Authorize

URL	https://<baseurl>/authorize
Request method	`GET` with URL query-parameters `POST` with parameters as `application/x-www-form-urlencoded` data
Client Authentication	See supported methods
Request parameters	See below
Response elements	See below
Example	See below

Authorize is a standard endpoint that triggers authentication of an enduser via one of the IDP options, followed by authorization in terms of consent handling. Authorization information is then returned in the reponse to the requesting OIDC Client. The content of the authorization response is different for each of the supported message flows. The Authorize endpoint does in any case trigger a series of re-directs, eventually returning to the requesting OIDC Client at a `redirect_uri` specified by the client. For security reasons only pre-registered redirect URIs are allowed.

Request parameters

✅ = According to standard. ❌ = Feature restriction. ⚠️ = In progress / future support.

Name	Support	Description
`scope`	✅	List of scope values specifying what kind of resources (dataset) the OIDC Client requests access to. The value `openid` must always be included in the list.
`response_type`	✅	Determines the message flow to be used, thus also governing the content and type of the response from the Authorize endpoint. The following combinations are supported: • `"code"` (Authorization Code flow) • `"id_token"` or `"id_token token"` (Implicit flow) • `"code id_token"`, `"code token"`, or `"code id_token token"` (Hybrid flow)
`client_id`	✅	Unique ID (arbitrary string) for the OIDC Client in question. This is created as part of the provisioning process.
`redirect_uri`	✅	Redirect URI to which the Authorize response will be sent. This URI **must** exactly match one of the Redirect URI values for the OIDC Client pre-registered at the OpenID Provider
`state`	✅	Opaque value used to maintain state between the request and the callback.
`response_mode`	✅	The response mode to be used for returning parameters from the Authorization Endpoint. The following values are supported: • `query` • `fragment` • `form_post` **Note**: The .NET/C# example GitHub uses the OWIN framework. OWIN only accepts `form_post` response mode
`nonce`	✅	String value used to associate a ODIC Client session with an ID Token, and to mitigate replay attacks. The value is passed through unmodified from the Authentication Request to the ID Token
`display`	❌	Not supported. The OIDC Provider from BankID does instead support GUI customization and JavaScript connectors to govern how IDP-dialogues are displayed.
`prompt`	⚠️	Specifies whether the Authorization Server prompts the enduser for re-authentication and consent.
`max_age`	❌	Not supported. The OIDC Provider determines life-time values in the ID Token.
`ui_locales`	✅	May be used to set a language preference for GUI handling. The default GUI experience supports nb (Norsk Bokmål) and en (English)
`acr_values`	⚠️	Requests use of any IDP at a given Level of Assurance (Authentication Context Class Reference) or above. A selector dialogue is shown to the enduser if more than one IDP option meet the required minimum level. Note that this parameter has no effect if the `login_hint` parameter contains a reference to any particular IDP. Nor does it have any effect if the `id_token_hint` parameter is set. If none of these parameters are set a selector dialogue is shown contianing all available IDP options.

`login_hint`	✓	This parameter may be used to specify the use of any particularly named IDP (Authentication Method Reference) along with any pre-configuration for the designated IDP. Note that this parameter has no effect f the `id_token_hint` parameter is set. If none of these parameter are set, the `acr_values` parameter determines IDP selection. See further details on login_hint support for each of the supported IDPs.
`id_token_hint`	⚠	An ID Token previously issued by the OIDC Provider used as a hint about the enduser's current or past authenticated session with the OIDC Client. Note that this parameter has precedence before both `acr_values` and `login_hint`. If the ID Token has expired, a new authentication is triggered for the IDP option that was used when the ID Token was issued. Otherwise the authentication is still granted valid and the OIDC Provider proceeds directly to the autorization stage via consent handling.

Response elements

Responses are different for each of the supported message flows as specified by the `response_type` and `response_mode` parameters in the Authorize request.

Authorization Code flow

The standard response for this flow is to add relevant parameters as URL query-parameters to the `redirect_uri`,unless a different `response_mode` was specified. For this flow only `code` is returned in the Authorize response. The Token endpoint must be called to retrive `id_token` and `access_token`.

Implicit Flow

The standard response for this flow is to add relevant parameters to the fragment component of the Redirection URI, unless a different response_mode was specified. For this flow both `id_token` and `access_token` is returned directly in the Authorize response.

Hybrid Flow

The standard response for this flow is to add relevant parameters to the fragment component of the Redirection URI, unless a different response_mode was specified. For this flow `code` and `id_token` is returned in the Authorize response. The Token endpoint must be called to retrive `access_token`.

Example

The following example shows a request for the Authorize endpoint. The example is generated from Postman (which is configured as a client at the OIDC Provider) and correspons to an Authorize Code flow. A minimum value for scope (`scope=openid`) is used in this example. The value for the access token in the authorization header (`Authorization: Bearer` 4497db915b5b479191c81a7854a2fa8) is taken from the corresponding example for the Token endpoint. The OIDC Provider responds with HTTP 302 to redirect the User-Agent to start IDP handling for BankID.

Request

```
GET /oidc/oauth/authorize?client_id=Postman&scope=openid&state=7908648&redirect_uri=https%3A%2F%2Fwww.
getpostman.com%2Foauth2%2Fcallback&response_type=code HTTP/1.1
Host: preprod.bankidapis.no
Connection: close
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_5) AppleWebKit/537.36 (KHTML, like Gecko) Postman/4.
10.7 Chrome/53.0.2785.143 Electron/1.4.12 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-GB

HTTP/1.1 302 Found
Content-Length: 0
Location: https://oidc-preprod.bankidnorge.no/bidview?sid=2b29ac70-800b-4eb5-bf3d-
f0bd61a9e520&oidcAuthenticationUrl=https%3A%2F%2Fpreprod.bankidapis.no%2Foidc%2Fauthenticate%2F
Server: Microsoft-IIS/8.5
X-Powered-By: ARR/3.0
X-Powered-By: ASP.NET
Date: Thu, 25 May 2017 11:08:58 GMT
Connection: close
```

The following intermediate request/response pair shows how the User-Agent returns to the OIDC Provder after IDP handling for BankID. The OIDC provider responds with HTTP 302 to redirect the User-Agent to the OIDC client that originated the Authorize request (Postman in this case).

Intermediate

```
POST /oidc/oauth/authorize?session_authentication_token=2b29ac70-800b-4eb5-bf3d-f0bd61a9e520 HTTP/1.1
Host: preprod.bankidapis.no
Connection: close
Content-Length: 0
Cache-Control: max-age=0
Origin: https://oidc-preprod.bankidnorge.no
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_5) AppleWebKit/537.36 (KHTML, like Gecko) Postman/4.
10.7 Chrome/53.0.2785.143 Electron/1.4.12 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Referer: https://oidc-preprod.bankidnorge.no/bidview/webclient
Accept-Language: en-GB

HTTP/1.1 303 See Other
Content-Length: 0
Location: https://www.getpostman.com/oauth2/callback?state=7908648&code=b860604adbf40f6c53a797290916771
Server: Microsoft-IIS/8.5
X-Powered-By: ARR/3.0
X-Powered-By: ASP.NET
Date: Thu, 25 May 2017 11:09:36 GMT
Connection: close
```

The following request/response pair shows how the originating OIDC Client (Postman in this case) resumes control in terms of a response from the Authorize endpoint.

Response

```
GET /oauth2/callback?state=7908648&code=b860604adbf40f6c53a797290916771 HTTP/1.1
Host: www.getpostman.com
Connection: close
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_5) AppleWebKit/537.36 (KHTML, like Gecko) Postman/4.
10.7 Chrome/53.0.2785.143 Electron/1.4.12 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Referer: https://oidc-preprod.bankidnorge.no/bidview/webclient
Accept-Language: en-GB
Cookie: _ga=GA1.2.129047571.1494663652; _mkto_trk=id:067-UMD-991&token:_mch-getpostman.com-1494663658399-21327

HTTP/1.1 301 Moved Permanently
Content-Type: text/html
Content-Length: 193
Connection: close
Date: Thu, 25 May 2017 11:09:40 GMT
Location: https://app.getpostman.com/oauth2/callback?state=7908648&code=b860604adbf40f6c53a797290916771
Server: nginx/1.10.2
X-Cache: Miss from cloudfront
Via: 1.1 4a74a9a6128ae727659616d5fe9bf745.cloudfront.net (CloudFront)
X-Amz-Cf-Id: L4wexO63ms38xVYDEl42oDlfOIJSDq11yWZciLD3p-NUEt8yFGiVYw==

<html>
<head><title>301 Moved Permanently</title></head>
<body bgcolor="white">
<center><h1>301 Moved Permanently</h1></center>
<hr><center>nginx/1.4.6 (Ubuntu)</center>
</body>
</html>
```