OAUTH2 REFERENCE DOCUMENT

© 2018 Esselte IPR AB. All rights reserved.

OAuth2 API Reference

The Leitz Cloud API (v2+) requires authentication via OAuth2. Currently it supports the *password* and *refresh token* grant types, with *authorization code* and *client credential* grants coming in the near future.

This document will not go into the details of the OAuth2 specification, but instead will focus on simple steps for using it with Leitz Cloud. Leitz Cloud's implementation of OAuth2 is based on RFC 6749.

Endpoint

```
https://web.leitz-cloud.com:<port>/oauth
```

All requests to the OAuth2 API must be performed over HTTPS. Non-HTTPS requests will be rejected.

Request an access token

To request an access token, call the `token` method, passing account credentials to authenticate.

```
POST /oauth/token
```

POST Fields

- **grant_type**: the requested grant type, "password" in this case. *Required*.
- **client_id**: the OAuth2 client ID, "anchor" only at this time. *Required*.
- **username**: the username for the account being authenticated. *Required*.
- **password**: the password for the account being authenticated. *Required*.
- **auth_code**: the two-step verification code for the account being authenticated. Required if the account has two-step verification enabled.
- **guid**: the GUID assigned to the client or empty if one has not yet been assigned.
- **dns_name**: a descriptive name for the client, such as the host name or device manufacturer and model.
- **os_type**: the client system type, such as "win", "osx", "android", "ios", or "winphone".
- **os_version**: the client system version, such as "4.2.2".

Example

Request

```
POST /oauth/token

grant_type=password&client_id=anchor&username=user@example.com&password=example
```

© 2018 Esselte IPR AB. All rights reserved.

Response

```
HTTP status code: 200
Response body:
{
    "access_token": "<access_token>",
    "expires_in": 3600,
    "guid": "<guid>",
    "token_type": "Bearer",
    "refresh_token": "<refresh_token>",
    "scope": "full"
}
```

The response should be saved. The access token will be passed to API methods requiring authentication. Access tokens are valid for a short period of time, usually 1 hour, and the refresh token will be used to get a new access token when it expires. The `expires_in` value is in seconds. `token_type` and `scope` will always be "Bearer" and "full," respectively.

The `guid` returned in the response is either the GUID passed in the original request, if it is found on the server, or a new GUID assigned to the client. It should be saved for future use when refreshing tokens.

Two-step verification

When an account has two-step verification enabled, the initial call will return the following:

```
HTTP status code: 401
Response body:
{
    "error": "missing_totp",
    "two_step_mode": "(email|sms|authenticator)"
}
```

In this case, the user should be prompted for a two-step verification code and the request should be repeated with the code appended. The Leitz Cloud server will have already triggered the notification process in the case of email or SMS delivery.

```
POST /oauth/token
grant_type=password&client_id=anchor&username=user@example.com&password=exa
mple&auth_code=<auth_code>
```

© 2018 Esselte IPR AB. All rights reserved.

If the code is valid, an access token will be returned. Otherwise, the following:

```
HTTP status code: 401
Response body:
{
    "error": "invalid_totp",
    "two_step_mode": "(email|sms|authenticator)"
}
```

If repeated failed authentication attempts occur, the account will be blocked from making more authentication attempts for a short period:

```
HTTP status code: 403
Response body:
{
    "error": "account_locked"
}
```

Refresh an access token

Refreshing an access token also uses the same `token` method as the initial authentication request. In this case, the `refresh_token` from a previous access token request is passed instead of account credentials.

```
POST /oauth/token
```

POST Fields

- **grant_type**: the requested grant type, "refresh_token" in this case. *Required*.
- **client_id**: the OAuth2 client ID, "anchor" only at this time. *Required*.
- **refresh_token**: a refresh token from the previous access token request. *Required*.
- **guid**: the GUID assigned to the client or empty if one has not yet been assigned. Typically this should be the GUID assigned by the server during the initial access token request.
- **dns_name**: a descriptive name for the client, such as the host name or device manufacturer and model.
- **os_type**: the client system type, such as "win", "osx", "android", "ios", or "winphone".
- **os_version**: the client system version, such as "5.1.0".

Example

Request

```
POST /oauth/token
grant_type=refresh_token&client_id=anchor&refresh_token=<refresh_token>
```

© 2018 Esselte IPR AB. All rights reserved.

Response

```
HTTP status code: 200
Response body:
{
    "access_token": "<access_token>",
    "expires_in": 3600,
    "guid": "<guid>",
    "token_type": "Bearer",
    "refresh_token": "<refresh_token>",
    "scope": "full"
}
```

Revoke access

To invalidate an access token, call the `revoke` method.

```
POST /oauth/revoke
```

POST Fields

- **client_id**: the OAuth2 client ID associated with the token, "anchor" only at this time. *Required*.
- **token**: the token to revoke. *Required*.
- **token_type_hint**: the type of token being submitted, "access_token" or "refresh_token".

Example

Request

```
POST /oauth/revoke
client_id=anchor&token=<token>&token_type_hint=access_token
```

Response

```
HTTP status code: 200
```

For valid requests the response is always HTTP 200. The response body is ignored.

© 2018 Esselte IPR AB. All rights reserved.

Authenticating requests

Once an access token is obtained, it may be used to authenticate requests by passing it in the Authorization header.

Example request

```
GET /api/2/files/1

Authorization: Bearer <access_token>

...
```

© 2018 Esselte IPR AB. All rights reserved.