[2018]

ข้อกำหนดทางเทคนิค (Technical Specification)

ในส่วนของการเชื่อมต่อด้วย WEB API สำหรับ IDENTITY PROVIDER

CONFIGURATION FOR IDENTIY PROVIDER

สำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์ (องค์การมหาชน) ELECTRONIC TRANSACTIONS DEVELOPMENT AGENCY (PUBLIC ORGANIZATION)

วัตถุประสงค์ (**Objective**)	เอกสารฉบับจัดทำขึ้นมาเพื่อเป็นคู่มือที่ใช้สำหรับการ Configuration เพื่อเชื่อมต่อกับ Identity Provider
เจ้าของโครงการ (**Ownership**)	สำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์ (องค์การมหาชน)
วันที่เริ่มโครงการ (**Start Date**)	1 มกราคม 2561
วันที่สิ้นสุดโครงการ (**Completion Date**)	30 มิถุนายน 2561
ประกาศโครงการ (**Important Notice**)	

ข้อมูลเอกสารและการอนุมัติเอกสาร (DOCUMENT INFORMATION AND APPROVALS)

วัตถุประสงค์ (Objective)	เพื่อเป็นแนวทางของ IdP ในการเตรียมการเชื่อมต่อกับระบบ Federation Proxy สำหรับการยืนยันตัวตนเพื่อใช้บริการภาครัฐและบริการอื่นที่ต้องการเชื่อมต่อกับภาครัฐ

การปรับปรุงเอกสาร (DOCUMENT VERSION HISTORY)			
เลขที่ รุ่น (Version No.)	**วันที่ (Date)**	**ผู้ปรับปรุง (Revised By)**	**เหตุผลการเปลี่ยนแปลง (Reason for Change)**
1.0	1 มี.ค. 2561	ETDA	สร้างเอกสาร
1.0	10 เม.ย. 2561	ETDA	ปรับปรุง Parameter และรายละเอียดเพิ่มเติมเกี่ยวกับ ID Token
1.0	25 พ.ค. 2561	ETDA	- เปลี่ยน Error code ของ invalid_request_uri จาก 302 เป็น 400

สารบัญ

1. ความต้องการของ IDP (IDP REQUIREMENT)

ความต้องการของ IdP จะต้องมีระบบที่รองรับ OpenID Connect 1.0 ประเภท Authentication Code โดยใช้ Protocol OpenID Connect 1.0 และมี Field ของ ID token ตามข้อกำหนดต่อไปนี้ openid, profile, profile_kyc และได้รับการตรวจสอบด้านมาตรฐานการลงทะเบียน Identity Assurance Level (IAL) และการยืนยันตัวตน Authentication Assurance Level (AAL) จากสำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์ (องค์การมหาชน) (สพธอ.) โดยผู้ที่จะยื่นคำขอเป็นผู้ให้บริการอัตลักษณ์นั้นจะต้องมีระบบรองรับการทำงานของ OpenID Connect 1.0 เพื่อเชื่อมต่อกับ Federation Proxy

1.1 ขั้นตอนก่อนการเชื่อต่อกับ Federation Proxy

IdP ต้องทำการออก Client ID และ Client Secret ให้กับ Federation Proxy เพื่อใช้ในการเชื่อมต่อด้วย OpenID Connect 1.0 โดย Federation Proxy จะกำหนด Redirect URL ที่ใช้ในการรับ Authentication Code ให้กับ IdP

1.1.1 ข้อมูลที่ Federation Proxy จะต้องส่งให้กับ IdP

พารามิเตอร์	คำอธิบาย
Redirect URL	URL ที่ใช้ในการรับ Authentication Code

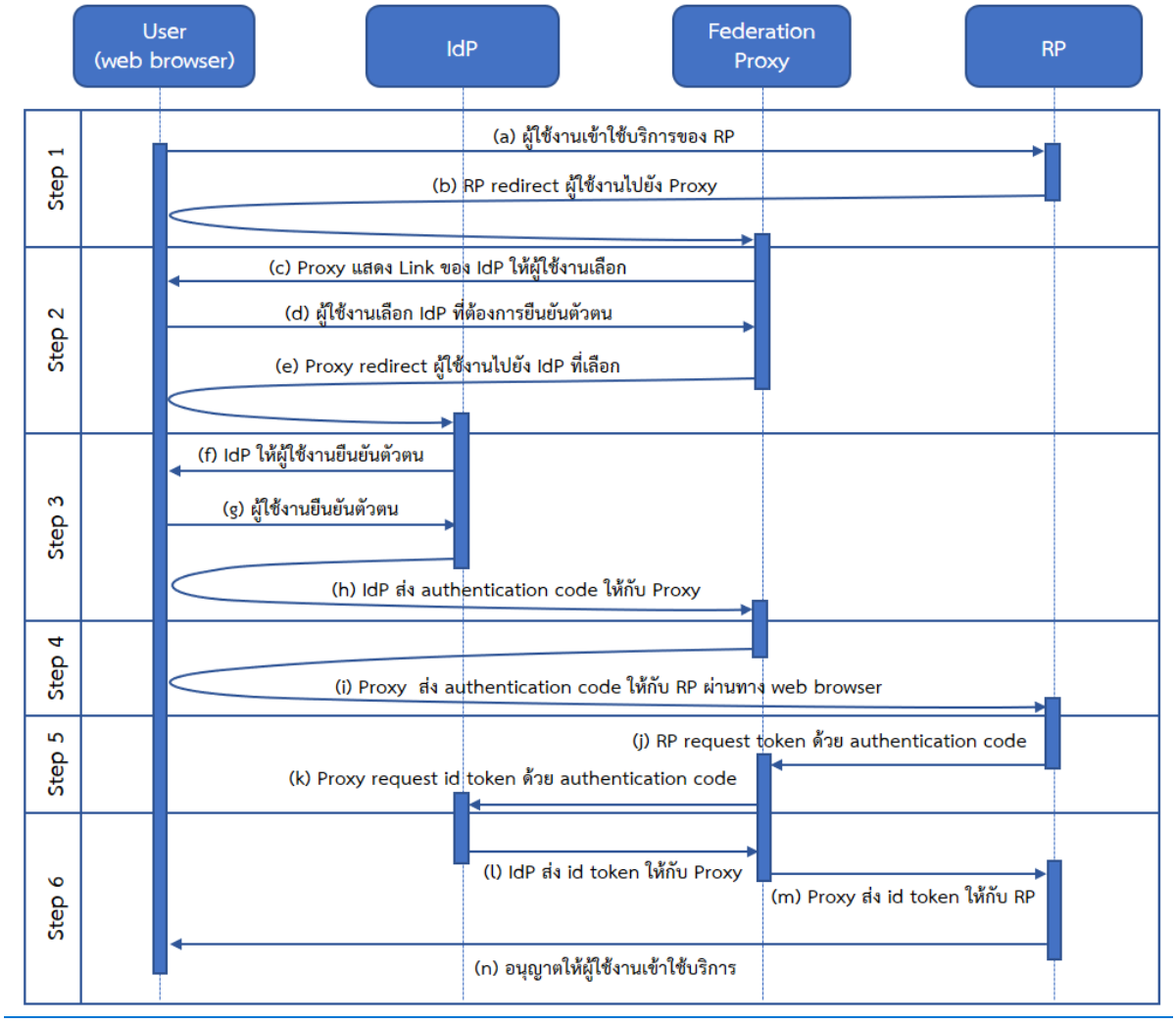
1.1.2 ข้อมูลที่ IdP จะส่งให้กับ Federation Proxy

พารามิเตอร์	พารามิเตอร์ (ภาษาไทย)	คำอธิบาย
IdP Name (TH)	**ชื่อหน่วยงาน (ไทย)**	ชื่อของ IdP ภาษาไทย (ห้ามมีอักขระพิเศษ)
IdP Name (EN)	**ชื่อหน่วยงาน (English)**	ชื่อของ IdP ภาษาอังกฤษ (ห้ามมีอักขระพิเศษ)
IdP Short Name	**ชื่อย่อ**	ชื่อย่อหน่วยงาน
Sector	**ประเภทหน่วยงาน**	ประเภทหน่วยงานของ IdP
Address	**ที่อยู่**	บ้านเลขที่ ชื่ออาคาร ถนน
Sub-District	**ตำบล**	แขวง/ตำบล
District	**อำเภอ**	เขต/อำเภอ
City	**จังหวัด**	จังหวัด
Post Code	**รหัสไปรษณีย์**	รหัสไปรษณีย์
Mobile	**โทรศัพท์เคลื่อนที่**	เบอร์โทรศัพท์เคลื่อนที่ของ IdP
Phone	**โทรศัพท์สำนักงาน**	เบอร์โทรศัพท์สำนักงานของ IdP
Client ID		Client ID ที่ออกให้กับ Federation Proxy
Client Secret		Client Secret ที่ออกให้กับ Federation Proxy ใช้ในการขอ ID Token
IdP Logo		รูปภาพ logo ของ IdP
Authorization Endpoint		URL ที่ใช้ใช้ในการ Redirect user ไปทำการยืนยันตัวตน
Token Endpoint		URL ที่ใช้ในการนำ Authorization Code ไปแลกเป็น ID Token

พารามิเตอร์	พารามิเตอร์ (ภาษาไทย)	คำอธิบาย
JWKs Endpoint		URL ที่มีข้อมูล JWKs ตาม https://tools.ietf.org/html/draft-ietf-jose-json-web-signature-41
Issuer		ชื่อผู้ออก Id Token
Discovery Endpoint		รายละเอียดในการเชื่อมต่อข้อมูลกับ IdP

2. MESSAGE FLOW

2.1 กระบวนการทำงานของการยืนยันตัวตนทางอิเล็กทรอนิกส์โดยมี Federation Proxy



2.2 ขั้นตอนการทำงานของการยืนยันตัวตนทางอิเล็กทรอนิกส์โดยมี Federation Proxy

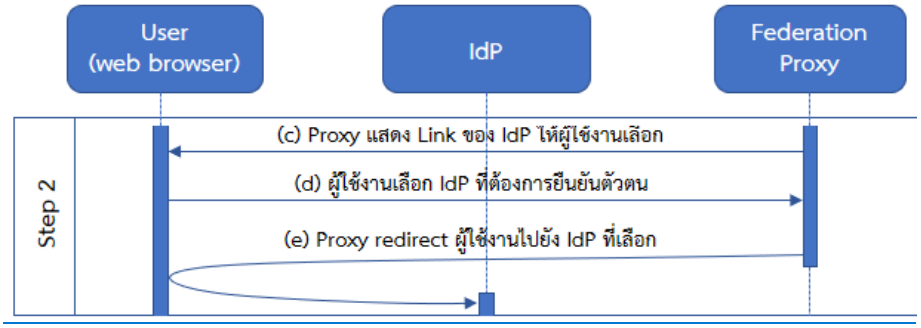
1. ผู้ใช้งานพิมพ์ URL ของ Relying Party เพื่อเข้าใช้งานระบบผ่านทาง Web Browser
2. Relying Party ทำการร้องขอการยืนยันตัวตนผู้ใช้งานไปยังระบบ Federation Proxy พร้อมทั้งกำหนด
    - เงื่อนไขในการแสดง IdP List เช่น Level of Assurance (LoA) เป็นต้น และ
    - รายละเอียดข้อมูลที่ Relying Party ต้องการ เช่น ชื่อ นามสกุล และหมายเลขประจำตัวประชาชน (ในกรณีที่ต้องการข้อมูลเพื่อระบุได้ว่าเป็นบุคคลใด) หรือ ข้อมูลประกอบการดำเนินการรู้จักลูกค้า (Know Your Customer: KYC) เป็นต้น ทั้งนี้ อาจมีการเพิ่มเติมรายละเอียดเพิ่มเติมได้
3. ระบบ Federation Proxy ทำการตรวจสอบคุณสมบัติของ IdP และข้อมูลที่จำเป็นต่อการแสดงรายการ Identity Provider (IdP List) ที่สอดคล้องกับเงื่อนไขในการแสดง IdP List ที่ Relying Party ต้องการ
4. ผู้ใช้งานทำการเลือก Identity Provider ที่ต้องการยืนยันตัวตน
5. ผู้ใช้งานถูก redirect ไปยัง Identity Provider ที่เลือก พร้อมรายละเอียดข้อมูลที่ Relying Party ต้องการ
6. Identity Provider ทำการยืนยันตัวตนผู้ใช้งาน
7. ผู้ใช้งานทำการยืนยันตัวตน หากการยืนยันตัวตนสำเร็จ Identity Provider จะต้องแสดงข้อมูลของผู้ใช้งานบนหน้าจอ พร้อมทั้งให้ผู้ใช้ยืนยันความถูกต้องของข้อมูลและยินยอม (Consent) ในการเปิดเผยข้อมูลแก่ Relying Party
8. เมื่อผู้ใช้งานยืนยันความถูกต้องของข้อมูลและยินยอมเปิดเผยข้อมูลแล้ว ผู้ใช้งานถูก redirect ไปยัง Federation Proxy พร้อมผลการยืนยันตัวตน ซึ่งเรียกว่า Authentication code ให้กับ Proxy
9. Proxy ทำการส่ง Authentication code ให้กับ Relying Party ผ่านทาง Web Browser
10. Relying Party ทำการขอข้อมูลจาก Identity Provider โดยการส่ง Authentication code ไปยัง Proxy
11. Proxy นำ Authentication Code ที่ได้รับจาก Relying Party ส่งไปยัง Identity Provider เพื่อขอข้อมูล
12. Identity Provider ตรวจสอบ Authentication Code หาก Authentication Code ถูกต้อง Identity Provider จะส่งข้อมูลมายัง Proxy ซึ่งเรียกว่า Assertion โดยการระบุข้อมูลไว้ใน ID Token ซึ่ง Assertion ต้องถูกลงลายมือชื่ออิเล็กทรอนิกส์ด้วยกุญแจส่วนตัว (Private key) ของ Identity Provider
13. Federation Proxy ทำการตรวจสอบ Assertion และทำการดึงข้อมูลจาก Assertion ของ IdP เพื่อมาสร้าง Assertion ที่ลงลายมือชื่ออิเล็กทรอนิกส์อีกครั้ง ด้วยกุญแจส่วนตัว (Private key) ของ Federation Proxy หลังจากนั้น Federation Proxy จะทำการ redirect ผู้ใช้งานพร้อม Assertion ไปยัง Relying Party

14. Relying Party ทำการตรวจสอบ Assertion ว่าถูกส่งมาจาก Federation Proxy และ Identity Provider จริง หาก Assertion ถูกต้อง Relying Party ก็สามารถเชื่อถือได้ว่าผู้ใช้งานได้ทำการยืนยันตัวตนแล้วกับ Identity Provider และอนุญาตให้ผู้ใช้งานเข้าใช้ระบบได้

2.3 ขั้นตอนการยืนยันตัวตนทางอิเล็กทรอนิกส์ที่เกี่ยวข้องกับ IdP มีดังต่อไปนี้

ผู้ใช้งานเข้าถึงเว็บไซต์ของ RP เพื่อขอใช้บริการ จากนั้นผู้ใช้งานกดปุ่มบนเว็บไซต์ RP เพื่อขอยืนยันตัวตนผ่านระบบ Federation Proxy จากนั้น RP จะทำการ redirect หน้าเว็บบราวเซอร์ไปยังหน้าจอยืนยันตัวตนของระบบ Federation Proxy เพื่อให้ผู้ใช้งานเลือก IdP โดยการส่ง Authentication Request

2.3.1 ขั้นตอนการเลือก IdP



(C) ระบบ Federation Proxy แสดงรายการ IdP ให้ผู้ใช้งานเลือก (d) โดยผู้ใช้งานเลือก IdP โดยคลิก Link รายการของ IdP ที่ผู้ใช้งานต้องการยืนยันตัวตน (e) จากนั้น ระบบ Federation Proxy จะ redirect หน้า web browser ด้วย HTTP GET โดยมีพารามิเตอร์ ดังต่อไปนี้

ตัวอย่าง HTTP Request

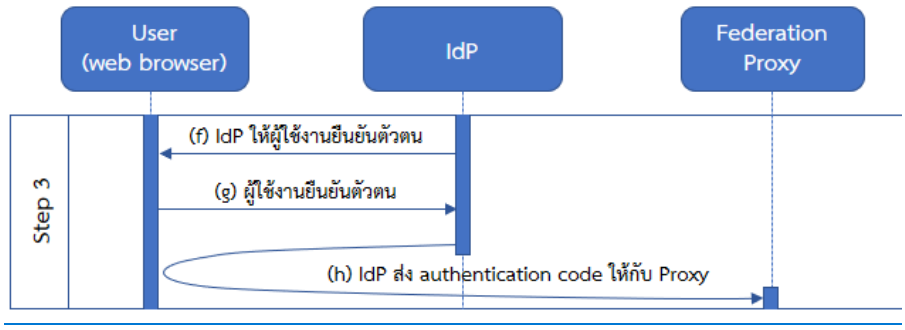
```
GET   https://idp.example.com/authorize?
      response_type=code
      &client_id=ae2fdDyId8asUW8sF9Ef0w
      &redirect_uri= https://proxy.digitalid.or.th/callback
      &scope=openid%20profile
      &state=usf3svanojasfninm6kg9s
      &prompt=login%20consent
```

คำอธิบายพารามิเตอร์

พารามิเตอร์	Required	รายละเอียด
response_type	Required	กำหนดค่าเป็น “code”

พารามิเตอร์	Required	รายละเอียด
client_id	Required	Identifier ของระบบ Federation Proxy ที่ลงทะเบียนไว้กับระบบ IdP
scope	Required	ขอบเขตของข้อมูลที่ RP ร้องขอ โดยค่าของ scope ให้กำหนดเป็น openid%20 และตามข้อมูลขอบเขตของข้อมูลที่กำหนดในหัวข้อ 3.4 เช่น profile หรือ profile_kyc
redirect_uri	Required	กำหนดค่าเป็น HTTP endpiont (URL) ของระบบ Federation Proxy ใช้สำหรับ redirect กลับไปยัง เว็บไซต์ของระบบ Federation Proxy เมื่อการยืนยันตัวตนเสร็จสิ้น
state	Required	string ใช้ในการตรวจสอบความสัมพันธ์ระหว่าง request กับ response ที่สร้างขึ้นจากระบบ Federation Proxy กับ IdP
prompt	Required	ให้กำหนดค่าเป็น “login consent ”

2.3.2 IdP จะส่ง Authentication Code ให้กับระบบ Federation Proxy



(f) IdP จะแสดงหน้าจอเพื่อให้ผู้ใช้งานยืนยันตัวตน **(g)** จากนั้นผู้ใช้งานทำการยืนยันตัวตนบนหน้า web browser ผ่านหน้า login ของ IdP หากผู้ใช้งานยืนยันตัวตนถูกต้องแล้ว **(h)** IdP จะส่ง authentication code กลับไปยังระบบ Federation Proxy ด้วย HTTP GET โดยมีพารามิเตอร์ดังต่อไปนี้

ตัวอย่าง HTTP Request

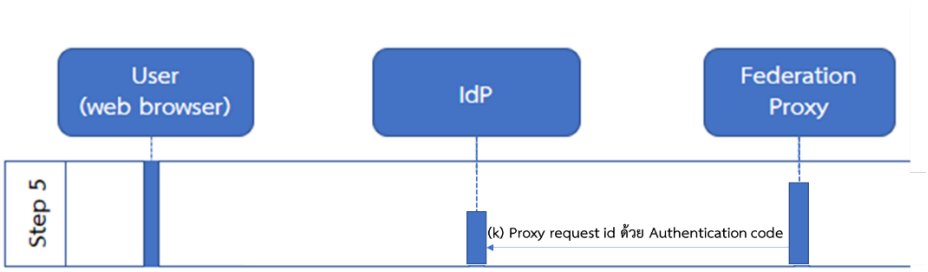
```
GET   https://openid1.digitalid.or.th/callback?
      code=SplxlOBeZQQYbYS6WxSbIA
      &state=usf3svanojasfninm6kg9s
```

คำอธิบายพารามิเตอร์

พารามิเตอร์	Required	รายละเอียด
code	Required	authentication code ที่ได้รับจาก IdP
state	Required	string ที่ใช้ในการตรวจสอบความสัมพันธ์ระหว่าง request กับ response ที่สร้างขึ้นจากระบบ Federation Proxy

จากนั้น Federation Proxy ส่ง Authentication Code ผ่านทาง Web Browser ให้ RP จากนั้นเมื่อ RP ร้องขอข้อมูล ID Token จากไปยัง Federation Proxy ด้วย Authentication Code

2.3.3 ระบบ Federation Proxy ร้องขอข้อมูล ID Token จาก IdP



ระบบ Federation Proxy ร้องขอข้อมูล ID Token จาก IdP โดยการส่ง HTTP POST พร้อมกำหนดพารามิเตอร์ “Authorization ในส่วนของ ”header ตามที่ระบุในมาตรฐาน HTTP Basic Authentication และ Oauth 2.0 section 2.3.1 ซึ่งพารามิเตอร์นี้จะถูก Encode ด้วย Base64)client_id + ”:”+client_secret ของระบบ (Federation Proxy

ตัวอย่าง HTTP Request

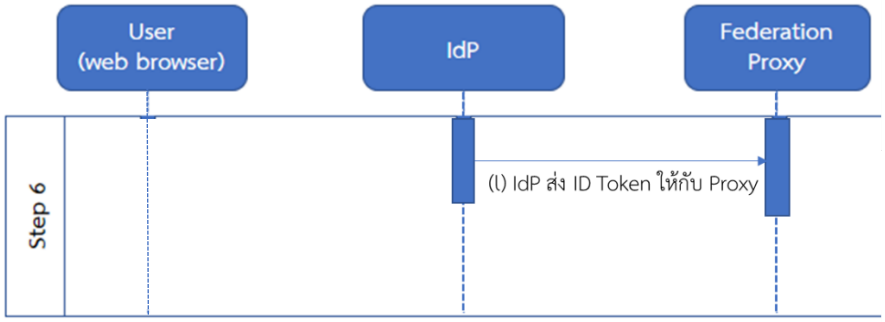
```
POST    https://idp.example.com/oauth/token HTTP/1.1
        Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW
        Content-Type: application/x-www-form-urlencoded

        grant_type=authorization_code
        code=SplxlOBeZQQYbYS6WxSbIA
        redirect_uri= https://proxy.digitalid.or.th/callback
```

คำอธิบายค่า พารามิเตอร์

พารามิเตอร์	Required	รายละเอียด
grant_type	Required	กำหนดค่าเป็น “authorization_code”
code	Required	Authentication Code ที่ได้รับจาก IdP

redirect_uri	Required	HTTPS URL ของ Federation Proxy ใช้ระบุช่องทางสำหรับส่งข้อมูล assertion กลับมายัง RP ทั้งนี้ ค่าของ redirect_uri จะต้องเป็น URL ที่ลงทะเบียนไว้กลับ IdP

2.3.4 IdP ส่งข้อมูล ID Token ให้กับ Federation Proxy



IdP ส่งข้อมูล Assertion (ตามตัวอย่าง HTTP Response ด้านล่าง) มายังระบบ Federation Proxy ซึ่ง ใน Assertion ดังกล่าว มี ID Token (id_token) ถูกรับรองโดยการลงลายมือชื่ออิเล็กทรอนิกส์ด้วยกุญแจส่วนตัว (Private key) ของ IdPProxy ส่ง ID Token ให้กับ RP และอนุญาติให้ผู้ใช้งานเข้าใช้บริการ

ตัวอย่างการส่ง HTTP Response ของ IdP

```
Content-Type: application/json
Cache-Control: no-cache, no-store
Pragma: no-cache
{
"access_token":"SlAV32hkKG",
"token_type":"Bearer",
"expires_in":3600,
"id_token":"eyJ0NiJ9.eyJ1cI6IjIifX0.DeWt4QuZXso ...",
}
```

คำอธิบายค่าพารามิเตอร์

พารามิเตอร์	Required	รายละเอียด
access_token	Required	เป็น token เพื่อใช้เข้าถึงบริการหรือข้อมูลจากผู้ให้บริการต่างๆ ที่เชื่อถือ access_token ดังกล่าว ซึ่งออกให้โดย IdP

พารามิเตอร์	Required	รายละเอียด
token_type	Required	กำหนดค่าเป็น Bearer เสมอ
expires_in	Required	อายุการใช้งานของ access_token มีระยะเวลาเป็นวินาทีนับจากเวลาของการเริ่มสร้าง access_token
id_token	Required	ข้อมูลอัตลักษณ์ของผู้ใช้งานอยู่ในรูปแบบ JWT (JSON Web Token) ถูกรับรองความถูกต้องครบถ้วน (Integrity) โดยลงลายมือดิจิทัลด้วยกุญแจส่วนตัว (Private Key) ของ IdP

ทั้งนี้ ในกรณีที่มีการกำหนด scope เป็น “openid profile” รายการข้อมูลของผู้ใช้งานใน id_token จะถูกกำหนดตาม ตารางที่ 5.4.1 สำหรับกรณีที่กำหนด scope เป็น “openid profile_kyc” เพื่อใช้ในกระบวนการรู้จักลูกค้า (KYC : Know Your Customer) รายการข้อมูลของผู้ใช้งานใน id_token จะถูกกำหนดตาม ตารางที่ 5.4.2

หลังจากนั้น Federation Proxy ทำการส่ง ID Token ให้กับ RP และ RP จะอนุญาตให้ผู้ใช้งานเข้าใช้บริการในระบบของ RP

3. CONFIGURATION ของ IDP เพื่อต้องการเชื่อมต่อกับ PROXY

ในการที่ IdP ที่จะให้บริการพิสูจน์และยืนยันตัวตนผ่าน Proxy นั้น IdP จำเป็นต้องปรับแต่งระบบให้สามารถสื่อสารกับ Proxy ผ่าน ด้วยโปรโตคอล OpenID Connect (OIDC) โดย IdP จำเป็นต้องกำหนด Configurations และรายการข้อมูลสำหรับ Response ให้เหมาะสมและสอดคล้องกับรายการข้อมูลในการ Request ที่ได้รับจาก Proxy มี Minimum Requirement ดังต่อไปนี้

3.1 Discovery Document

ใน OpenID Protocol นั้น Proxy หรือ IdP สามารถกำหนดรายละเอียดในการเชื่อมต่อข้อมูลผ่าน URL (Discovery Document) ในรูปแบบ https://{IdP _URL}/.well-known/openid-configuration เพื่อให้ Proxy ใช้เป็นข้อมูลในการเชื่อมต่อระบบ โดยข้อมูลใน Discovery Document จะระบุ endpoint และเซ็ตของค่าพารามิเตอร์ต่าง ๆ อย่างน้อย ดังตารางต่อไปนี้

พารามิเตอร์	Required	ค่าที่กำหนดสำหรับ IdP
issuer	Required	https://idp.example.com
authorization_endpoint	Required	https://idp.example.com/authorize
token_endpoint	Required	https://idp.example.com/token

พารามิเตอร์	Required	ค่าที่กำหนดสำหรับ IdP
jwks_uri	Required	https://idp.example.com/jwks
scopes_supported	Required	• openid • profile (หมายเหตุ: หากมี scope นอกเหนือจากที่ระบุไว้ ระบุทั้งหมด)
response_types_supported	Required	• code
grant_types_supported	Required	• authorization_code
subject_types_supported	Required	• public
id_token_signing_alg_values_supported	Required	• RS256 • RS384 • RS512 • ES256 • ES384 • ES512
claims_supported	Required	อ้างอิงจาก ตารางคุณสมบัติของข้อความ 3.4.1 และ - 3.4.2

ตัวอย่างการ Request Discovery Endpoint

```
GET  https://idp.example.com/.well-known/openid-configuration
```

ตัวอย่างของการ Response Discovery Document

```
HTTP/1.1 200 OK
  Content-Type: application/json
  {
   "issuer":
     "https://proxy.example.com",
   "authorization_endpoint":
     "https://proxy.example.com/authorize",
   "token_endpoint":
```

```
    "https://proxy.example.com/token",
  "jwks_uri":
    "https://proxy.example.com/jwks",
  "response_types_supported":
    ["code"],
  "subject_types_supported":
    ["public"],
  "grant_types_supported":
    ["authorization_code"],
  "id_token_signing_alg_values_supported":
    ["RS256", "RS384", "RS512", "ES256", "ES384", "ES512"],
  "claims_supported":
    ["sub", "iss", "auth_time", "acr",
     "given_name", "family_name", "national_id", "passport_number", "acr",
     "https://proxy.example.com/info/claims"
    ]
  }
```

3.3 ID Token

ข้อมูล id_token จะอยู่ในรูปแบบ JSON Web Signature (JWS) จะประกอบด้วยข้อมูล 3 ส่วน ได้แก่ ส่วน Header ส่วน Payload และ ส่วน Signature

3.3.1 ส่วน Header

รายการข้อมูล	Required	รายละเอียด
typ	Required	รูปแบบของ ID Token ให้กำหนดเป็น “JWT”
alg	Required	พารามิเตอร์กำหนดอัลกอริทึมที่ใช้ในกระบวนการ hashing และการเข้ารหัสลับ “RS256” = RSA using SHA-256 hash algorithm “RS384” = RSA using SHA-384 hash algorithm “RS512” = RSA using SHA-512 hash algorithm “ES256” = Elliptic Curve using SHA-256 hash algorithm “ES384” = Elliptic Curve using SHA-384 hash algorithm “ES512” = Elliptic Curve using SHA-512 hash algorithm

รายการข้อมูล	Required	รายละเอียด
x5c	Optional	X.509 Public Key Certificate หรือ Certificate Chain ที่เป็นคู่กุญแจที่ใช้ในการลงลายมือชื่อดิจิทัล *** <u>หมายเหตุ</u> *รายการข้อมูลนี้ จะปรากฏในขั้นตอน* ***(l)*** *IdP ส่ง ID Token ให้กับ* Proxy

3.3.2 ส่วน Payload

รายการข้อมูล	Required	รายละเอียด
iss	Required	Identifier ของผู้ตอบข้อมูล id_token กลับ (IdP / Proxy) โดยกำหนดรูปแบบเป็น https url
sub	Required	Unique ID จาก IDP เช่น หมายเลขบัตรประชาชน
aud	Required	Proxy Client ID
exp	Required	ข้อมูลเวลาหมดอายุของ ID Token อยู่ในรูปแบบของ UNIX timestamp
iat	Required	ข้อมูลเวลาของ ID Token ขณะที่ถูกสร้างขึ้น อยู่ในรูปแบบของ UNIX timestamp
acr	Required	ระบุ IAL, AAL

นอกเหนือจากรายการข้อมูลตามตารางด้านบนแล้ว มีรายการข้อมูลที่เกี่ยวข้องกับผู้ใช้งาน เช่น ชื่อ ที่อยู่ อีเมล ของผู้ใช้งาน ขึ้นอยู่กับการกำหนดพารามิเตอร์ scope ในขั้นตอน (b) หากกำหนด scope เป็น *profile* จะมีรายการข้อมูลเพิ่มเติมตามตาราง 5.5.1 และหากกำหนด scope เป็น *profile_kyc* จะมี รายการข้อมูลเพิ่มเติมตามตาราง 5.5.2

3.3.3 ส่วน Signature

ลายมือชื่ออิเล็กทรอนิกส์ใน JWS จะมีรูปแบบที่แตกต่างกันไปขึ้นอยู่กับอัลกอริทึม (Algorithm) ที่ IdP ใช้ในการลงลายมือชื่ออิเล็กทรอนิกส์ เช่น RSA เป็นต้น

ตัวอย่าง ID Token ที่ IdP ส่งให้ Proxy

HEADER

```
{
    "alg": "RS256",
    "typ": "JWT",
```

"x5c": ["MIIDQjCCAiqgAwIBAgIGATz/FuLiMA0GCSqGSIb3D ... BqLdElrRhjZkAzVvb3du6/KFUJheqw NTrZEjYx8 OuH0aBsXBTWVU+4=", "MIIE+zCCBGSgAwIBAgICAQ0wDQYJKoZIhvcNAQEFBQ ... Awgbsx JDAiBgNVBAcTTmV0d29yazEXMBUGA1UNlcnQsxTXdMwzzjsvI"]

}

PAYLOAD

{
"iss": " http://idp01.com",
"sub": "114386995432676543513",
"aud": "dcd27uq4nojetqu8e1kf8p8vatsbnd55",
"exp": 1519798006,
"iat": 1519794406,
"given_name": "Somchai",
"family_name": Wahnpong,
"national_id": 4724747767301,
"passport_number": AA7562739
}

SIGNATURE

dLP19D4HoJ_6E-0vAsufmli8C58LlSHpCO1VFOFKnJe5rW20egUxnzWENA5Pxd2F5FHX7quOHTKVzw 1EtpQjGdAuaVAfl5e42vI8AnDPPMymcsLC2zKthDCnYud6cN7ciemI7vx9ysmyrmVRqT-Jen9JRL6FTdv 3QH_DQHLAaPClw-_fAFVYVz7k8pEGJ2wQL8RANMF2zil-bG8tZmAW4OwqZB_ sj9fCmmiwHrXmWa QduS9ceSpRbdDngcjs8lOwTqoA4fqeI147Vzc6HFAvQ

3.4 ข้อมูลสำหรับการใช้งานบริการของผู้ใช้งาน

IdP จะตอบ assertion กลับให้ Proxy หลังจากที่ผู้ใช้งานทำการ ประเภทของข้อมูลที่รับส่งสามารถกำหนดได้จากรูปแบบ (Profile) ข้อมูลที่กำหนดไว้ในคำร้องขอการยืนยันตัวตน (Authentication Request) โดย

1) รูปแบบ (Profile) ของการยืนยันตัวตน (Authentication)

2) รูปแบบ (Profile) ของการรู้จักลูกค้า (KYC)

โดยมีรายการข้อมูลดังตารางดังต่อไปนี้

ตารางที่ 3.4.1 รายการข้อมูลใน **id_token** เมื่อกำหนด **scope** เป็น “profile” ใช้สำหรับการยืนยันตัวตน (Authentication)

No.	Short Name	Type	คำอธิบาย	Required / Optional
1	**given_name**	string	ชื่อ	Required
2	**family_name**	string	นามสกุล	Required
3	**national_id**	string	เลขประจำตัวประชาชน	Required (กรณีบุคคลที่มีสัญชาติไทย)
4	**passport_number**	string	หนังสือเดินทาง	Required (กรณีบุคคลต่างชาติ)

ตารางที่ 3.4.2 รายการข้อมูลใน **id_token** เมื่อกำหนด **scope** เป็น “profile_kyc” ใช้สำหรับการรู้จักลูกค้า (KYC)

No.	Short Name	Type	คำอธิบาย	Required / Optional
1	**given_name**	string	ชื่อ	Required
2	**family_name**	string	นามสกุล	Required
3	**national_id**	string	เลขประจำตัวประชาชน	Required (กรณีบุคคลที่มีสัญชาติไทย)
4	**passport_number**	string	หนังสือเดินทาง	Required (กรณีบุคคลต่างชาติ)
6	**birthdate**	string	วัน เดือน ปีเกิด	Required
7	**address**	JSON object	ที่อยู่ที่อาศัย	Required
7.1	formatted	string	ที่อยู่ (ไม่มีโครงสร้าง)	Optional
7.2	street_address	string	บ้านเลขที่ , ถนน , ตำบล	Optional
7.3	locality	string	เขต/อำเภอ	Required
7.4	region	string	จังหวัด	Required
7.5	postal_code	string	รหัสไปรษณีย์	Optional
7.6	country	string	ประเทศ	Optional
8	**career**	string	อาชีพ	Required
9	**business_address**	JSON object	สถานทำงาน	Required
9.1	formatted	string	ที่อยู่ (ไม่มีโครงสร้าง)	Optional

No.	Short Name	Type	คำอธิบาย	Required / Optional
9.2	street_address	string	บ้านเลขที่ , ถนน , ตำบล	Optional
9.3	locality	string	เขต/อำเภอ	Required
9.4	region	string	จังหวัด	Required
9.5	postal_code	string	รหัสไปรษณีย์	Optional
9.6	country	string	ประเทศ	Optional
10	**phone_number**	string	หมายเลขโทรศัพท์	Required
11	**email**	string	อีเมล	Required

3.5 JWK (JSON Web Key)

JWK (JSON Web Key คือ คีย์ที่ใช้ในการรับรองข้อมูล ถูกนำไปใช้ตรวจสอบข้อมูลที่ได้รับจาก IdP โดย RP จะทำการเรียกไปยัง JWKS Endpoint ซึ่งมีพารามิเตอร์ของ Public Key ดังต่อไปนี้

พารามิเตอร์	Required	รายละเอียด
alg	Required	อัลกอริทึมของคีย์ กำหนดให้ใช้ “RS256”
kty	Required	ชนิดของคีย์ กำหนดเป็น “RS”
use	Required	การใช้งานของคีย์เช่นใช้สำหรับการลงนามรับรอง กำหนดเป็น “sig”
x5c	Required	x.509 Certificate Chain
e	Required	ค่า exponent ของ pem
n	Required	ค่า modulus ของ pem
kid	Required	ค่าเฉพาะที่ไม่ซ้ำกันของคีย์ (unique identifier)
x5t	Optional	Thumbprint ของ x.509 Certificate (SHA-1 thumbprint)

ตัวอย่างการ Response ของ JWK

```
{
"keys": [
 {
   "alg": "RS256",
   "kty": "RSA",
   "use": "sig",
   "x5c": [
```

```
        "MIICszfY3BR9TPK8xmMmQwtIvLu1PMttNCs7niCYkSiUv2sc2mlq1i3IashGkkgmo=....."
        ],
        "n": " _TMDg7pOWm_zHtF53qbVENoejj_ytspMmGW7yMRxzUqgxcAqOBpV.....",
        "e": "AQAB",
        "kid": "NjVBRjY5MDlCMUIwNzU4RTA2QzZFMDQ4QzQ2MDAyQjVDNjk1RTM2Qg",
        "x5t": "NjVBRjY5MDlCMUIwNzU4RTA2QzZFMDQ4QzQ2MDAyQjVDNjk1RTM2Qg"
      }
    ]}
```

4. ข้อความแจ้งกลับข้อผิดพลาด ERROR RESPONSE

ในกระบวนการยืนยันตัวตนทางอิเล็กทรอนิกส์ด้วยโปรโตคอล OpenID Connect กำหนดรายละเอียดในการแจ้งกลับข้อผิดพลาด (Error Response) ของ Authentication Request โดยมีต้องกำหนดพารามิเตอร์ดังต่อไปนี้

พารามิเตอร์	Required	รายละเอียด
error	Required	Error code แจ้งสาเหตุของข้อผิดพลาดที่เกิดขึ้น
error_description	Optional	ใช้ในการแสดงรายละเอียดข้อผิดพลาดของระบบ ในรูปแบบ text
state	Required	string ใช้ในการตรวจสอบความสัมพันธ์ระหว่าง request กับ response
error_uri	Optional	URI เพื่อแจ้งข้อมูลของ Error ที่ระบุ โดยอยู่ในรูปแบบ Webpage

ตัวอย่าง Error Response ของ Authentication Request

```
HTTP/1.1 302 Found
  Location: https://rp.example.org/callback?
    error=invalid_request
    &error_description=
      Unsupported%20response_type%20value
    &state=af0ifjsldkj
```

สำหรับพารามิเตอร์ในการแจ้งกลับข้อผิดพลาด (error response) สามารถระบุ Error code ได้ดังตารางต่อไปนี้

Error code	คำอธิบาย	HTTP Return Code
invalid_request	ข้อมูล request ไม่ถูกต้อง	302
unauthorized_client	Client ไม่ได้รับอนุญาตให้ร้องขอด้วยวิธีการที่ระบุ	302
unsupported_response_type	Server ไม่รองรับการร้องขอตามที่ระบุ	302
invalid_scope	ข้อมูล Scope ที่ร้องขอไม่ถูกต้อง หรือขาดหาย	302
server_error	พบข้อผิดพลาดของ Server (Return Error 500 Internal Server Error เนื่องจากไม่สามารถส่งพารามิเตอร์ให้ Client ผ่านทาง HTTP Redirect ได้)	500
temporarily_unavailable	พบข้อผิดพลาดของ Server เนื่องจาก Server ไม่สามารถรองรับภาระ Load ได้ หรืออยู่ระหว่างการปรับปรุง Server (Return Error 503 Service Unavailable เนื่องจากไม่สามารถส่งพารามิเตอร์ให้ Client ผ่านทาง HTTP Redirect ได้)	503
interaction_required	Authorization Server ต้องการให้ผู้ใช้บริการทำการยืนยันตัวตนผ่านแบบฟอร์ม หรือ หน้าจอ ข้อมูล Error นี้อาจถูกส่งกลับในกรณีที่ Parameter Prompt ถูกกำหนดเป็น none การยืนยันตัวตนจึงไม่สามารถดำเนินการต่อได้	302

login_required	Authorization Server ต้องการให้ผู้ใช้บริการทำการยืนยันตัวตน ข้อมูล Error นี้อาจถูกส่งกลับในกรณีที่ Parameter Prompt ถูกกำหนดเป็น none การยืนยันตัวตนจึงไม่สามารถดำเนินการต่อได้	302
consent_required	Authorization Server ต้องการให้ผู้ใช้บริการทำการยินยอมเพื่อให้ข้อมูล Error นี้อาจถูกส่งกลับในกรณีที่ Parameter Prompt ถูกกำหนดเป็น none การให้ข้อมูลจึงไม่สามารถดำเนินการต่อได้	302
invalid_request_uri	ข้อมูล request_uri ไม่ถูกต้อง	400
invalid_request_object	ข้อมูล request_object ไม่ถูกต้อง	302

ในกระบวนการยืนยันตัวตนทางอิเล็กทรอนิกส์ด้วยโปรโตคอล OpenID กำหนดรายละเอียดในการแจ้งกลับข้อผิดพลาด (Error Response) ของ Token Request โดยมีต้องกำหนดพารามิเตอร์ดังต่อไปนี้

```
HTTP/1.1 400 Bad Request

Content-Type: application/json

Cache-Control: no-store

Pragma: no-cache

{

    "error": "invalid_request"

}
```

Error code	คำอธิบาย	HTTP Return Code
invalid_request	ข้อมูล request ไม่ถูกต้อง	400
invalid_client	ข้อมูล client ไม่ถูกต้อง หรือ ไม่รองรับการวิธีการร้องขอที่ client ระบุ	401
invalid_grant	ข้อมูล authorization code ไม่ถูกต้องหรือหมดอายุ	400
unauthorized_client	Client ไม่ได้รับอนุญาตให้ร้องขอด้วยวิธีการที่ระบุ	400
unsupported_grant_type	Authorization Server ไม่รองรับ grant type ตามที่ระบุ	400
invalid_scope	ข้อมูล Scope ที่ร้องขอไม่ถูกต้อง หรือขาดหาย	400

ภาคผนวก ก. รูปแบบและข้อกำหนดของพารามิเตอร์

การกำหนดค่าของพารามิเตอร์ สามารถกำหนดค่าได้ตามรูปแบบดังต่อไปนี้

ชื่อพารามิเตอร์	Data Type	Validate format	Min	Max
response_type	**string**	{“code”}	-	-
client_id (Federation Proxy)	**string**	a-z A-Z 0-9 -	32	100
scope	**string**	{“openid”, “profile”, “profile_kyc”}	-	-
redirect_uri	**string**	https URL format	10	250
state	**string**	a-z A-Z 0-9	10	100
prompt	**string**	{“login consent”}	-	-
acr_values	**string**	urn:did:ial:{1_1, 1_2, 1_3, 2_1, 2_2, 2_3, 3} urn:did:aal:{1, 2_1, 2_2, 3} urn:did:sector:[sector name] urn:did:idp:[IdP short name]	-	-
code	**string**	a-z A-Z 0-9	10	100
expire_in	**string**	0-9	5	100
grant_type	**string**	{“authorization_code”}	-	-
access_token	**string** **or JWT**	**กรณีเป็น string** a-z A-Z 0-9 **กรณีเป็น JWT** [a-z A-Z 0-9]n . [a-z A-Z 0-9]n . [a-z A-Z 0-9]n	-	-

ชื่อพารามิเตอร์	Data Type	Validate format	Min	Max

หมายเหตุ

- ในคอลัมน์ Valdiate format เมื่อมีการระบุค่าในเครื่องหมาย { } ให้ตรวจสอบค่าของพารามิเตอร์ให้เป็นไปตามค่าที่กำหนดในเครื่องหมาย { } เท่านั้น