OpenID + FAPI
Introduction

2021-05-17
OpenID Foundation FAPI WG:
OpenID Foundation


Specialized in the standardization of internet identity layer and API access management.
OpenID Foundation Members

Members spread over 36 countries.
OpenID Standards are used everywhere

Sign in with Apple, Google Sign-in, Microsoft Sign-in, GSMA Mobile Connect etc. are based on OpenID Connect and is estimated to be used by over 3 Billion people.

Number of transactions are also large. As of 2019, over 94% of Microsoft Azure sign-in are performed using OpenID Connect.

So, it was a great starting point on which to build a common highly secure financial data sharing ecosystem.

OpenID FAPI is being used as the API access control standard by UK Open Banking, Australian CDR, New Zealand and others that require higher level API protection.
<table>
<thead>
<tr>
<th></th>
<th>Three Business Lines of OpenID Foundation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Standards Creation - Standardization of the technologies in cooperative area</td>
</tr>
<tr>
<td>2</td>
<td>Testing and Certification Creation of shared testing and self-certification suite for the standards</td>
</tr>
<tr>
<td>3</td>
<td>Marketing and Education Seminars, white-papers and others</td>
</tr>
</tbody>
</table>
Standards Creation

Standardization of the technologies in cooperative area

- Working Groups creates standards and other technical documentations.
- Anybody who signs the IPR Contribution Agreement can join WGs as a contributor. There is no fee associated with it.
- By agreeing to the IPR Contribution Agreement, individuals and corporations agrees to provide their IPR free of charge for the implementations of the Standards and Implementer’s Drafts.
- The standardization process follows “OpenID Process” that adheres to WTO TBT Treaty Annex 3.
There are 10 active working groups (WGs) tackling on the topics that are defined in their charters. (As of 2020-04-27)

WG meets mostly online to share information and develop documents using issue tracking systems and git repositories and mailing lists.
OpenID Connect Core

The Digital Identity Layer for the Internet

- OAuth 2.0 has no notion of identity
- OpenID Connect Provides
  - Who is the user that was authenticated
  - Where were they authenticated
  - When did they authenticate
  - How did they authenticate
  - What attributes have they shared
- OIDC adds additional security requirements
- Introduces the concept of an id_token and userinfo endpoint which enables identity information to be shared in a standardized way.
- Defines a protocol to allow the sharing and confirmation of resource owner (customer) attributes
- Published in 2014
- Enables - Login with [Google, Microsoft, LinkedIn, Apple]
- Enables - Open Banking in the UK, NZ, Australia.
FAPI - Financial-grade API

General Purpose High Security Level API
Protection Protocol based on OAuth 2.0
Family of specifications.
OpenID Connect: Redirect Overview

Use when the Resource Owner (Customer) interacts with the Client (TPP) and Provider (Bank) on the same physical device.

E.g. A mobile application or desktop/laptop browser
OpenID Connect: Client Initiated Back Channel Authentication

Use when the Resource Owner (Customer) interacts with the Client (TPP) and Provider (Bank) on different physical devices.


1. Give Consent + mcdonalds_id + Bank Name
2. Please Authenticate and Authorise + id_token
3. Do you want to authorise?
4. Authorise
5. Authorisation Complete
6. AT/RT/ID Token
7. Refresh
OpenID Connect: Client Initiated Back Channel Authentication

Very Powerful Capability Enables Non Digital Customers to Engage with Open Banking.

1. Give Consent + User Hint + Bank Name
2. Please Authenticate and Authorise + User Hint
3. Do you want to authorise?
4. Authorise
5. Authorisation Complete
6. AT/RT/ID Token
7. Resources
Secure Everything

Designed to address all possible security concerns.

Formal threat model created by Daniel Fett from the University of Stuttgart.

FAPI 1 Baseline Original - Not So Good.

FAPI 1 Advanced Original - Quite Good
RFC6749 - Defines scope as space ‘ ‘ separated list of predefined scopes known ahead of time to the Client and the Authorisation Server.

Completely useless when you have more than one dimension to scoping a resource.
OpenID Connect: Issues with Authorisation

Open Banking Scope

- Requires N dimensions.
  - Length of Access, Range of Access, Type of Access granularity of Data.

E.g I want access to accounts data (not API) between the 2nd of January to the 10th of September for all Credits over $5000 AUD where sender account was 123244:1122334455 and I want this access for 1 week.
OpenID Connect: Issues with Authorisation

Open Banking Scope

- Requires Multi Party Authorisation
  - Joint Accounts for Data Access
  - Multi party consent for Payments

E.g. Give me access to my joint account where the account property setting requires 2 to sign.
OpenID Connect: Issues with Authorisation

Resource Life Cycle ≠ Scope

- Payment Initiation is not the same as Payment Authorisation.
  - Authorisation may take a long time to be given.
  - The service available may not be on offer when the authorisation

E.g Glastonbury or Big Day Out tickets. The merchant isn’t going to hold tickets for the 5 days it requires a corporate account holder to obtain sign off for a payment with 5 to sign for a spend over 100 dollars.
Lodging ‘Intent’ pattern

- Introduces Granularity of Access Ability
- Non-standard way of access resources
- Implement differently in different jurisdictions
  - UK, Brazil and NZ are very very similar.
- Mixes Authorisation Life Cycle with Resource Life Cycle in the case of payments
OpenID Connect: Issues with Authorisation

Rich Authorisation Request

- Introduces Granularity of Access Ability
- Standard way of accessing resources
- Does not introduce cycle management.
Ensure Privacy

Simplifies Implementations for Third Parties - TPPs.

Requires use of Pushed Authorization Requests

Enforces Global Use of Rich Authorisation Requests

1. shall support the `authorization_details` parameter according to [@I-D.ietf-oauth-rar] to convey the authorization clients want to obtain if the `scope` parameter is not expressive enough for that purpose
Specs are nice, but for real interoperability, implementations need to be tested.
Create the future together.