ADVENTURES IN OPENBANKING: UNDERSTANDING OAUTH AND OPENID CONNECT CLIENT ECOSYSTEMS

Pamela Dingle
Director of Identity Standards @ Microsoft
@pamelarosiedee
The work I describe here was accomplished through my previous employer.

This presentation represents my personal experiences and should not be interpreted as representing any corporation.

I am proud to have the opportunity to work for and collaborate with companies that contribute time, money, and expertise to standards efforts.
Standards ❤ Interoperability
Current State:
Financial Institutions offer proprietary banking APIs (or no APIs at all)

Goal State:
Banking APIs regulated by Competent Authorities* across the EU

* Competent Authority: entity designated by a country to supervise & monitor compliance
Financial Institutions MUST
- Make APIs for Payment Initiation & Account Information Sharing
- Allow access to your banking APIs from any ‘trusted third party’ approved by the relevant Competent Authority
- Adhere to the EU “Secure Customer Authentication RTS” (aka SCA)
- Ask for consent! Make it intuitive! No extra barriers! Also, make each thing consented to *unbundled*! You figure it out!

Competent Authorities MUST
- Figure out how this all works, in time for Financial Institutions to comply
PSD2 Roles Decoded*

* Entities can play multiple roles
UK OpenBanking: A Competent Authority

- A jointly funded effort of the 9 largest financial institutions in the UK by parliamentary edict
- “CMA 9” hold > 90% of the UK’s financial accounts
- Mandated to implement Jan 13 2018
- Deadline for challenger banks is September
- Nobody has enough time
- OBUK: https://www.openbanking.org.uk
Open Banking UK World

(Financial Conduct Authority) FCA

(European Banking Authority) EBA

Regulatory Technical Specifications

API Security Specs

Federation Registry Certificate Authority Software Statement Issuer

CMA9

Daily Participation

Federated Trust

(Account Servicing Payments Services Providers)

ASPS, ASPS, ASPS

(Account Information Service Provider)

AISP

(TTP, TPP)

(Trusted Third Parties or Third Party Providers)

ASPS, ASPS

(TTP, TPP)

(TTP, TPP)

(TTP, TPP)

(Payment Service Users) PSU

Relationships
OpenBanking UK Invested in Standards

- OAuth2 and OpenID Connect selected
  - Participants do not have to build bespoke solutions, they can utilize off-the-shelf platforms and services (if they want to)
  - The solution naturally leverages an already mature threat model
  - Vendors can rationalize investment as serving more than just one vertical
  - The solutions are not ingrown, they receive scrutiny from everywhere
Why OpenID Connect?

- OAuth 2.0 standardizes a request for an access token
- OpenID Connect standardizes request and format for identity assertions
  - Format of an assertion, including issuer, destination, encryption/signing, and claims
  - An endpoint where assertion data can be accessed
  - Description of the authentication instant
  - Additional identity-grade security requirements (HTTPS)
- OpenID Connect layered on top of OAuth 2.0 makes OAuth 2.0 both interoperable and certification-capable
Standards World

- FAPI, OIDF, IETF
- API Security Specs
- OBUK
- Federation Registry Certificate Authority Software Statement Issuer
- API Access
- Federated Trust
- IDP/AS
- RP/Client
- Resource Owner
- Consent, MFA
FAPI Profiles

- Financial API WG at the OpenID Foundation
  - Working towards a financial API profile that can work worldwide
  - FAPI read-only and read-write profile tunes OpenID Connect

- OB Profiles (Security & Dynamic Client Registration)
  - Further profile FAPI specifically for OB
  - Initially developed by the OBIE but moved into a standards arena
Banks use Mutually Authenticated TLS. A lot. Everywhere.

Desirable as foundation to OB API Security

Not mature spec-wise (at the time)

Version 00 extension draft (now at v7)

Required layer 3/7 to talk

Group preferred MTLS and client secret

OAuth spec explicitly states there can be only one client authentication method

V7 draft:
https://datatracker.ietf.org/doc/draft-ietf-oauth-mtls/

Photo credit: https://www.flickr.com/photos/drainrat/16171736429/
Problems & Pushes: Communicating Intent

- Requirement to tie incoming requests to a transaction context
- OpenID Connect Request objects allow context to be included in requests
  - Signed request objects
  - Very little vendor support at the time
  - Request objects act as hints
Problems & Pushes: Transitive Trust at Scale

- Financial Institutions are not in charge of deciding who can register a client
- Requirement to support a possibly staggering number of independent clients in the long term
- RFC 7591 - Dynamic Client Registration
  - A standardized endpoint enabling clients to register their metadata with an authorization server in return for a client credential
  - Supports presentation of a ‘software statement’

Photo credit: www.flickr.com/photos/stincodiporco/39447341885
1.3. Protocol Flow

-------------(A)--- Initial Access Token (OPTIONAL)
    +------(B)--- Software Statement (OPTIONAL)
    |     |     |
    ˅     ˅     ˅
   +-----------+
   |           |
   | Client or |
   | Developer |
   |------------+
   |            |
   |             |
   |             |
   | --(C)--- Client Registration Request -->
   | or Client Information Response ---|
---(D)--- Client Error Response ---|
   +-----------------------------+
        Client Registration Endpoint

Figure 1: Abstract Dynamic Client Registration Flow
OAuth 2 & OpenID Connect Dynamic Profiles are like swiping right

Dynamic automation is all about negotiating metadata

1. Supported IDP/AS capabilities are advertised at a discovery endpoint

2. A Client/RP requests a relationship with the AS/IDP by asserting metadata about itself including which of the supported capabilities it will use to engage

3. The IDP/AS responds to the request and tells the client what client_id and other metadata was registered

4. In a success situation, the Client/RP then proceeds to use the registered metadata to successfully request tokens from the IDP/AS
Software Statements constrain the Dating Pool to vetted clients

- A valid assertion from a recognized authority is the pre-requisite to making the request
- Some or all metadata comes from the assertion rather than the client request
- OpenBanking UK software statements describe 1) the authority, 2) the organization, 3) the software
- Organizations must register software and software metadata with the registry
OpenBanking Client Registration Overview (Options A, B)

TPP Primary Technical Contact (PTC)

1: Login
2: Download SSA
3A: Automated Client Registration
6B: Download Client Credentials
4B: SS0 Request
5B: SS0 Response

TPP Client

4A: OAuth Client Registration request w/ SSA
5A: Response with Client Credentials
3B: Manual Client Registration (Login to Portal)

OpenBanking Directory Developer Portal

Option A: Dynamic Client Registration Endpoint
Option B: Developer Web Portal

ASPSAP
Dynamic Client Reg Challenges

- Spec development, vendor development, and developer adoption had to happen simultaneously
  - !!!
- Metadata values specified in RFC 7591 were listed as optional in the software statement
  - OpenBanking treated the software statement as a business assertion, and defined schema analogous to but not the same as RFC 7591.
  - The result is a *lot* of additional rules and mapping for the ASPSP that wants to maintain RFC 7591 compliance
- OpenBanking wanted both the request and the software statement to be assertions. RFC 7591 only specifies the request as a POST.
<table>
<thead>
<tr>
<th>Metadata</th>
<th>Description</th>
<th>OBUK Software Statement Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>redirect_uri</td>
<td>Pre-authorized callback location(s)</td>
<td>software_redirect_uri, redirect_uri</td>
</tr>
<tr>
<td>grant_types</td>
<td>What is the client trading in?</td>
<td></td>
</tr>
<tr>
<td>response_types</td>
<td>Thing returned from the authorization endpoint</td>
<td></td>
</tr>
<tr>
<td>token_endpoint_auth_method</td>
<td>Credential used to retrieve tokens from the token endpoint</td>
<td></td>
</tr>
<tr>
<td>jwks_uri</td>
<td>Client’s public keys</td>
<td>software_jwks_endpoint</td>
</tr>
<tr>
<td>scope</td>
<td>Scopes to be used</td>
<td>software_roles</td>
</tr>
<tr>
<td>software_id</td>
<td>Unique id (by client about software)</td>
<td>software_id</td>
</tr>
<tr>
<td>client_name</td>
<td>Human readable description</td>
<td>software_client_name</td>
</tr>
<tr>
<td>org_id</td>
<td></td>
<td>org_id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>org_jwks_endpoint</td>
</tr>
</tbody>
</table>
OBUK Example Software Statement

{

"iss": "OpenBanking Ltd",

"iat": 1492756331,

"jti": "id12345685439487678",

"software_id": "65d1f27c-4aea-4549-9c21-60e495a7a86f",

"software_client_id": "OpenBanking TPP Client Unique ID",

"software_client_name": "Amazon Prime Movies",

"software_redirect_uris":
[ "https://prime.amazon.com/cb", "https://prime.amazon.co.uk/cb" ],

"software_roles": [ "PISP", "AISP" ],

"org_id": "Amazon TPPID",

"org_name": "OpenBanking TPP Registered Name",

"org_jwks_endpoint": "https://jwks.openbanking.org.uk/org_id/org_id.jwks",

"org_jwks_revoked_endpoint": "https://jwks.openbanking.org.uk/org_id/revoked/org_id.jwks",

"software_jwks_endpoint": "https://jwks.openbanking.org.uk/org_id/software_id.jwks",

"software_jwks_revoked_endpoint": "https://jwks.openbanking.org.uk/org_id/revoked/software_id.jwks",

"software_policy_uri": "https://tpp.com/policy.html",

"software_tos_uri": "https://tpp.com/tos.html",

"software_on_behalf_of_org": "https://api.openbanking.org.uk/scim2/OBTrustedPaymentParty/1234567789"}

So much excellent work

- Strong effort to work with industry
- Strong effort to educate and to be test-driven

https://openbanking.atlassian.net/wiki/spaces/DZ/overview?mode=global

- Amazing work in short timeframe
Where Might this Go?

- End-to-end open banking profile for non-browser interactions
  - Limited input and/or output
  - CIBA/FIDO/???
- Strong proof of possession
  - Based on authenticators
  - Token & Certificate Binding
Next Steps for Standards World

- Need a proper industry profile for dynamic client registration via transitive trust
  - Can Certificate-bound access tokens get rid of the current chained assertions issue (and therefore restore the profile to compliance with 7591)?
  - Can we assert a schema within software statements so that compliance gets easier
  - This should work for any authority
- Conformance should be front and CENTER
Apply UK OpenBanking to YOUR World

- Check your API Strategy
  - Are you passing user credentials on every API fetch?
  - Or are you using a standards-based API security strategy?
- How badly can your clients act?
  - Consider using the OpenID Foundation open source certification tests (even if unofficially)
- What is your client authentication mechanism?
  -Jwt-private and MTLS offer big benefits
- What is your version of SCA?
  - If you aren’t using MFA you are in a world of hurt
  - Start with your admins if you have to
Standards ❤ You
Resources

- OpenBanking UK Developer Zone
  https://openbanking.atlassian.net/wiki/spaces/DZ/overview

- OpenBanking FAPI Profiles (in Bitbucket)
  https://bitbucket.org/openid/obuk

- OpenID Foundation Certification Page
  http://openid.net/certification

- EBA PSD2 Start Page

- Twitter - @pamelarosiedee @openid @UKOpenBanking