General

XMLdation API supports authentication with OAuth 2.0 with OpenID. From authentication response user must use "id_token". id_token is signed JWT token which includes required claims for accesssing resources. Authorization header must be set to all requests made to the XMLdation API. API functionality is enabled per user and is managed by XMLdation. If you need API usage privileges, please contact XMLdation at xmldation@xmldation.com

OAuth Token Endpoint URL

Production API

https://auth.xmldation.com/oauth2/token

OAuth 2.0 Bearer Token Authentication

XMLdation API uses OAuth 2.0 Bearer Token Authentication to authenticate the valid user account. Following header must be passed to all requests (excluding the token retrieval):

Authorization: Bearer <id_token>

Generating OAuth Token

OAuth Token is a key to autenticate user to API Service. Same username and password combo is used to access XMLdation Service and API Service OAuth Token retrieval. OAuth Token is always defined per user account. User can have one OAuth Token active per time.

curl --user ***ClientId***:***ClientSecret*** -k -d "grant_type=password&username=xmldation/***Username***&password=***Password***&scope=openid" -H "Content-Type:application/x-www-form-urlencoded" https://auth.xmldation.com/oauth2/token

Where

- ***ClientId*** is provided by XMLdation
- ***ClientSecret*** is provided by XMLdation
- ***Username*** is XMLdation Service username (URLencoded)
- ***Password*** is XMLdation Service password (URLencoded)

Example of authentication response

curl -u ClientId:ClientSecret -d "grant_type=password&username=xmldation/Username&password=Password&scope=openid" -H "Content-Type:application/x-www-form-urlencoded" https://auth.xmldation.com/oauth2/token


```
{
 "expires_in": 3293,
 "access_token": "ef1034a6-f984-3666-93d4-952b186a8b89",
 "refresh_token": "72fec40a-3eb9-3aff-a321-0d954f44d85c",
 "token_type": "Bearer",
```

```
 "scope": "openid",
 "id_token": "<LONG JWT Token here>"
}
```


Error messages

In case of authentication failure the API will response with HTTP 401 response code. Also error message body is returned.

Unauthorized error message is returned in following cases:

- Authorization header is missing
- Authorization header content is invalid
- Token is expired
- User does not have permissions to given product code (e.g. in /v1/validate/{pcode})

If you you have checked that your authentication details are correct, but you still get error messages, please contact XMLdation for further assistance.

401 unauthorized response body

Response is returned in application/json format.

HTTP/1.1 401 Unauthorized


```
{
 "error" : {
  "status" : 401,
  "message" : "Unauthorized"
 }
}
```