AUTHLETE

OAuth 2.0

Authorization Code flow

Recap

OAuth 2.0 enable api clients parties to access protected resources in a domain with limited permissions

Recap - Authorization Server Endpoints

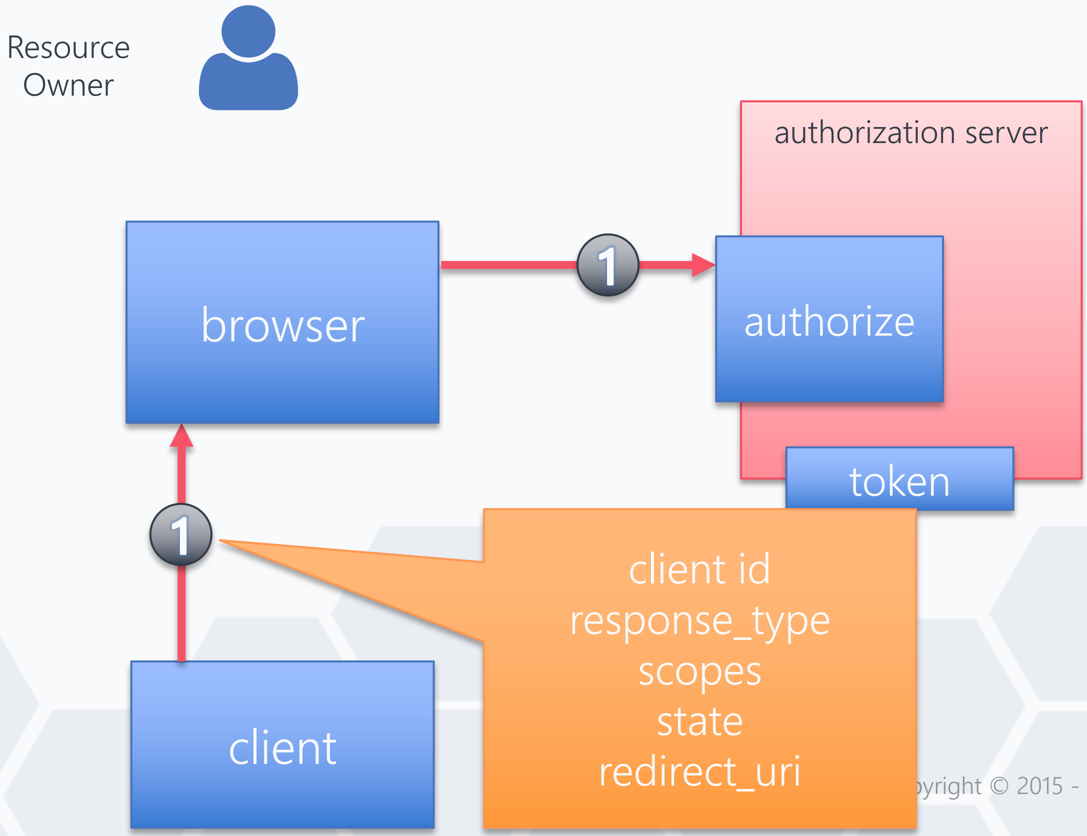
- Authorization
  - Allow the resource owner interact with the authorization server in order to grant permissions to client
- Token
  - Used by the client to retrieve access tokens using grants from authorization or refresh tokens

Recap

Authorization code is one of the grant types that can be used by clients and AS

Authorization code flow

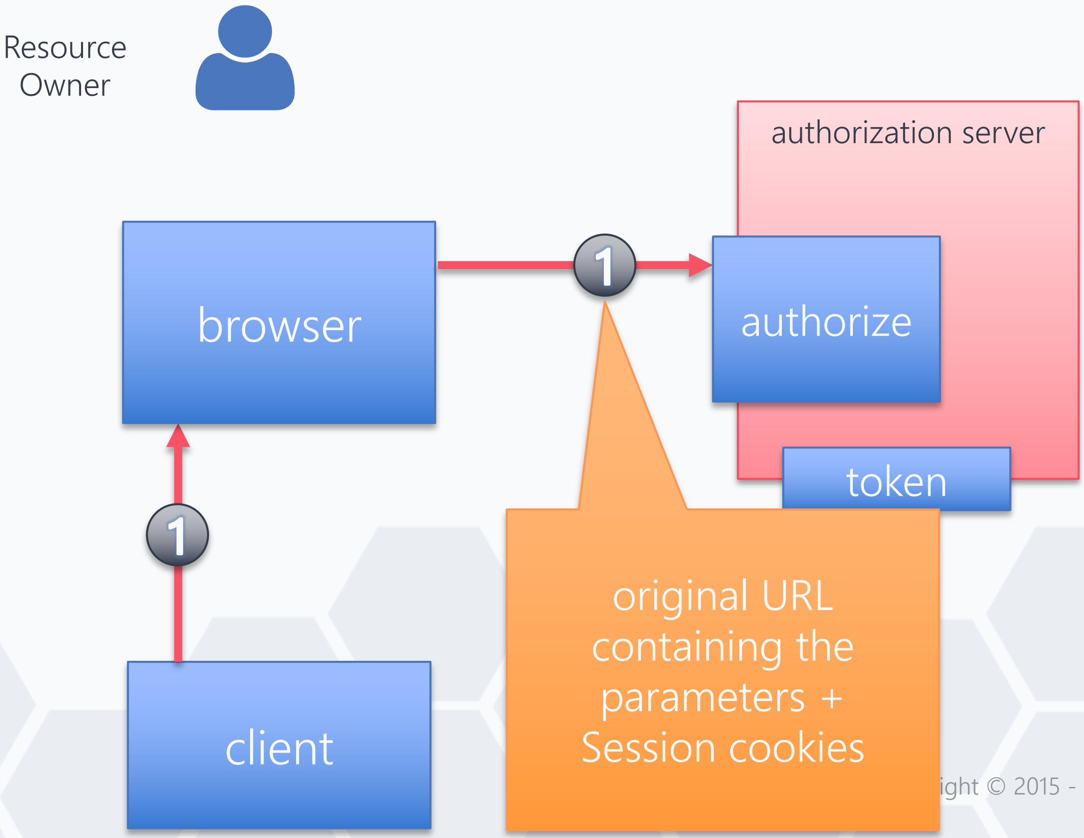
1 – Redirect to authorize endpoint with response_type=code



Client redirect_uri endpoint

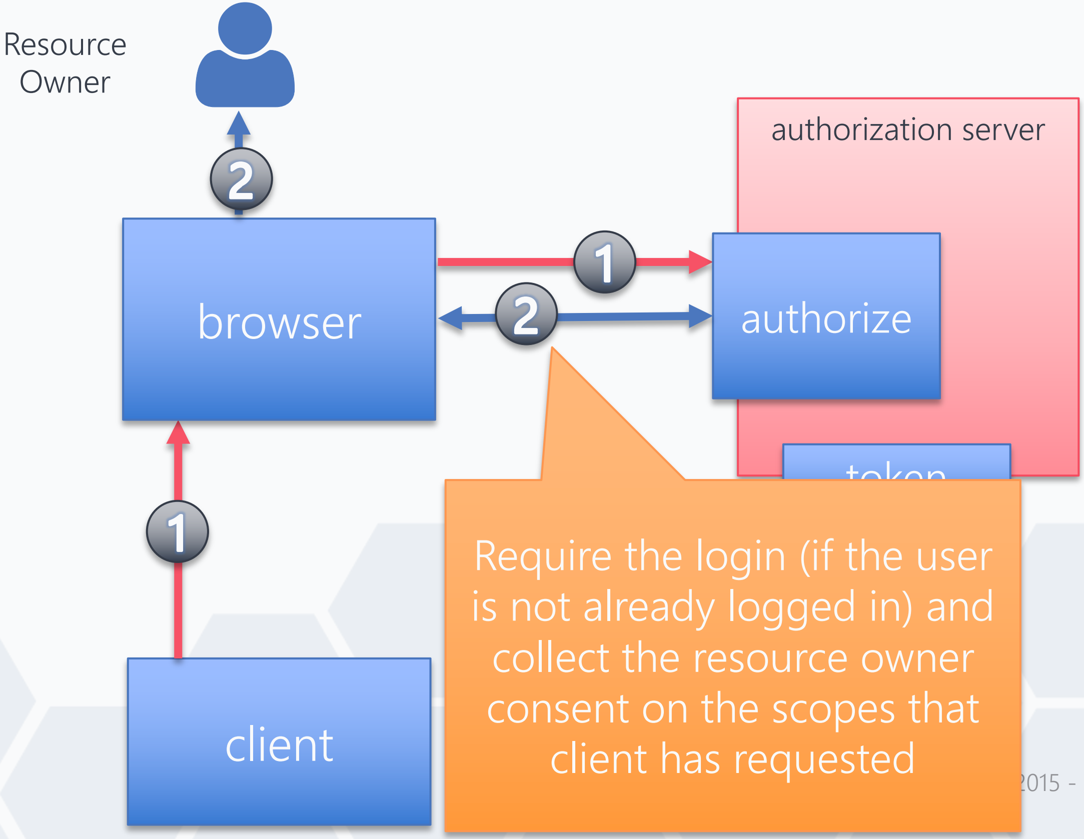
- Client can redirect full page, use iframe, use popups
  - The OAuth specification does not restrict
- The AS response will be a GET request to redirect_uri endpoint of client with the authorization code
- The client might have more than one redirect uri endpoint
  - e.g. one redirect for mobile and another for web flow

Authorization code flow



1 – Redirect to authorize endpoint
with response_type=code

Authorization code flow

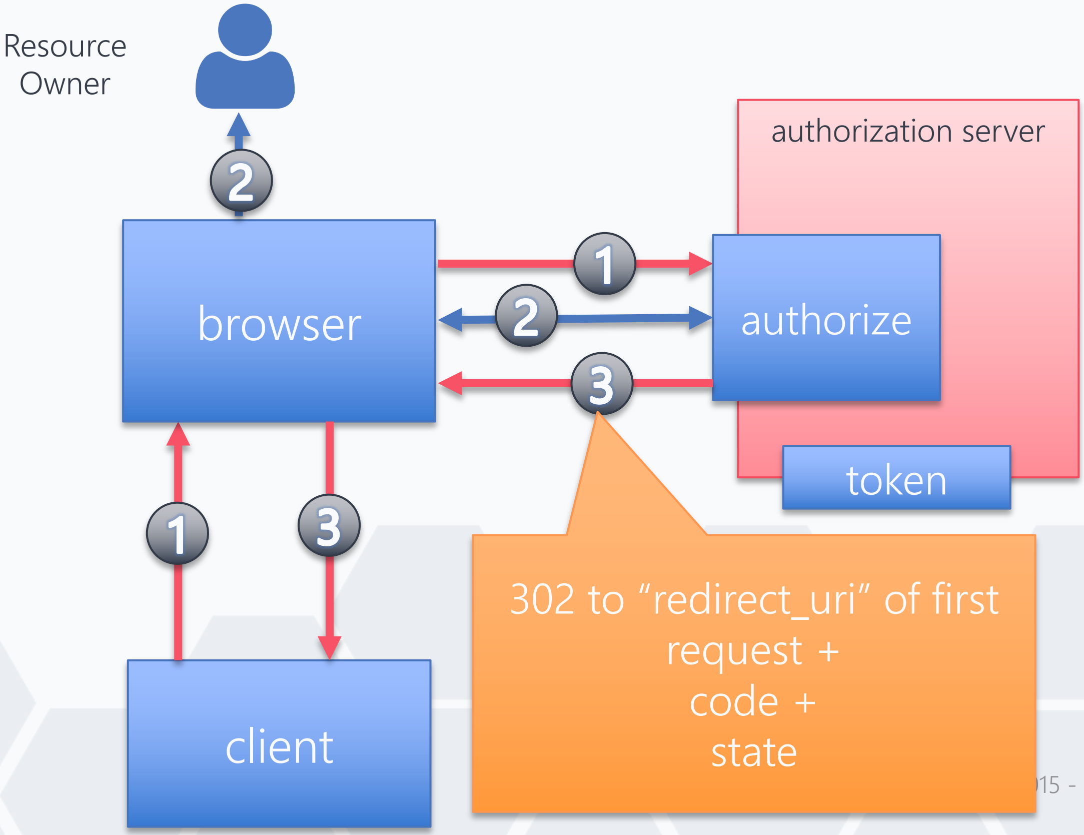


Require the login (if the user is not already logged in) and collect the resource owner consent on the scopes that client has requested

1 – Redirect to authorize endpoint with response_type=code

2 – The user login and grant the permission

Authorization code flow



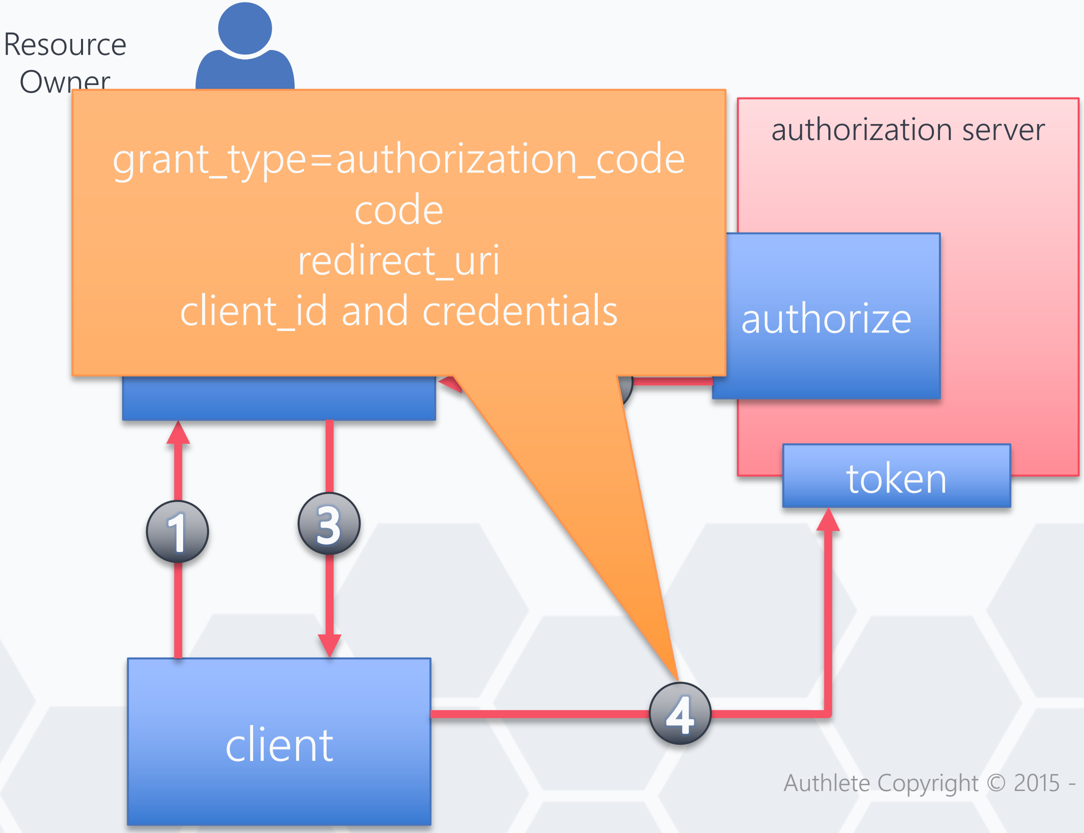
302 to “redirect_uri” of first request +
code +
state

1 – Redirect to authorize endpoint with response_type=code

2 – The user login and grant the permission

3 – redirect to client with the authorization code

Authorization code flow



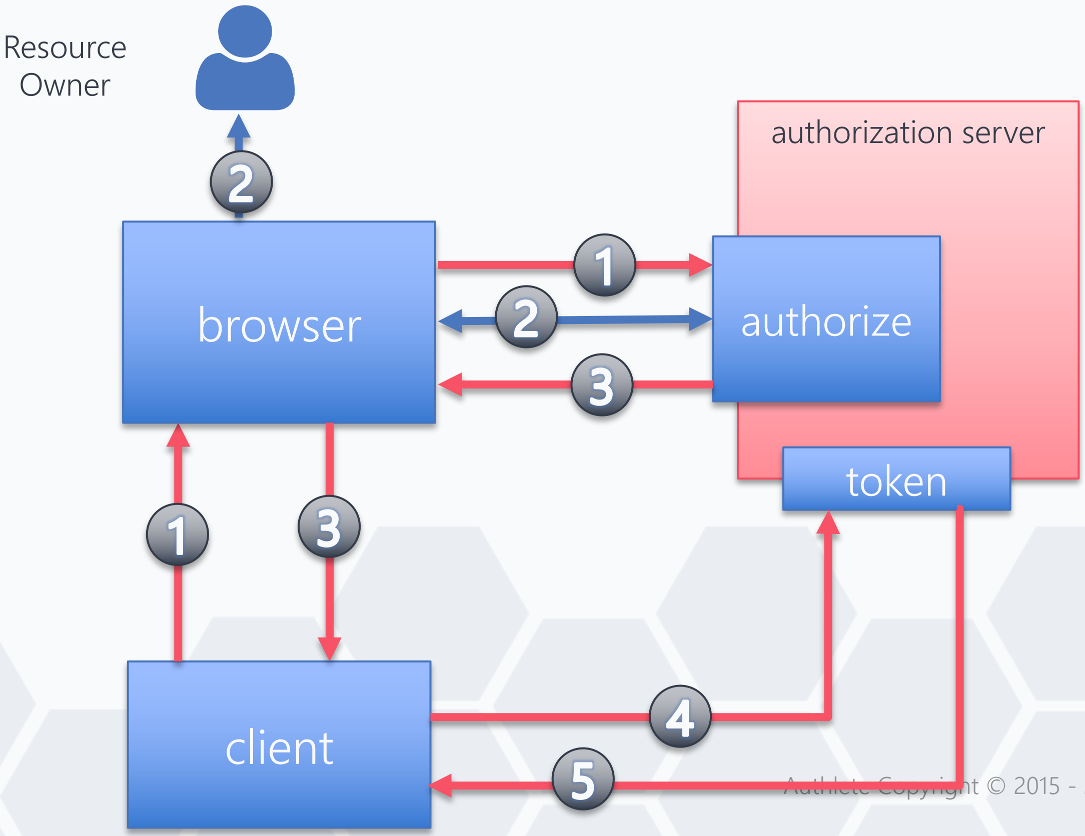
1 – Redirect to authorize endpoint with response_type=code

2 – The user login and grant the permission

3 – redirect to client with the authorization code

4 – client sends the authorization code to token with credentials

Authorization code flow



1 – Redirect to authorize endpoint with response_type=code

2 – The user login and grant the permission

3 – redirect to client with the authorization code

4 – client send the authorization code to token with credentials

5 – AS returns the access token, refresh token, granted scopes and time to live of the token

KEY POINTS

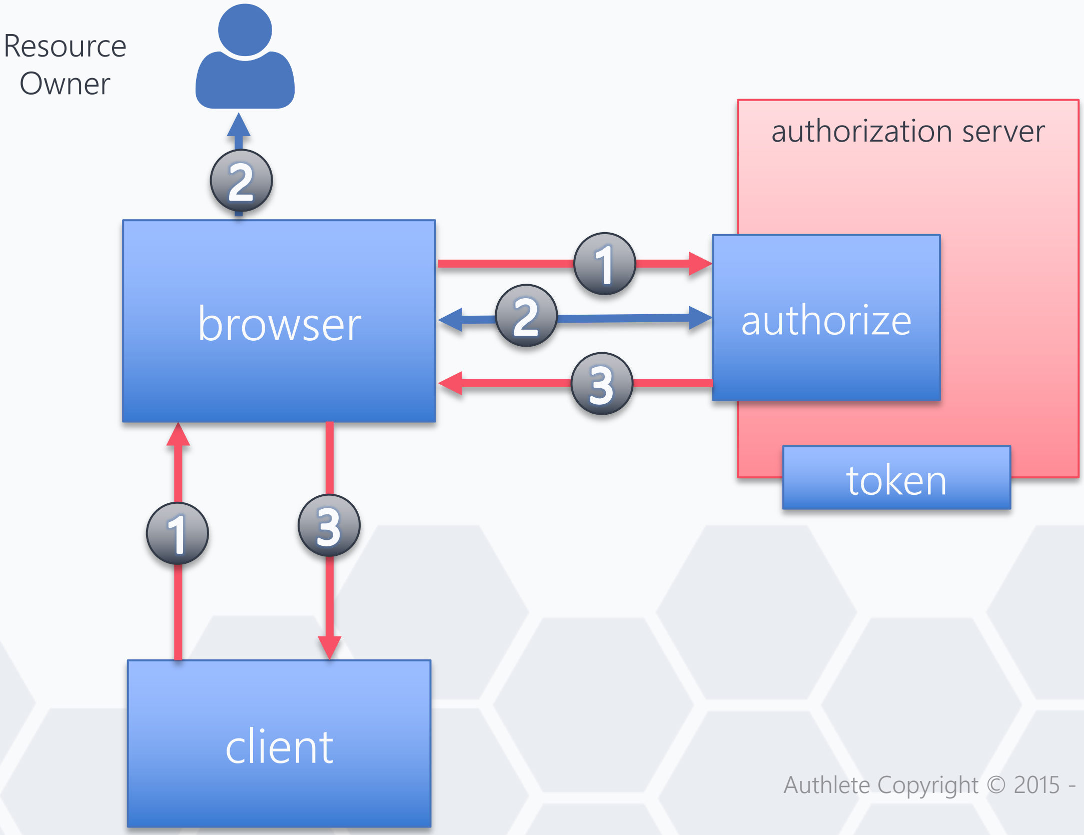
Authorization code

Suitable to every scenario

Authorization code flow

It relies on user’s agent to move the request parameters and authorization code between client and authorization server

Authorization code flow



1 – Redirect to authorize endpoint with response_type=code

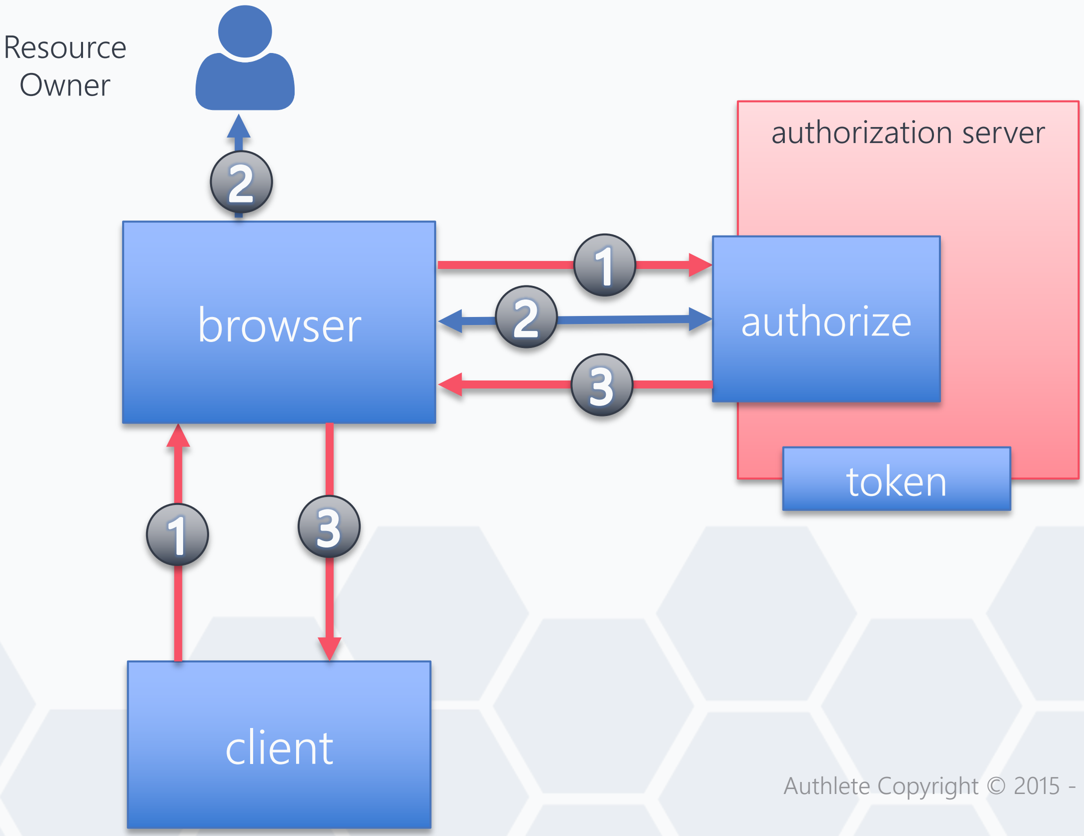
2 – The user login and grant the permission

3 – redirect to client with the authorization code

Authorization code

Clear segregation between authentication and authorization

Authorization code flow



1 – Redirect to authorize endpoint with response_type=code

2 – The user login and grant the permission

3 – redirect to client with the authorization code

Authentication x Authorization

OAuth 2.0 authorization does not preclude what you already have in house for authentication

Authlete approach

Build OAuth and OpenID Connect capabilities on top of existing infrastructure

The Rise of In-house Software Development

“We develop all our technology **in-house** – and that is one of our competitive advantages.”

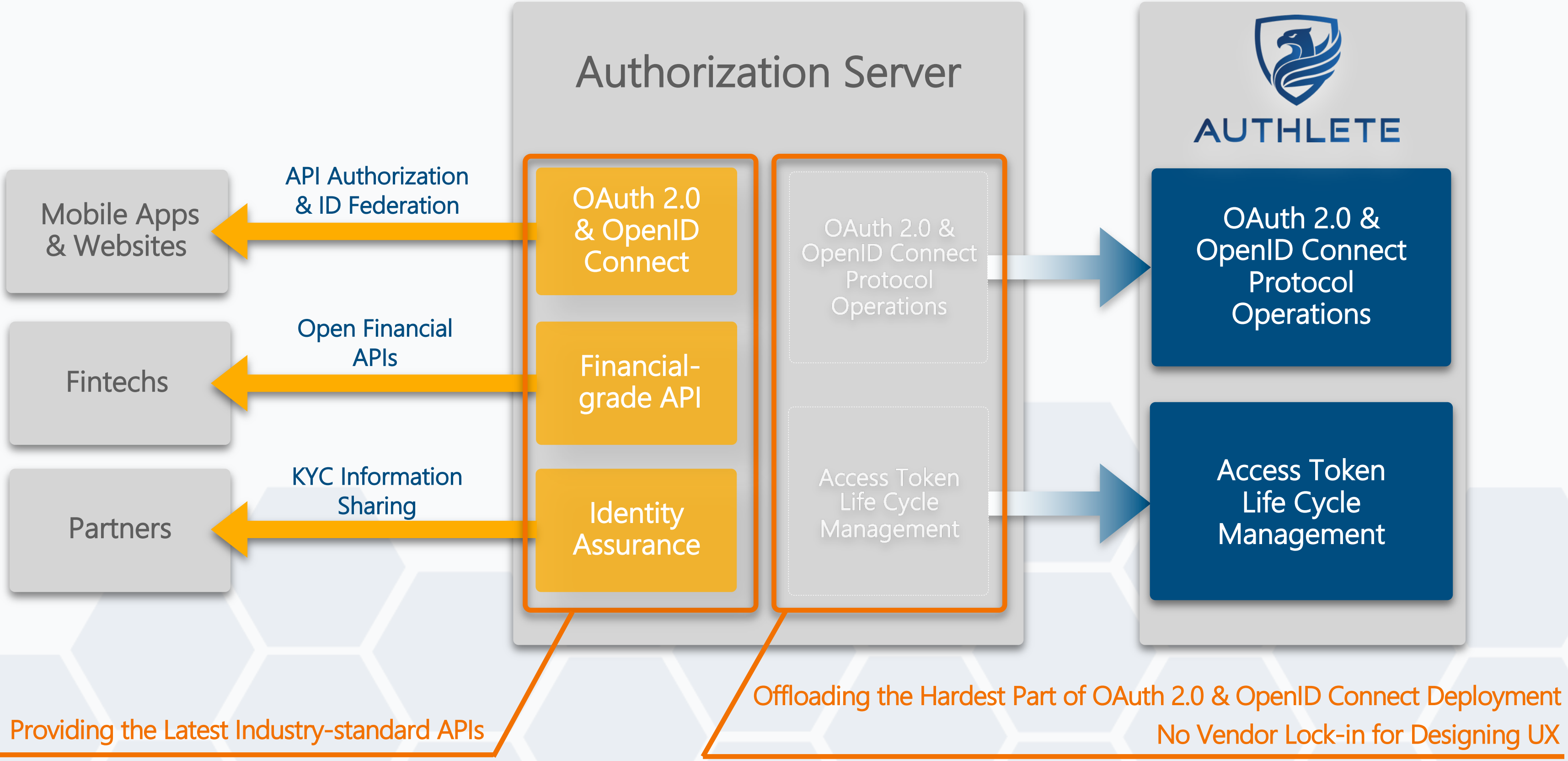
– Nubank, the largest digital bank globally

“I've never heard of DX being left up to vendors. **In-house** software development is a global trend.”

– Minna Bank, cloud native digital bank in Japan

Source: https://building.nubank.com.br/nubank-culture-and-values/, https://bizzine.jp/article/detail/6351?p=3

Authlete: Your Authorization Server Backend



Mobile Apps & Websites

API Authorization & ID Federation

Fintechs

Open Financial APIs

Partners

KYC Information Sharing

Authorization Server

OAuth 2.0 & OpenID Connect

Financial-grade API

Identity Assurance

OAuth 2.0 & OpenID Connect Protocol Operations

Access Token Life Cycle Management

AUTHLETE

OAuth 2.0 & OpenID Connect Protocol Operations

Access Token Life Cycle Management

Providing the Latest Industry-standard APIs

Offloading the Hardest Part of OAuth 2.0 & OpenID Connect Deployment
No Vendor Lock-in for Designing UX

AUTHLETE

www.authlete.com