Technical guide Local OIDC Server

Endpoints

The local/oidcserver server uses endpoints to answer the requests of OAuth clients:

- authorize_endpoint : <wwwroot>/local/oidcserver/endpoints/authorize.php
- token_endpoint : <wwwroot>/local/oidcserver/endpoints/token.php
- userinfo_endpoint : <wwwroot>/local/oidcserver/endpoints/userinfo.php
- revoke_endpoint : <wwwroot>/local/oidcserver/endpoints/revoke.php
- logout_endpoint : <wwwroot>/local/oidcserver/endpoints/logout.php
- jwks : <wwwroot>/local/oidcserver/endpoints/jwks.php
- config : <wwwroot>/local/oidcserver/endpoints/config.php

Flows

Authorization endpoint

This endpoint receives the authorization requests issued by clients, and delivers a temporary authorization token if the client has been recognised in the server configuration. The authorization request is sent by a user agent from the end user's workstation.

The authorization server checks whether a session is already active for the end user (based on the session cookie associated with the authentication server).

If no session is detected, the authorization server redirects the end user to its login sequence. The authorization request is remembered during this exchange, so that it can be processed at the end of the process. The user logs in on the authorization server, which creates a session, and checks the user's eligibility conditions (right to be authorized for this client). The last step is triggered.

If a session has been detected, the user is considered already logged in and the next step is automatically triggered.

At the end of processing, the authorization server sends the user agent back to an entry of the resource server able to process the authorization token and continue the authentication exchanges.

Access token endpoint

This endpoint answers a resource server to validate access, by checking the integrity and “sincerity” of the authorization token that the user agent sends it as a bounce of the authorization request.

The authorization token is carried by an `Authorize:` header, in the form of a “Bearer <crypted_token>” token.

The authorization server confirms access by returning an access token (OAuth2). The local/oidcserver server supports the additional OIDC layer by sending the “scopes” required by an OIDC identity transmission.

User info endpoint

This endpoint provides an extended metadata sheet about the user, according to the disclosure rules configured in the authorization server, and/or the transmission “acceptances” of the user who owns the data.

Revocation endpoint (revoke)

This endpoint is used by a client to log out of the OAuth2 session and force the loss of validity of its associated authorization token. Revocation alone is not an act of logging out of the federation. Other sessions of tools tied to the federation may stay connected, as well as the main session of the identity provider.

Federated logout endpoint (logout)

This endpoint receives a logout request coming from the resource server, when the end user logs out locally from the latter. The session present on the authorization server is destroyed. The user concerned is recognised by the value of the `Authorize:` header field, which must carry the access token.

The authorization server keeps a memory of the “active” access tokens. If, during a logout request, other active tokens are identified for the user, then the local/oidcserver server will issue logout requests to the logout endpoints of the client resource servers.

The client logout request will also carry the access token (Bearer) in the `Authorize:` header field

Configuration endpoint (config)

This endpoint allows reading the “public” configuration of the server without authentication.

JWKS configuration endpoint

This endpoint allows reading the configuration in JWKS (JSON Web Key Set) format without authentication. Among other things, it allows any third party to retrieve the public keys needed to talk to the server securely.

Plugin summary - Plugins index - Catalogue home

From:
https://docsen.activeprolearn.com/ - **Documentation Moodle ActiveProLearn**

Permanent link:
https://docsen.activeprolearn.com/doku.php?id=local:oidcserver:technique

Last update: **2026/11/13 21:59**