AUTHLETE

2022-01-12 Financial-grade API (FAPI) 勉強会

FAPI 2.0の動向とAuthlete

工藤達雄

Authlete, Inc.

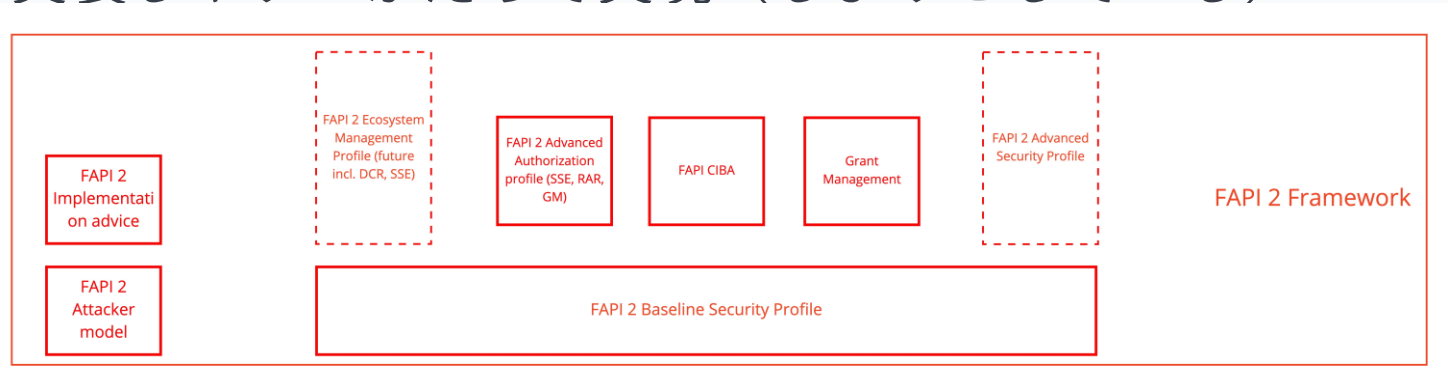
はじめに (重要!)

- "In short, the FAPI working group **recommends** that **deployments** being specified at present use **FAPI 1.0 Advanced**, since it is Final and has been formally verified. At the same time, we also **encourage** people interested in **FAPI 2.0 development** to **participate in the working group**."

Source: What are the differences between FAPI 1.0 and FAPI 2.0 and what do they mean to you?
https://openid.net/2021/12/14/what-are-the-differences-between-fapi-1-0-and-fapi-2-0-and-what-do-they-mean-to-you/

FAPI 2.0 とは

- FAPI 1.0 をベースにした、よりシンプル・エレガント・多機能・相互運用可能な仕様スイート（を目指している）
- “**FAPI 2.0 Baseline**”: ”FAPI 1.0 Advanced” と同等のセキュリティを実装しやすいかたちで実現（しようとしている）

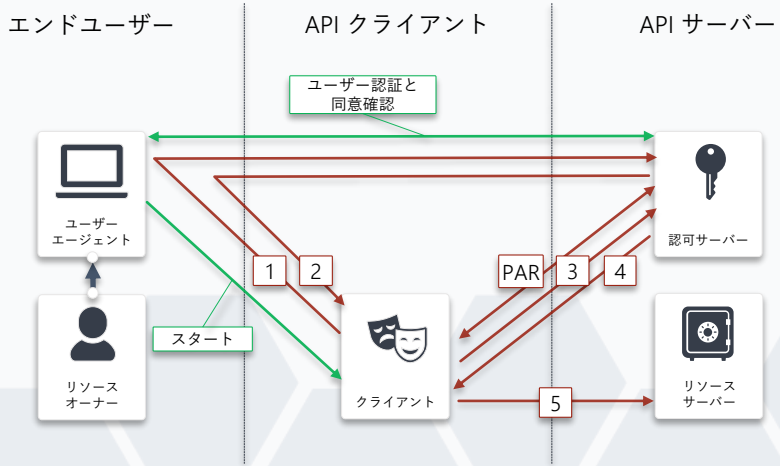


Source: openid / fapi / issues / #432 - FAPI2 Trust Framework structure — Bitbucket https://bitbucket.org/openid/fapi/issues/432/fapi2-trust-framework-structure

"FAPI 2.0 Baseline"

"FAPI 1.0 Advanced" からの主な変更点

	1.0 Advanced	2.0 Baseline
1. 認可リクエスト	Request Object or PAR + PKCE	**PAR + PKCE**
2. 認可レスポンス	"Hybrid" or JARM	**iss**
3./4. トークンリクエスト・レスポンス (送信者限定アクセストークンの発行)	mTLS	mTLS or **DPoP**
5. APIリクエスト (送信者限定アクセストークンの利用)	mTLS	mTLS or **DPoP**

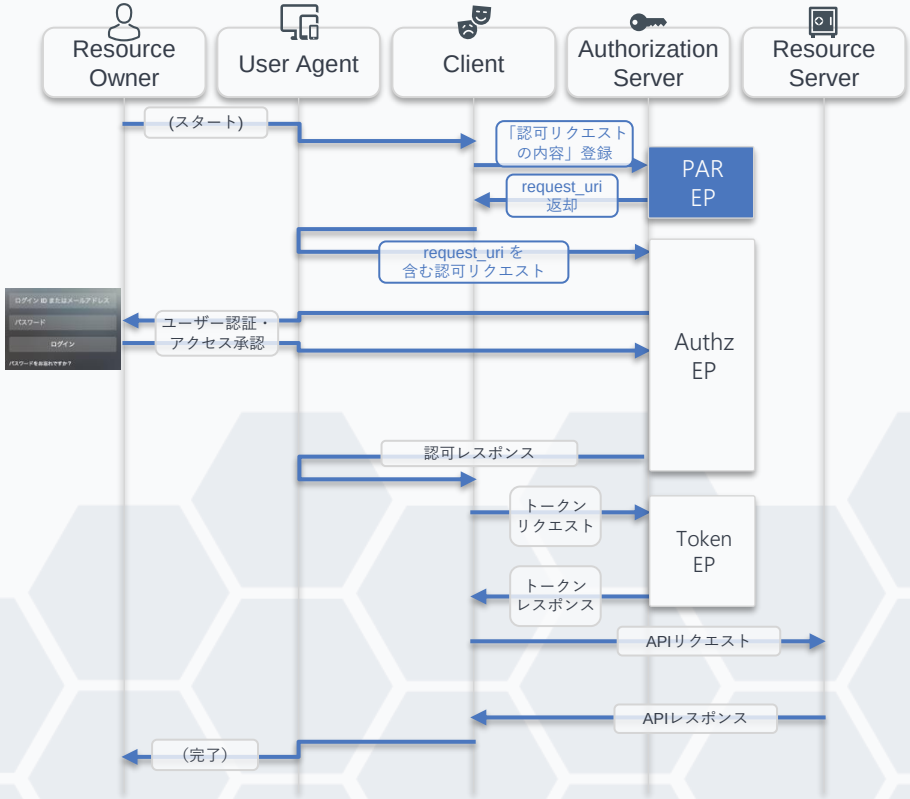


- **Request Object**: OpenID Connect Core 1.0 incorporating errata set 1 https://openid.net/specs/openid-connect-core-1_0.html#JWTRequests
- **PAR**: RFC 9126: OAuth 2.0 Pushed Authorization Requests https://www.rfc-editor.org/rfc/rfc9126.html
- **PKCE**: RFC 7636: Proof Key for Code Exchange by OAuth Public Clients https://datatracker.ietf.org/doc/html/rfc7636
- **"Hybrid"**: OpenID Connect Core 1.0 incorporating errata set 1 https://openid.net/specs/openid-connect-core-1_0.html#HybridFlowAuth
- **JARM**: Financial-grade API: JWT Secured Authorization Response Mode for OAuth 2.0 (JARM) https://bitbucket.org/openid/fapi/src/master/Financial_API_JWT_Secured_Authorization_Response_Mode.md
- **iss**: OAuth 2.0 Authorization Server Issuer Identification https://datatracker.ietf.org/doc/html/draft-ietf-oauth-iss-auth-resp
- **mTLS**: RFC 8705: OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens https://datatracker.ietf.org/doc/html/rfc8705#section-3
- **DPoP**: OAuth 2.0 Demonstrating Proof-of-Possession at the Application Layer https://datatracker.ietf.org/doc/html/draft-ietf-oauth-dpop

"FAPI 2.0 Baseline"

PARの利用が必須

- PARを用いた認可リクエスト
  - クライアントが認可サーバーのPAR EP (エンドポイント) に「認可リクエストの内容」を登録
  - PAR EP がrequest_uriを返却
  - クライアントはrequest_uriを認可リクエストに使用



"FAPI 2.0 Baseline"

「PARを用いた認可リクエスト」に関する主な規定

- PAR EPでのクライアント認証が必須
  - mTLS or private_key_jwt
- 許容される認可グラントは code のみ
- PKCE必須 & S256限定
- redirect_uriの指定必須

```
POST /as/par HTTP/1.1 Host: as.example.com
Content-Type: application/x-www-form-urlencoded

client_assertion_type=
 urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-bearer&
client_assertion=eyJraWQiOiI0MiIsImFsZyI6IkVTMjU2In0.eyJpc3MiOiJDTE
 lFTlQxMjM0Iiwic3ViIjoiQ0xJRU5UMTIzNCIsImF1ZCI6Imh0dHBzOi8vc2VydmVyL
 mV4YW1wbGUuY29tIiwiZXhwIjoxNjI1ODY4ODc4fQ.Igw8QrpAWRNPDGoWGRmJumLBM
 wbLjeIYwqWUu-ywgvvufl_0sQJftNs3bzjIrP0BV9rRG-3eI1Ksh0kQ1CwvzA&
response_type=code&
code_challenge=E9Melhoa2OwvFrEMTJguCHaoeK1t8URWbuGJSstw-cM&
code_challenge_method=S256&
redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb%2Fexample.com&
client_id=CLIENT1234&
scope=payment
```

PAR EPへのリクエスト例

"FAPI 2.0 Baseline"

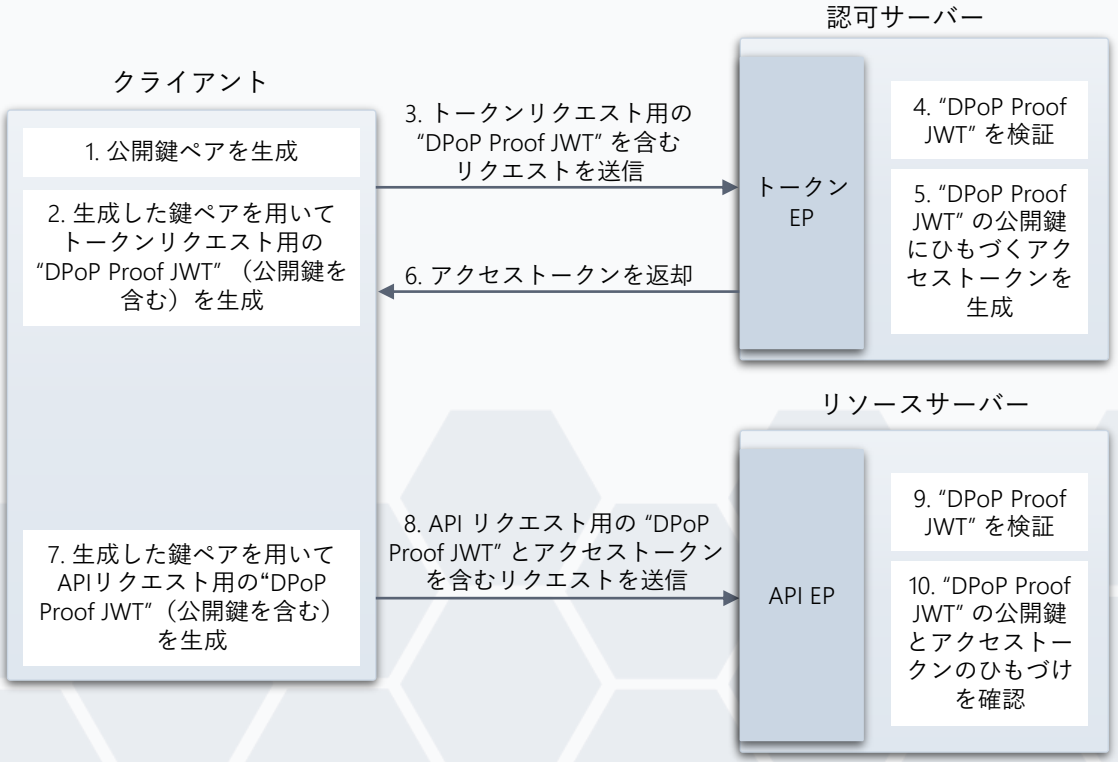
署名関連処理の簡略化

- クライアント側での「認可リクエストの署名」と「認可レスポンスの署名検証」が不要に
- メッセージの否認防止は "FAPI 2.0 Advanced" に規定
  - 認可リクエスト: PAR + Request Object
  - 認可レスポンス: JARM
  - (イントロスペクション: JWT Response for OAuth Token Introspection)
    https://datatracker.ietf.org/doc/html/draft-ietf-oauth-jwt-introspection-response

"FAPI 2.0 Baseline"

DPoP: 送信者限定アクセストークンの新たな選択肢

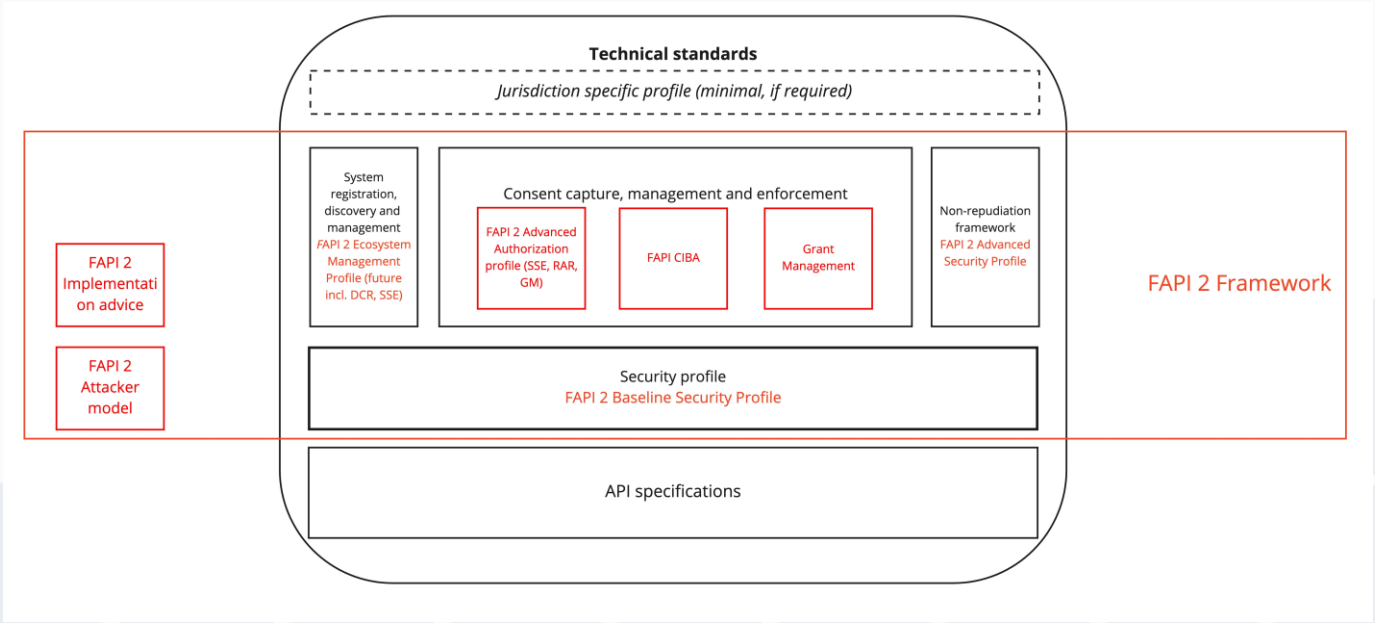
- アプリケーション層でPoP*トークンを実現するしくみ
- mTLS適用が難しい分野をカバー



* Proof of Possession

Beyond “FAPI 1”

- 「細かな粒度の認可」と「グラント管理」が焦点



Source: openid / fapi / issues / #432 - FAPI2 Trust Framework structure — Bitbucket https://bitbucket.org/openid/fapi/issues/432/fapi2-trust-framework-structure

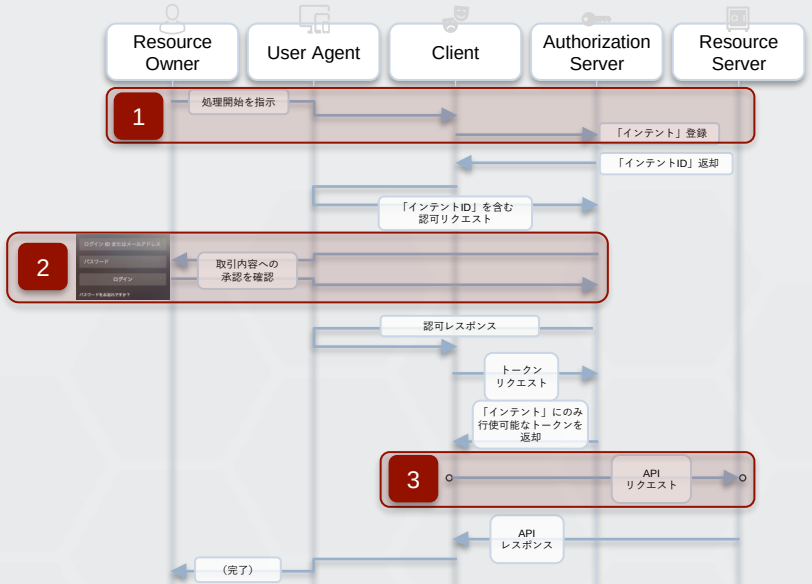
Beyond “FAPI 1”

細かな粒度の認可

- scopeよりも複雑な「認可詳細」を扱うために、FAPIを採用した各エコシステムはそれぞれ独自に対応していた

例: Lodging Intent Pattern

1. サードパーティが銀行の API に口座情報取得や決済指図伝達などの「インテント」を事前登録
2. 登録された内容を、利用者が銀行のWebサイトやアプリにて確認・承認
3. 承認後、サードパーティが「インテント」の内容の実行を銀行のAPIにリクエスト



cf. https://www.authlete.com/ja/resources/templates/sequence-diagrams/

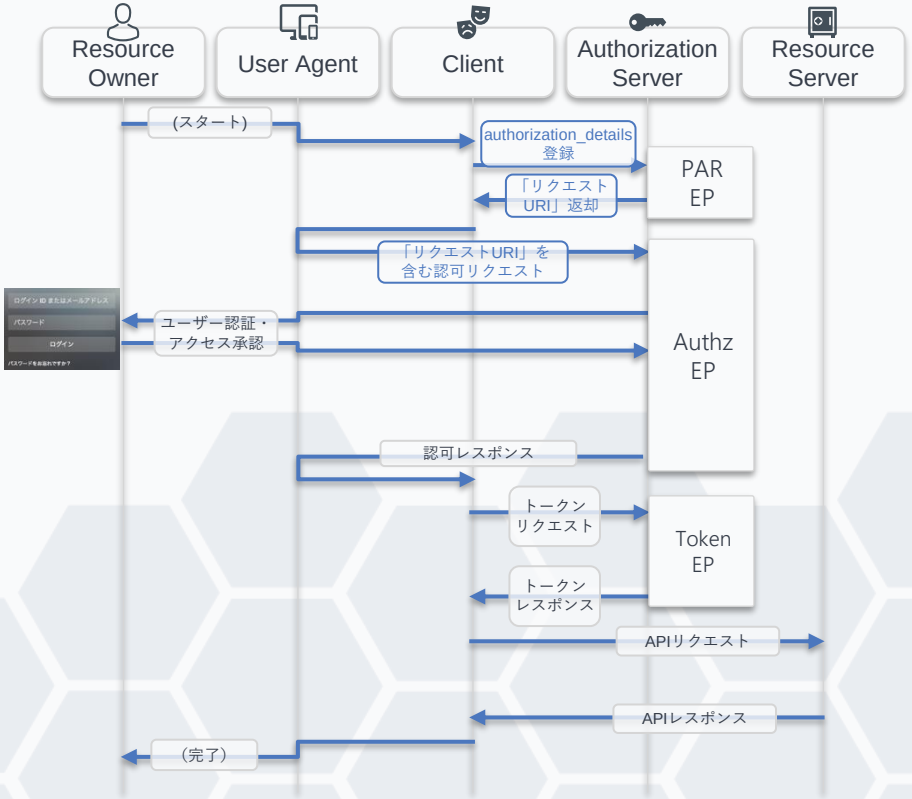
Beyond “FAPI 1”

細かな粒度の認可 (cont.)

- RAR (authorization_details) による「認可詳細」の標準化

```
[
 {
  "type": "account_information",
  "actions": [ "list_accounts", "read_balances", "read_transactions" ],
  "locations": [ "https://example.com/accounts" ]
 },
 {
  "type": "payment_initiation",
  "actions": [ "initiate", "status", "cancel" ],
  "locations": [ "https://example.com/payments" ],
  "instructedAmount": { "currency": "EUR", "amount”: "123.50" },
  "creditorName": "Merchant A",
  "creditorAccount": { "iban": "DE02100100109307118603" },
  "remittanceInformationUnstructured": "Ref Number Merchant”
 }
]
```

authorization_detailsの例

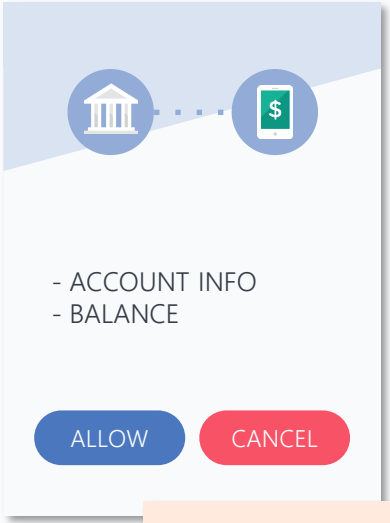


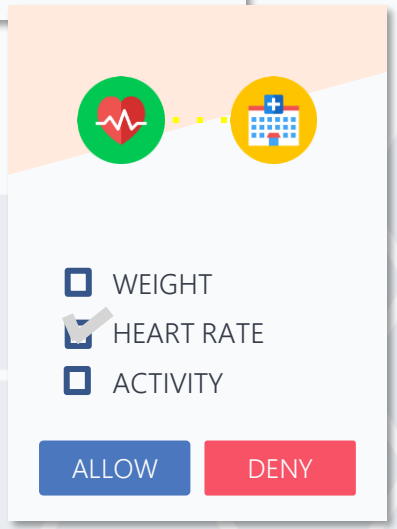
Source: OAuth 2.0 Rich Authorization Requests https://datatracker.ietf.org/doc/html/draft-ietf-oauth-rar#section-2

Beyond “FAPI 1”

グラント管理

- ユーザーはクライアントへ権限 (grant) を与えることに同意 (consent) する
- クライアントは権限 (grant) に関し、
  - 既存のgrantを照会したい・取り消したい
  - consentの範囲内で既存のgrantを更新（e.g., 追加の権限を要求）したい
  - 同一ユーザーに、client_idは共通だが複数に分かれるサービスごとにgrantを扱いたい





Beyond “FAPI 1”

グラント管理 (cont.)

- Grant Management for OAuth 2.0
  - grantのライフサイクル管理の拡張仕様
  - grant ごとに一意な grant_id を採番
  - 生成・更新・置換: 認可リクエストのパラメーターとして grant_management_action を追加
  - 照会・失効: Grant Management API を新設

Beyond “FAPI 1”

グラント管理 (cont.)

- 生成・更新・置換
  - 認可リクエストを利用 (i.e., リソースオーナーが関与)
  - grant_idを含むトークンレスポンスが返却される

```
GET /authorize?response_type=code
 &client_id=s6BhdRkqt3
 &grant_management_action=update
 &grant_id=TSdqirmAxDa0_-DB_1bASQ
 &scope=write
 &redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
 &code_challenge_method=S256
 &code_challenge=K2-ltc83acc4h... HTTP/1.1
Host: as.example.com
```

認可リクエスト (grant_management_action=update)

```
HTTP/1.1 200 OK
Content-Type: application/json
Cache-Control: no-cache, no-store

{
   "access_token": "2YotnFZFEjr1zCsicMWpAA",
   "token_type": "example",
   "expires_in": 3600,
   "refresh_token": "tGzv3JOkF0XG5Qx2TlKWIA",
   "grant_id":"TSdqirmAxDa0_-DB_1bASQ"
}
```

トークンレスポンス

Source: Grant Management for OAuth 2.0
https://openid.net/specs/fapi-grant-management-01.html#name-authorization-request

Beyond “FAPI 1”

グラント管理 (cont.)

- 照会・失効
  - Grant Management API を利用
  - GET / DELETE /grants/<grant_id>

```
GET /grants/TSdqirmAxDa0_-DB_1bASQ
Host: as.example.com
Authorization: Bearer 2YotnFZFEjr1zCsicMWpAA
```

照会リクエスト (GET)

```
HTTP/1.1 200 OK
Cache-Control: no-cache, no-store
Content-Type: application/json

{
  "scopes": [
    {
      "scope": "contacts read write",
      "resources": [ "https://rs.example.com/api" ]
    },
    {
      "scope": "openid"
    }
  ],
  "claims": [ "given_name", "nickname", "email", "email_verified” ],
  "authorization_details": [
    {
      "type": "account_information",
      "actions": [ "list_accounts", "read_balances", "read_transactions” ],
      "locations": [ "https://example.com/accounts" ]
    }
  ]
}
```

レスポンス

Source: Grant Management for OAuth 2.0 https://openid.net/specs/fapi-grant-management-01.html#section-6.4-3

FAPI 2.0 仕様の動向

- 実装者向けドラフト
  - FAPI 2.0 Baseline Profile
    https://openid.net/specs/fapi-2_0-baseline-ID1.html
  - FAPI 2.0 Attacker Model
    https://openid.net/specs/fapi-2_0-attacker-model-ID1.html
  - Grant Management for OAuth 2.0
    https://openid.net/specs/fapi-grant-management-ID1.html
- その他準備中
  - openid / fapi — Bitbucket
    https://bitbucket.org/openid/fapi/src/master/

Authlete の FAPI 2.0 対応

- “FAPI 1.0 Advanced” 認定取得済
- “FAPI 2.0 Baseline” を構成する拡張仕様を実装済
- Grant Management を実装済

	1.0 Advanced	2.0 Baseline
1. 認可リクエスト	Request Object or PAR + PKCE	**PAR + PKCE**
2. 認可レスポンス	“Hybrid” or JARM	**iss**
3./4. トークンリクエスト・レスポンス (送信者限定アクセストークンの発行)	mTLS	mTLS or **DPoP**
5. APIリクエスト (送信者限定アクセストークンの利用)	mTLS	mTLS or **DPoP**

まとめ

- “FAPI 2.0” は “FAPI 1.0” の後継となりうる仕様
  - シンプル・エレガント・多機能・相互運用可能
- 現時点での実運用への適用は要注意
  - 「実装者向けドラフト第1版」もしくは「準備中」
- Authlete は “FAPI 2.0” を構成する仕様を実装済
  - ”FAPI 1.0” を PAR, RAR, DPoP, Grant Management で強化

AUTHLETE

Thank You

Tatsuo Kudo
www.linkedin.com/in/tatsuokudo