OpenID Foundation
Certification Program

Joseph Heenan
Certification Technical Lead
OpenID Foundation
Who Am I?

- OpenID Certification Team lead developer
- Software engineer & architect with over 25 years’ experience
- Active contributor to the OpenID Connect FAPI/CIBA/FAPI-CIBA/eKYC specifications
- 20+ years of mobile app experience
- Assisted 30+ UK banks with achieving compliance to the OpenID/FAPI specifications

https://www.linkedin.com/in/josephheenan/
OpenID Certification Program Overview

- A light-weight, low-cost, certification program to serve members, drive adoption and promote high-quality implementations
  - Identity Providers launched in early 2015
  - Relying Parties launched in late 2016
  - Financial-grade profiles launched in 2019
- Each certification makes it easier for those that follow and helps make subsequent deployments more trustworthy, interoperable and secure
- All certified implementations are freely available at https://openid.net/developers/certified/
- OIDF certification pricing has been widely accepted to date
Certification Program Success

581 certifications of 189 deployments

<table>
<thead>
<tr>
<th>Total OP Certifications</th>
<th>419</th>
<th>Total OP Deployments</th>
<th>120</th>
<th>Total RP Certifications</th>
<th>94</th>
<th>Total RP Deployments</th>
<th>34</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total FAPI Certifications</td>
<td>56</td>
<td>Total FAPI-CIBA Certifications</td>
<td>8</td>
<td>Total FAPI-CIBA Deployments</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total FAPI Deployments</td>
<td>31</td>
<td>Total FAPI RP Certifications</td>
<td>4</td>
<td>Total FAPI Deployments</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(All figures as of January 2021)
Open Banking Adoption of FAPI & FAPI Certification

- UK led the way with FAPI adoption and FAPI certification under the direction of the Open Banking Implementation Entity
  - Currently 15 UK banks have 31 FAPI certifications of 16 deployments
  - Most of the CMA9 have certified
  - OIDF anticipates OBIE requiring CMA9 to recertify annually

- Additional jurisdictions adopting FAPI and FAPI certification
  - US – OIDF anticipates the Financial Data Exchange formally adopting FAPI and requiring FAPI certification
  - AU – OIDF coordinating with AU DSB team who has adopted FAPI as a normative standard and will be encouraging AU banks to FAPI certify
  - Other jurisdictions – OIDF working with regulators and coordinators in Europe, Brazil, Bahrain and other locals to encourage and support the adoption of FAPI and FAPI conformance
Conformance Suite Architecture

- Multi-party protocol testing
- Structured configuration
- Structured logging and results
- Deterministic, modular execution units
  - Small pieces of java code
  - Easily unit testable
- Protect sensitive configuration and results data
- Transparent process
- Usable as part of CI
Architecture - continued

- Loosely bound backend, frontend and test modules
  - Clear interfaces
  - Heavy use of JSON

- Consistent logging of all inputs and outputs

- Easily extensible to new protocols
  - E.g. CIBA added without requiring any changes to frontend/backend

- Does not use existing OAuth2/OpenID Connect libraries
  - Easier to introduce negative tests
  - Easier to show the user exactly what happened and why
Why use the OIDF’s conformance program?

- OIDF tests are developed with close support of relevant working group
  - Tests are updated based on requests from working group

- Testers get direct support from the OIDF certification team
  - Domain experts familiar with all the specs
  - Team have access to OIDF/OAuth2 spec authors when necessary

- Internationally recognized, award winning

- Tests are maintained and updated by OIDF when:
  - new versions of underlying specs published
  - new potential security vulnerabilities are found
  - new interoperability problems are found
  - testers find failures difficult to interpret

- Issues found by testers are raised back to the relevant OIDF working groups
  - Specs can be improved / clarified / disambiguated as necessary
OIDF FAPI Certification Program

- FAPI-RW ID1 OP testing (OBUK specific) started December 2017
- FAPI-RW ID2 OP testing launched April 2019
- FAPI-RW ID2 RP testing launched in June 2019
- FAPI-CIBA ID1 OP testing launched September 2019

Optionally supports:
- OpenBanking UK intent lodging
- Australian Consumer Data Rights for OPs – launched January 2021
- FAPI-RW ID2 OP using PAR (Pushed Authentication Requests – launched January 2021
- App2app authentication/authorization

Visit https://openid.net/certification/instructions/ for details
PAR (Pushed Authentication Requests)

- IETF Standard from OAuth2 Working Group


- An evolution of FAPI-RW’s request object endpoint

- Avoids passing authorization request details via the front channel
  - Better for privacy
  - Better for security (client authenticates before authentication begins)
  - Avoids any size limits on URLs

- Working Group Last Call was August 2020

- Australian CDR planning to go live with PAR from July 2021, wide vendor support

- Certification program for FAPI-RW with PAR launched January 2021
Australian CDR

- Based on FAPI-RW

- 4 or 5 banks (OPs) live, 3 RPs live
  - Many of banks are now going through FAPI conformance testing

- Some extra restrictions compared to base FAPI-RW spec
  - private_key_jwt must be used
  - x-v header must be sent to resource server endpoint
  - Refresh tokens must be supported
  - Returned id_tokens must be encrypted
  - For ACR claims, a CDR specific value is used, “urn:cds.au:cdr:2”

- Development of CDR version of FAPI RP tests under discussion
Final FAPI 1.0 parts 1 and 2 published March 12, 2021

Relatively few normative changes

New names
- FAPI-R -> FAPI Baseline
- FAPI-RW -> FAPI Advanced

Tests for the new version will be added in due course
- Implementers Draft 2 versions of the tests will be retained
FAPI-RW Certification: Core goals

- Interoperability
- Security
- Correct deployment of certified software

However:

- FAPI tests do not test all of OpenID Connect Core or OAuth
  - ‘Pretty good’ coverage of relevant parts though
  - Vendors should run OpenID Connect Core tests as well (if they support non-FAPI)
FAPI-RW Certification: Reasons to Test

- Reduced support costs
  - If your implementation is interoperable it will “just work” for third parties

- Evidence of compliance to show government regulators

- Evidence of compliance may reduce insurance costs, chances of security breach, etc.

- It can be embarrassing if other people test your server & you fail
  - Anyone can test a server
3.3. Authorization Server Metadata Validation

The "issuer" value returned MUST be identical to the authorization server's issuer identifier value into which the well-known URI string was inserted to create the URL used to retrieve the metadata. If these values are not identical, the data contained in the response MUST NOT be used.

<table>
<thead>
<tr>
<th>Time</th>
<th>Status</th>
<th>Check/Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>15:05:56</td>
<td>SUCCESS</td>
<td>CheckDiscEndpointDiscoveryUrl</td>
</tr>
<tr>
<td></td>
<td></td>
<td>discoveryUrl</td>
</tr>
<tr>
<td></td>
<td></td>
<td>actual  <a href="https://fapidev-as.authlete.net/.well-known/openid-configuration">https://fapidev-as.authlete.net/.well-known/openid-configuration</a></td>
</tr>
<tr>
<td>15:05:56</td>
<td>SUCCESS</td>
<td>CheckDiscEndpointIssuer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>issuer is consistent with the discovery endpoint</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OIDCD-4.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OIDCD-7.2</td>
</tr>
</tbody>
</table>
Security Checks - Keys

EnsureServerJwksDoesNotContainPrivateOrSymmetricKeys

private_keys

```json
{
    "p": "uKADG9h1fV0aWcdBAKbIoMwlswTa_3vWMGymWaA0McIfmoYt0_MNQAqos3hKE
u17p2vw3xBo6Dj2zoqptD4646C0nKD3oDawc5yH170m7gePf!YV7n7u8",
    "kt": "RSA",
    "q": "ufhhMtgXTP9u_Upv6i1C7T-YHk_jJ2e3P09Rx74gfkhpO35N6KORV5LzaAC0qJ
x65I7yKylRlB3y4Y4KwW39x57p436g6x0Pmh7yXRF2kP7Gc1EqiujK3uM",
    "d": "Fsd7A9m9kW4M4bvsV0_r_aAXHORr22Aqjw9fR09AaBIY8bJSUXK1CjZbqZQb5-
U5hl1D4nVePzy_LFnPe5xnQLE8B8LiFcaFuzcA8AKPI5N5Hz_rywXiwaS7y1KeIbXr_dyMG
eiNp6_mABXTWmFqvyVw5MT8Ufda-Ew8PKd6yR0cub-1F9hBAnqaqa7FovHIQAe5MuKB
e",
    "e": "AQAB",
    "use": "sig",
    "kid": "sig-2020-07-21T11:27:04Z",
    "qi": "kzvXCY02K9w9kI3833DQJApkXjcx4Phd5J98bAqZzLP3o3smbLWqdvl92acP0-
aw-PxSuRkt6MUFI1CpgeN169L6326kkffm_eATO0r-MNagZembd4r3JgI6k",
    "dp": "lvJMWgHbp3pVfA3DvS9YEv2g1e9zw8ypEh86RttW3r_rKRD6e57zoJhLPEKOHe
zwQ2I1mFdkbRz_9AAJLemFDWk0bhaA0Sng97110t_MXLd3DjFkvOG2G0u",
    "alg": "PS256",
    "dq": "Dm99TP1slEaqX11R3jilQb1onSO-b_RlHOQVe-G6UdrrspRqpowvzRI4FwMy
EwSdztTk5N5VEd4FxMyrbjNakG7k0N8-d00BuXlCbb012hPMTYAHv1ZDLE",
    "n": "hPK_VckSw5jtfaCRPbllNjYrsnpn9m1CCZHvSJIIP8creg10HVS2CjF66Lg
YzesHvTrJ-dDrgfAGW8_U-g0zW_7MQH4Zkhw_RllLGP814hhWml-xzEH5-Syrv8H_L80hX2
rwF898BknLeeDIPdaxHxzY0khaP7cc03W7EzkUud9y64TEMxG_YeEMDCbDr-mayCrHy54AgZk
}
```

symmetric_keys

[]
Security Checks – JWS Request Objects Processing

Signed the request object

08:45:05  SUCCESS

More ...

More ...

Built requests object by value to redirect to authorization endpoint

08:45:05  SUCCESS

More ...

More ...

Fapi-rw-id-2 ensures different nonce inside and outside request object

Redirecting to authorization endpoint

More ...
Security Checks – objects ‘signed’ with alg ‘none’
Further Security Checks – Request Object

- ‘exp’ already expired
- Incorrect ‘aud’
- Correctly signed, but with non-permitted alg
- With a syntactically valid, but incorrect, signature
- Valid signature but from a different client
- With nonce only outside request object
- With non-registered redirect uri
Security Checks – Token Endpoint

- Calling token endpoint
  - Without client authentication
  - With expired client authentication assertion
  - With client authentication assertion intended for different server (‘aud’)
  - Valid client authentication, but passing client_id for target client
  - With already-used authorization code
  - With authorization code issued to another client
  - No MTLS client cert supplied for binding access token to
Security Checks – continued

- **JWKS**
  - Keys too short

- **Authorization code**
  - Too short
  - Not enough entropy

- **Calling resource server**
  - With valid mTLS client cert, but not the one bound to access token

- **TLS 1.0/1.1 not allowed**

- **Insecure ciphers not allowed**

- **And many more…**
Interoperability Checks – Time Stamps

“Seconds since 1st Jan 1970” has been a well-known standard for years... but:

![Image showing a failed interoperability check with a timestamp discrepancy]

- updated_at: May 31, 52521, 1:30:00 AM
- now: Jul 21, 2020, 8:37:08 AM
Interoperability checks - continued

- The standard ‘happy’ flow
- Variants on Accept: headers
  - With/without charset
  - With q parameters
  - With multiple options
- With optional fields
  - All present
  - All missing
- Where case insensitive, testing both cases
- With allowed variants
  - ‘aud’ is an array
- Discovery document
  - Reflects what’s supported
  - Syntactically valid
FAPI Certification: First, FAPI compliance

- First, become FAPI compliant

- Ideally upgrade to a FAPI certified version of your vendor’s product

- Software that is not FAPI certified is likely to be missing:
  - Important configuration controls
  - “Recent” required standards like MTLS sender constrained access tokens
  - Well established but higher security OAuth2 options
    - e.g. client authentication using replay-proof asymmetric cryptography
  - Tamper proof (JWT Secured) OAuth2 authorization requests

- Check any HSMs (Hardware Security Modules) in use
  - Older ones may only support RSASSA-PKCS1-v1_5, which has known weaknesses
FAPI Certification: Pre-testing steps

- Two registered OAuth2 clients are required

- Tester needs to be able to create & register client credentials
  - Or be provided with them in the correct format

- Recommended that tester has existing domain knowledge
  - TLS certificates, JWKS manipulation, OAuth2, FAPI
  - For first run, a developer or highly-technical tester is desirable
Conformance Suite source code etc publicly available on gitlab:  
https://gitlab.com/openid/conformance-suite

Instructions for testing/certifying:  
https://openid.net/certification/instructions/

Production deployment:  
https://www.certification.openid.net/  
(Login with any google/gitlab/openid account)

Contact me if you’d like some help:  
o joseph.heenan@oidf.org or certification@oidf.org  
o https://twitter.com/josephheenan  
o https://www.linkedin.com/in/josephheenan