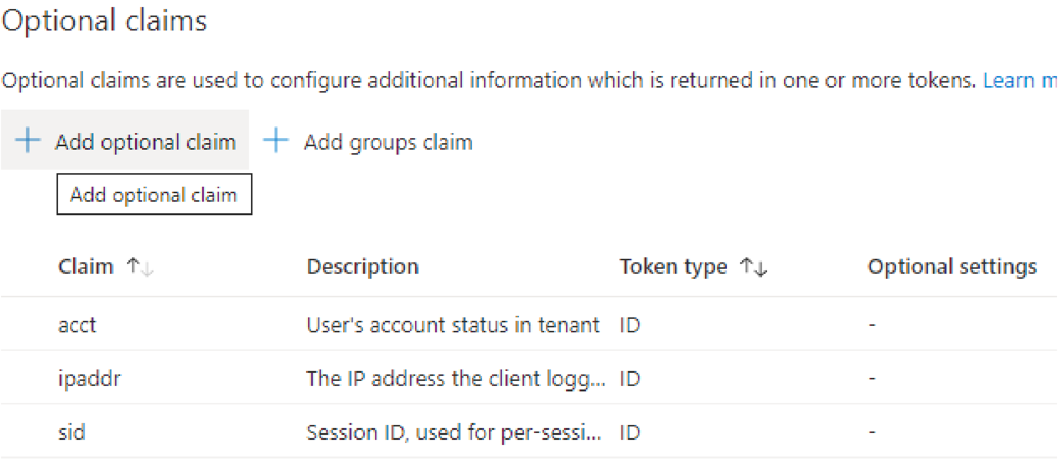
● Accounts in this organizational directory only (Documents Online Limited only - Single tenant)

○ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

Help me decide...

Step 2

1. On the **Token configuration** blade select **Add optional claim.**
2. Select **Token type** of **ID** option.
3. Select (tick) the following **claims** from the list:
    a. acct
    b. ipaddr
    c. sid
4. Click **Add**.



Optional claims

Optional claims are used to configure additional information which is returned in one or more tokens. Learn m

Claim ↑↓	Description	Token type ↑↓	Optional settings
acct	User's account status in tenant	ID	-
ipaddr	The IP address the client logg...	ID	-
sid	Session ID, used for per-sessi...	ID	-

Step 3

1. Under the **Azure Active Directory** pane select **Enterprise applications** and select the App registration created in step 1.
2. Under **Manage** select **Properties**.
3. Select **Yes** for **Assignment required?**
4. Select **Yes** for **Visible to users?**
5. Click **Save**.

Assignment required? ⓘ [✓ Yes] No

Visible to users? ⓘ [✓ Yes] No

Step 4

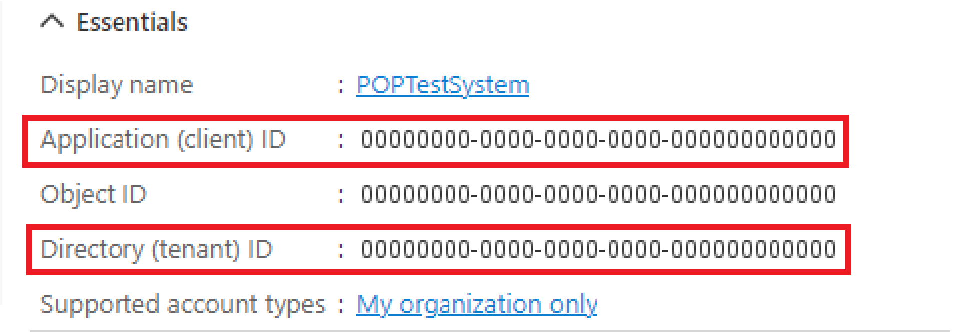
Assign required Azure Active Directory Users, Groups or Roles to the App registration, for more information on App registration access see Azure Docs – Assign a user account to an enterprise application.

Step 5

1. Locate the **Application (client) ID** which can be found on the app's **Overview** blade in **App registrations** for the App that was created in step 1.

2. Locate the **Directory (tenant) ID** which can be found on the app's **Overview** blade in **App registrations** for the App that was created in step 1.

3. Provide both the **Application (client) ID** and the **Directory (tenant) ID** to Cloud B2B so the configuration of your Cloud B2B OpenID Connect module can be completed ready for testing.



^ Essentials

Display name	:	POPTestSystem
Application (client) ID	:	00000000-0000-0000-0000-000000000000
Object ID	:	00000000-0000-0000-0000-000000000000
Directory (tenant) ID	:	00000000-0000-0000-0000-000000000000
Supported account types	:	My organization only

Notes

- The OpenID Connect module doesn’t support automatic User management (creation/removal) in the PO system. All user management must be completed using the PO System Admin UI with matching Username and Email Address for the User.

  When new Users are on-boarded they should be added to the relevant Azure Active Directory groups and created in the PO System using the Admin UI. The User should be created with a secure password that isn’t shared with the User, the password can be set and forgot (unless the User should also be able to logon directly).

  When existing Users leave or need to be removed from the PO System, they should be removed from the Azure Active Directory groups and their account in the PO System should be disabled (with designated leaver account for future PO tasks).

- Users attempting to access the PO System via the Azure Active Directory App will be redirected to the standard system logon screen with a logon error message if a PO System User Account doesn’t exist or their account is disabled.

- User session management is fully handled by the PO System and each concurrent session will require an available Concurrent User license. Once the User is authenticated into the PO System, their session is fully handled by the PO System only. Revoking User sessions within Azure won’t terminate active PO System sessions for the User, instead the User Account should be deactivated in the PO System.

- User Password Management (i.e. self-service password reset and change password tool) will be disabled in the PO System as all authentication will be handled by Azure Active Directory and the Microsoft identity platform.

- The App registration in Azure must be configured to only allow **Accounts in this organizational directory only** to be authenticated, otherwise authentication will fail.

- During setup of the OpenID Connect module, a test version of the module will be deployed connected to a Test instance of the PO System, so a Test App registration can be setup (by client) for distribution to test Users.

Cloud B2B
solutions

7 Venture Court, Edison Road,
St. Ives, Cambs, PE27 3JX

Freephone 0800 840 3336
sales@cloudb2b.co.uk
cloudb2b.co.uk