Cheatography

Open ID Connect Cheat Sheet

by iMalignus via cheatography.com/20929/cs/3784/

Overview

The OpenID Connect protocol, in abstract, follows the following steps.

1. The RP (Client) sends a request to the OpenID Provider (OP).
2. The OP authenticates the End-User and obtains authorization.
3. The OP responds with an ID Token and usually an Access Token.
4. The RP can send a request with the Access Token to the UserInfo Endpoint.
5. The UserInfo Endpoint returns Claims about the End-User.

ID Token

iss: issuer.
sub: subject.
aud: audience.
exp: expiration time.
iat: Time at which the JWT was issued.
auth_time: authentication time.
nonce: String value used to associate a Client session with an ID Token.
acr: Authentication Context Class Reference.

Response Types

code	Authorization Code Flow
id_token	Implicit Flow
id_token token	Implicit Flow
code id_token	Hybrid Flow
code token	Hybrid Flow
code id_token token	Hybrid Flow

Display Types

- page
- popup
- touch
- wap

Prompt types

- none
- login
- consent
- select_account

Authorization Code Flow

The Authorization Code Flow goes through the following steps.

Client prepares an Authentication Request containing the desired request parameters.

1. Client sends the request to the Authorization Server.
2. Authorization Server Authenticates the End-User.
3. Authorization Server obtains End-User Consent/Authorization.

Authorization Code Flow (cont)

4. Authorization Server sends the End-User back to the Client with an 5.
5. Authorization Code.
6.Client requests a response using the Authorization Code at the Token Endpoint.
7. Client receives a response that contains an ID Token and Access Token in the response body.
8. Client validates the ID token and retrieves the End-User's Subject Identifier.

Authentication Request

scope: scope values.
response_type: authorization processing flow to be used.
client_id: valid client id.
redirect_uri:** Redirection URI to which the response will be sent.
state: used to maintain state between the request and the callback.
nonce: String value used to associate a Client session with an ID Token.
display: display interface page (page, popup, touch, wap).

Authentication Request (cont)

prompt: reauthentication and consent prompts (none, login, consent, select_acount).

Successful Authentication Response

HTTP/1.1 302 Found Location: https://client.example.org/cb?code=SplxlOBeZQQYbYS6WxSbIA&state=af-0ifjsldkj`

Access Token Request

```
POST /token HTTP/1.1
Host: server.example.com
Content-Type: application/x-www-form-urlencoded
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW
grant_type=authorization_code&code=SplxlOBeZQQYbYS6WxSbIA&redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
```

Access Token Response

access_token: OAuth 2.0 access token.
token_type: token_type value.
refresh_token: token to refresh authorization.
expires_in: expiration time of the Access Token.
id_token: ID Token.

Implicit Flow Overview

The Implicit Flow follows the following steps:
1. Client prepares an Authentication Request containing the desired request parameters.
2. Client sends the request to the Authorization Server.
3. Authorization Server Authenticates the End-User.
4. Authorization Server obtains End-User Consent/Authorization.
5.Authorization Server sends the End-User back to the Client with an ID Token and, if requested, an Access Token.
6. Client validates the ID token and retrieves the End-User's Subject Identifier.

Authentication Request

response_type: value is id_token token or id_token.
redirect_uri: Redirection URI to which the response will be sent.
nonce: String value used to associate a Client session with an ID Token.