Issued:	Enters into force:	Validity:
[pp.kk.vvvv]	[pp.kk.vvvv]	[esim. toistaiseksi]

Legal basis:

Modification details:

The Recommendation will be supplemented and modified as necessary. In that case, the Recommendation number 213 will be maintained, but the date and the year will be changed appropriately. The modified versions of the Recommendation are listed in the following table:

Recommendation version and date	**Modifications**
Published recommendation v1.0 213/2018 S 2018-01-26	First published version
Draft version v2.0 213 2020	Added change requests from previous years (2018 → 2020)
Draft version v2.0.1 213 2020	Tidying up after first technical workgroup meeting and integrating the 4 proposals from FTN stakeholders.
Draft version v2.0.2 213 2020	Error response definitions added, TLS 1.1 removal. SSO reverted to the original text (2018 recommendation). Final draft for comments to the technical working group.

Current recommendation version is published on the Traficom website at

https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/regulation/ftn_oidc_profile_v1.0_ficora_rec_213_2018_s.pdf

Johdanto ja OpenID Connect rajapintasuosituksen tarkoitus

Asiakirjan nimi
Finnish Trust Network OpenID Connect Protocol Profile version 2.0

Tiivistelmä

Vahvasta sähköisestä tunnistamisesta ja sähköisistä allekirjoituksista annetun lain (2009/617) 12 a §:n 2 momentin mukaan luottamusverkostoon kuuluvien tunnistuspalvelun tarjoajien on muun muassa tarjottava tekniset rajapinnat, jotka luovat edellytykset tunnistuspalveluita tarjoavien ja niitä hyödyntävien toimijoiden väliselle toiminnalle. Tämä suositus on tarkoitettu luottamusverkostoon kuuluville tunnistuspalvelun tarjoajille.

Suositus määrittelee vahvan sähköisen tunnistuspalvelun tarjoajien luottamusverkostosta annetun asetuksen (2016/169) 1 §:n mukaisen tunnistusvälineen tarjoajan ja tunnistusvälityspalvelun tarjoajan välisen OpenID Connect protokollaa käyttävän rajapintakuvauksen. Suosituksessa on huomioitu Viestintäviraston määräyksen 72/2016 (määräys sähköisistä tunnistus- ja luottamuspalveluista) mukaiset vähimmäistiedot, joita toimijoiden välisissä rajapinnoissa on kyettävä siirtämään, sekä määräyksessä asetetut tietoliikenteen salausvaatimukset.

Suositus on laadittu yhteistyössä luottamusverkoston toimijoiden kanssa ja sen avulla toimijat voivat rakentaa yhteentoimivia järjestelmiä. Suositus on sovellettavissa myös asiointipalvelun ja luottamusverkoston välisessä rajapinnassa käytettäväksi. Yksinkertaisuuden vuoksi rajapintakuvauksessa käsitellään kuitenkin johdonmukaisesti toimintaa tunnistusvälineen tarjoajan ja tunnistusvälityspalvelun välisessä rajapinnassa.

Rajapintakuvaus on julkaistu vain englanniksi, jotta se olisi laajasti suoraan hyödynnettävissä ja jotta tulkintaeroja eri kieliversioiden välillä ei pääse syntymään.

Tälle suositukselle rinnakkaisena on julkaistu vastaavan toiminnallisuuden tarjoava SAML 2.0 protokollaa käyttävä rajapintakuvaus numerolla S 212 2021.

Avainsanat

luottamusverkosto, rajapinta, sähköinen tunnistaminen, OpenID Connect, OIDC

Inledning och syfte med OpenID Connect protokoll profil dokument

Namnet på dokumentet

Finnish Trust Network OpenID Connect Protocol Profile version 2.0

Referat

Enligt 12 a § 2 mom. i lagen om stark autentisering och elektroniska signaturer ska en leverantör av identifieringstjänster som hör till förtroendenätet bl.a. erbjuda tekniska gränssnitt som skapar förutsättningar för verksamheten mellan aktörerna som tillhandahåller identifieringstjänster och aktörerna som använder tjänsterna. Denna rekommendation är avsedd för de leverantörer av identifieringstjänster som hör till förtroendenätet.

Rekommendationen specificerar en gränssnittsbeskrivning för användning av OpenID Connect 1.0-protokollet mellan en leverantör av identifieringsverktyg och en leverantör av tjänster för förmedling av identifiering i enlighet med 1 § i förordningen om förtroendenätet för leverantörer av tjänster för stark autentisering (169/2016). I rekommendationen har beaktats den minimiuppsättning uppgifter som ska kunna överföras mellan aktörernas gränssnitt enligt Kommunikationsverkets föreskrift 72/2016 (föreskrift om elektroniska identifieringstjänster och betrodda elektroniska tjänster), samt kraven på trafikkryptering som anges i föreskriften.

Rekommendationen har utarbetats i samarbete med aktörerna i förtroendenätet, och med hjälp av den kan aktörerna bygga interoperabla system. Rekommendationen kan också tillämpas för användning i gränssnittet mellan ärendehanteringstjänster och förtroendenätet. För enkelhetens skull behandlas i gränssnittsbeskrivningen dock konsekvent verksamheten i gränssnittet mellan en leverantör av identifieringsverktyg och en leverantör av tjänster för identifieringsförmedling.

Gränssnittsbeskrivningen publiceras enbart på engelska för att vara direkt användbar för många och för att det inte ska bli några tolkningsskillnader mellan olika språkversioner.

Parallellt med denna rekommendation publiceras en gränssnittsbeskrivning för användning av SAML 2.0-protokollet som erbjuder motsvarande funktionalitet som OpenID Connect. Rekommendationens nummer är S 212 2021.

Nyckelord

förtroendenät, gränssnitt, elektronisk identifiering, OpenID Connect, OIDC

1 Introduction

This document defines the OpenID Connect protocol interface for the Finnish Trust Network (FTN). Specifically this means the interface between FTN IdPs and FTN Brokers, but it is also usable between Service Providers and FTN Brokers. Some recommendations are relevant to a uniform way of creating the user experiences (UX).

1.1 About the Finnish Trust Network

The Finnish Trust Network (FTN) is a mechanism for connecting large scale, consumer facing services with trusted identity and service providers. The Trust Network delivers the following benefits:

- For citizens, the FTN delivers a sign-on experience that is familiar, fast and simple.
- For online services, the FTN lowers barriers of security and complexity related to implementing strong authentication based on mutually accepted levels of assurance.
- For identity providers that issue authentication credentials, the FTN provides new opportunities to leverage the success of their credentials platform and expand credential usage.

The Trust Network follows the requirements and objectives of the European eIDAS regulation for a network of trust service providers enabling Citizen-to-Business-to-Government secure and trusted electronic service provisioning. The Network is built upon strong privacy and security principles and enables a user-centric attribute consent model.

1.2 Audience and Scope

This profile specifies OpenID Connect protocol requirements for identity providers that provide authentication credentials and authorisation services within the Finnish Trust Network. This specification is intended for online service providers integrating with the FTN as Identity Providers and Identity Service Brokers.

Although this specification is worded to define only the interface between an Identity Provider and an Identity Service Broker, the same interface and attributes are also directly usable between an FTN Broker and an online Service Provider/Relying Party. It is assumed the reader is generally familiar with the OpenID Connect (OIDC) protocol.

User consent information transfer is not included in the scope of this profile. Asking for user consent when needed is the responsibility of the party needing the consent. For the typical use case of authenticating a user to a Service Provider (without enrichment) the consent is implicit and it is not necessary for the FTN Broker or IdP to separately ask for user consent for each authentication transaction.

A sister document is published as FICORA Recommendation S 212 2021 to define a corresponding SAML2 profile for the FTN.

1.3 Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The use of SHOULD, SHOULD NOT, and RECOMMENDED reflects broad consensus on deployment practices intended to foster both interoperability and guarantees of security and confidentiality needed to satisfy the requirements of many organisations that engage in the use of federated identity. Deviating may limit a deployment's ability to technically interoperate without additional negotiation, and should be undertaken with caution.

This document uses native, eIDAS and STORK2 based abbreviations:

AP = Attribute Provider

AS = Authorisation Server

CA = Certificate Authority

FTN = Finnish Trust Network

(FTN) IdP = Identity Provider within the FTN

(FTN) Broker = Broker that handles authentication requests between Service Providers and IdPs in the FTN. The Broker MAY provide multiple interfaces for IdPs and Service Providers to integrate to.

JOSE = Javascript Object Signing and Encryption

JWA = JSON Web Algorithms (RFC 7518)

JWE = JSON Web Encryption (RFC 7516)

JWKS = JSON Web Key Set (RFC 7517)

JWS = JSON Web Signature (RFC 7515)

JWT = JSON Web Token (RFC 7519)

OIDC = OpenID Connect

SP = Service Provider, provides a service the end-user is trying to access and is being authenticated to, from the viewpoint of the FTN Broker

1.4 Single Sign On considerations

Single Sign On (SSO) authentications MUST NOT happen by chance in the FTN. All implementations MUST respect the built-in parameters in the OIDC protocol that can be used to limit or forbid the (re)use of cached/SSO authentications. Separate guidelines or reports on the use of SSO within the FTN may be published later.

2 System Requirements

2.1 OIDC Protocol

FTN IdP's and FTN Brokers must support communication using the **Authorization Code** flow of the OpenID Connect protocol. Other parties that implement the OpenID Connect protocol to communicate with the Finnish Trust Network (FTN) must be able to consume and send OIDC messages.

- Please visit the OpenID website for the specification of OpenID Connect protocol [i]. The OpenID Connect protocol specification is normative in the FTN, i.e. the OIDC Core protocol specification SHOULD be followed. Some minor exceptions to the protocol handling may be specified in this document for FTN use.
- Please see the OAuth 2.0 Threat Model and Security Considerations document for additional security considerations in IETF RFC 6819. [ii]

2.2 JSON Signature and Encryption Security

2.2.1 Keys

Asymmetric public signing/encryption keys MUST be exchanged beforehand between FTN IdP's and FTN Brokers to ensure they can process cryptographically protected OIDC messages (JWTs). The exact process used for exchange of these cryptographic keys is outside the scope of this document, but it MUST NOT rely solely on dynamic discovery, i.e. keys MUST be explicitly configured/pinned as trusted for OIDC use between the FTN IdP and Broker.

To facilitate co-use of multiple algorithms and key/algorithm rollover, FTN participants MUST be configurable to trust multiple keys simultaneously for each FTN peer. To follow cryptographic best practices, separate keys MUST be used for encryption and signing. JOSE header `kid` (RFC 7515) MUST be used for identifying the key used in all cryptographically secured JWTs.

2.2.2 Algorithms and key sizes

Cryptographic algorithms are listed in Table 1 for protection of JWTs within the FTN. These fulfil the encryption requirements of FICORA Regulation M72A. [iii] FTN participants MUST support algorithms that are marked REQUIRED in the table.

Kuva 1 Table 1: Cryptographic algorithms for JWT protection in the FTN (RFC 7518)

Header	Usage	Value	Algorithm	Status in FTN
alg	JWS	RS256	RSASSA-PKCS1-v1_5 using SHA-256	REQUIRED
alg	JWS	PS256	RSASSA-PSS using SHA-256 and MGF1 with SHA-256	OPTIONAL
alg	JWS	ES256	ECDSA using P-256 and SHA-256	OPTIONAL
alg	JWE	RSA-OAEP	RSAES OAEP using default parameters	REQUIRED
alg	JWE	RSA-OAEP-256	RSAES OAEP using SHA-256 and MGF1 with SHA-256	OPTIONAL
alg	JWE	ECDH-ES	Elliptic Curve Diffie-Hellman Ephemeral Static key agreement using Concat KDF	OPTIONAL
enc	JWE	A128GCM	AES GCM using 128-bit key	REQUIRED

RSA keys used MUST be 2048 bits or longer. Elliptic curve keys MUST be long enough to provide a symmetric key equivalent security strength of at least 112 bits (typical EC algorithm key length of 224 bits or longer). Symmetric keys MUST be 128 bits or longer. Hash algorithms used MUST have a digest size of 224 bits or longer.

Algorithms/key sizes that provide cryptographically equivalent or stronger security level than described here MAY be used. Weaker algorithms/key sizes MUST NOT be used.

2.2.3 Signatures

OIDC ID Tokens, Token Request client assertions, and User Info tokens (if used) in the FTN MUST be digitally signed JWTs and must be validated by the receiver using the pre-exchanged signature validation keys. OIDC Authentication Requests MAY also be signed. Thus an asymmetric signing key is REQUIRED for both FTN IdP and FTN Broker.

2.2.4 Encryption

Encryption of OIDC Tokens is mandatory. The ID Token and User Info tokens (if used) from the FTN IdP MUST be first signed by one of the IdP's signing private keys and then encrypted using the FTN Broker's public encryption key. In other words, the Token is first secured with JWS and then with JWE to create a nested JWT as described in the OIDC Core specification section 10. For this to be possible the FTN Broker needs to have an asymmetric encryption key.

2.3 TLS Requirements

TLS MUST be used on the transport layer to protect all HTTP traffic related to OIDC, i.e. all URLs requested by the user browser and OIDC backend server-to-server requests MUST begin with `https://`. An exception to this requirement is CRL and/or OCSP traffic. The TLS server X.509 certificates that are used to protect communication with client web browsers/apps MUST be generally trusted by browsers/OS's (95+%). These certificates MUST be valid based on all commonly implemented validators in web browsers (certificate path to trusted root provided, certificates not expired, strong enough cryptographic primitives used, etc). The use of extended validation (EV) certificates is RECOMMENDED. eIDAS-notified IdPs may also be subject to additional security requirements based on the eIDAS regulations.

X.509 certificates used for protecting OIDC backend server-to-server traffic MUST be issued by a generally trusted CA, or be explicitly configured/pinned. Use of client certificates in backend server-to-server TLS connections is OPTIONAL.

All TLS servers in the FTN MUST support TLS version 1.2 (RFC 5246) or higher. See FICORA Regulation 72 notes for more details on the requirements and sample cipher suites that fulfil the requirements. When using TLS 1.3, 0-RTT data MUST NOT be used. It is RECOMMENDED to use the HTTP Strict Transport Security header (RFC 6797) in TLS servers.

3 OpenID Connect profile

3.1 Attribute and claim requirements

Built-in OIDC claims (email, phone, etc) are not used in the FTN, unless they are explicitly referred to in this document.

All claim values MUST be encoded in UTF-8 character set and SHOULD be in Latin script. Precomposed Unicode characters SHOULD be used when possible, instead of decomposed characters. If transliteration is required from non-Latin scripts, the currently used standard of the Finnish Population Registry SHOULD be used.

3.1.1 Natural person attributes

Note that the eIDAS technical specifications are not directly applicable within the FTN. eIDAS technical specifications should be used within the context of this document only when explicitly referred to.

3.1.1.1 Required attributes

Claim Name	**FriendlyName** (not used in OIDC)	**eIDAS MDS Attribute**	**Comments, Example value(s) in `Courier New`**
urn:oid: 2.5.4.4	`FamilyName`	Current Family Name	`Meikäläinen` `von Essen`

urn:oid:
1.2.246.575.1.14
	`FirstNames`	Current First Names	`Matti Elmeri Valdemar`
			`Anna-Liisa Hilkka`
			(all known current first/given names, space separated)

urn:oid: 1.3.6.1.5.5.7.9.1	`DateOfBirth`	Date of Birth	`1971-06-28` (YYYY-MM-DD)

urn:oid:
1.2.246.21
	`HETU`	-	`220750-999Y`
			`141002A909X`
			(Finnish personal identity code, henkilötunnus) *

urn:oid: 1.2.246.22	`SATU`	-	`99999999D` (Finnish Unique Identification Number, sähköinen asiointi-tunnus) *
http://eidas.europa.eu/attributes/naturalperson/PersonIdentifier	`PersonIdentifier`	Unique Identifier	`XX/YY/123456ABCDEF` (as defined by eIDAS SAML Attribute Profile [eIDASTech], subject to change) *

* One of these three claims is mandatory, the rest are OPTIONAL to include in a claims token. It is up to the FTN Broker and IdP to agree which identifier between them is used as the mandatory claim. The eIDAS PersonIdentifier is not expected to be commonly used nationally within the FTN, but is referred to here in case eIDAS cross-border authentication becomes relevant to the FTN.

3.1.1.2 Optional attributes

If the ID Token received by an FTN Broker contains unrecognized optional attributes, the token SHOULD NOT be discarded solely due to the token containing unsupported attributes, as long

as the required attributes specified above are provided. This is suggested to futureproof implementation behavior, so new attributes can be added in later versions of this recommendation in a backwards compatible manner.

Claim Name	**FriendlyName** (not used in OIDC)	**eIDAS MDS Attribute**	**Comments, Example value(s) in `Courier New`**
urn:oid: 1.2.246.575.1.3	`FamilyBirthName`	Family Name at Birth	`Möttönen` `von Essen`

urn:oid:
1.2.246.575.1.4
	`FirstBirthName`	First Names at Birth	`Matias Jalmari Valdemar`
			`Anna-Liisa Hilkka`
			(all known first/given names at birth, space separated)

urn:oid:
1.3.6.1.5.5.7.9.2
	`PlaceOfBirth`	Place of Birth	`Helsinki`
			`Kittilä Finland`
			(typically city and/or country. No specific separator is defined, but more specific areas should precede less specific areas)

urn:oid: 1.2.246.575.1.16	`CurrentAddress`	Current Address	(see OIDC Core 1.0 section 5.1.1 address claim and the example below)

urn:oid:
1.2.246.575.1.15
	`Gender`	Gender	`Male`
			`Female`
			`Not Specified`
			(string with restriction of selection to one of the three options specified above)

urn:oid:
2.5.4.42
	`GivenName`	-	`Elmeri`
			`Anna-Liisa`
			(kutsumanimi in Finnish, one of the current registered first names normally used by the person)

urn:oid:
1.2.246.575.1.18
	`AuthCachingDisabled`	-	`true`
			`false`
			(boolean (lower case), indication from the IdP/user that forbids caching of the current user authentication session (SSO), if set to true. If the attribute is not sent, the default value is false)

Address claim example:

```
{
  "urn:oid:1.2.246.575.1.16": {
    "street_address": "Itämerenkatu 3 A 75",
    "locality": "Helsinki",
    "postal_code": "00180",
    "country": "FI" }
}
```

3.1.2 Legal person attributes

Note that including attributes of a legal person in claims in the FTN does not in itself convey authority for the natural person being authenticated to enter into binding contracts on behalf of the legal person.

3.1.2.1 Required attributes

Note that the eIDAS technical specifications are not directly applicable within the FTN. eIDAS technical specifications should be used within the context of this document only when explicitly referred to.

Legal person claims MUST always be accompanied by the mandatory natural person claims of the natural person being authenticated. Optional natural person claims MAY also be included.

Claim Name	**FriendlyName** (not used in OIDC)	**eIDAS MDS Attribute**	**Comments, Example value(s) in `Courier New`**
urn:oid: 2.5.4.10	`LegalName`	Current Legal Name	`Widget Factory Oy`
http://eidas.europa.eu/attributes/legalperson/LegalPersonIdentifier	`LegalPersonIdentifier`	Uniqueness Identifier	`XX/YY/123456ABCDEF` (as defined by eIDAS SAML Attribute Profile [eIDASTech], subject to change) *
urn:oid: 1.2.246.575.1.7	`VATRegistration`	VAT Registration Number	`FI98765432` (company identifier (Y-tunnus) in EU format) *

* One of these two claims is mandatory, the other is OPTIONAL to include in a claims token. It is up to the FTN Broker and IdP to agree which identifier between them is used as the mandatory claim. The eIDAS LegalPersonIdentifier is not expected to be commonly used nationally within the FTN, but is referred to here in case eIDAS cross-border authentication becomes relevant to the FTN.

3.1.2.2 Optional attributes

Claim Name	**FriendlyName** (not used in OIDC)	**eIDAS MDS Attribute**	**Comments, Example value(s) in `Courier New`**
urn:oid: 1.2.246.575.1.6	`LegalAddress`	Current Address	(see OIDC Core 1.0 section 5.1.1 address claim and the example for natural person address above)
urn:oid: 1.2.246.575.1.8	`TaxReference`	Tax Reference Number	
urn:oid: 1.2.246.575.1.9	`BusinessCodes`	Directive 2012/17/EU Identifier	
urn:oid: 1.2.246.575.1.10	`LEI`	Legal Entity Identifier (LEI)	
urn:oid: 1.2.246.575.1.11	`EORI`	Economic Operator Registration and Identification (EORI)	
urn:oid: 1.2.246.575.1.12	`SEED`	System for Exchange of Excise Data (SEED)	
urn:oid: 1.2.246.575.1.13	`SIC`	Standard Industrial Classification (SIC)	

3.2 Levels of Assurance (LoA) / ACR

Authentication requests in the FTN MUST identify the requested Level of Assurance (LoA) for end-user authentication using the `acr_values` (Authentication Context Class Reference) parameter. Multiple values may be provided in the Authentication request, space separated, in the order of preference, as described in OIDC Core 1.0 specification section 3.1.2.1. The ID token response MUST specify the actual `acr` value used to perform authentication. If the IdP can't fulfil any of the requested Authentication Context Class References, it MUST return an error. The FTN Broker MUST verify that the ID token contains an acceptable `acr` value.

The ACR values relevant within the FTN are the following URIs:

ACR value	Meaning
`http://ftn.ficora.fi/2017/loa2`	Finnish level substantial (korotettu)
`http://ftn.ficora.fi/2017/loa3`	Finnish level high (korkea)
`http://eidas.europa.eu/LoA/low`	eIDAS level low
`http://eidas.europa.eu/LoA/substantial`	eIDAS level substantial
`http://eidas.europa.eu/LoA/high`	eIDAS level high

Use of an ACR value that contains the string `eidas.europa.eu` signals that the authentication method requested/used has been successfully notified on the given level or higher to the European Commission for cross-border authentication within the EU/EEA. ACR values containing the string `ftn.ficora.fi` signal that the authentication method has been approved to be a part of the FTN on the given level or higher. The eIDAS level low SHOULD NOT be used within the FTN, it is included in the table above for completeness.

In the Finnish Trust Network only levels substantial and high are used. An eIDAS level ACR meets the requirements of the corresponding Finnish level, but not vice versa. If an FTN IdP receives a request from an FTN Broker for level substantial authentication (only one ACR value requested), the request can also be fulfilled with a level high device/mechanism. In this case the response ACR value MUST correspond to the requested level, i.e. substantial.

For **test and/or demo purposes**, the following ACR values MUST be used. Authentication transactions done using these values MUST NOT be relied on for any purpose, they are only meant for testing. Production data or services MUST NOT be used for testing or demonstration purposes. The test data used MUST NOT use real life data (personal data).

ACR value **for test purposes**	Meaning
`http://ftn.ficora.fi/2017/loatest2`	Finnish test level substantial (korotettu)
`http://ftn.ficora.fi/2017/loatest3`	Finnish test level high (korkea)

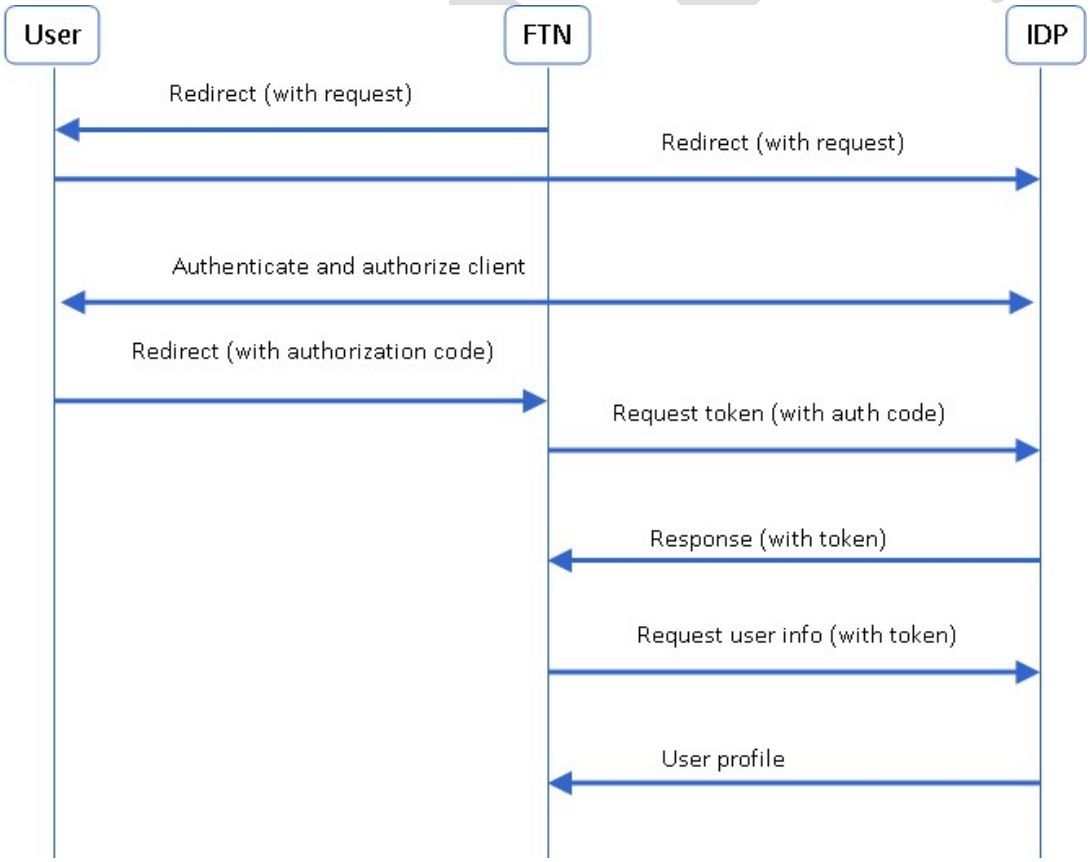
4 OpenID Connect protocol exchanges

4.1 Protocol flow overview

Authentication requests define the properties and conditions that must be met in order to authenticate an end user. Details on the Authentication Request Protocol can be found in "OpenID Connect Basic Client Implementer's Guide" (OpenID Basic Client Implementation Guide, 2014), Section 2.1.1. FTN uses the *Authorization Code Flow* variant of OIDC.

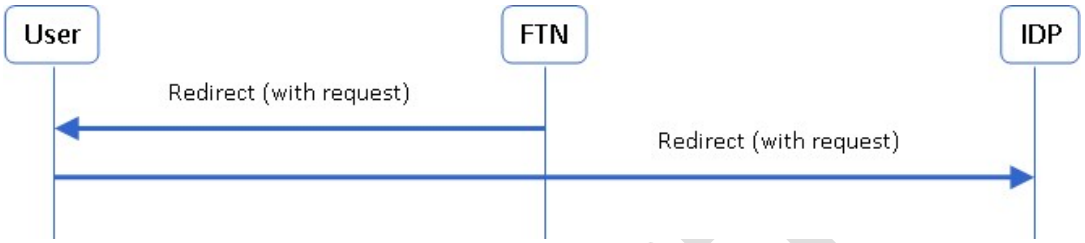
The following signalling process illustrates the phases that follow when the user has selected to perform authentication and authorisation through an OpenID Connect Identity Provider. "FTN" in the image represents an FTN Broker. The use of user info request/response messages is OPTIONAL within the FTN.

The whole OIDC protocol exchange MUST be completed within 10 minutes from the first message. Both FTN parties SHOULD monitor the time independently and either party may abort the exchange if the time limit is exceeded.



4.2 Authentication request

In the first phase of the FTN authentication, an authentication request is sent from an FTN Broker to the IdP, typically using a redirect via user's web browser. The request MAY be sent either as a GET or POST http request. The OIDC authentication and authorisation process is the following:



The authentication request sent to the FTN IdP contains the following parameters, described in more detail in OIDC Core 1.0 section 3.1.2.1:

Parameter	(Example) Value	Comments

`scope`	`openid`	This value MUST include `openid` to indicate the use of the OpenID protocol.
		
		In addition to requesting specific single claims by their OID, one or more of the following custom scopes MAY be used for requesting eIDAS minimum person data sets described in section 3.1.1.1 (FriendlyName column):
		`ftn_hetu` includes `HETU`
		`ftn_satu` includes `SATU`
		`ftn_personidentifier` includes `PersonIdentifier`
		All three also include claims `FamilyName`, `FirstNames`, and `DateOfBirth`.
		
		Example: when "`openid ftn_hetu`" is specified as the request scope, the response SHOULD include claims `FamilyName`, `FirstNames`, `DateOfBirth`, and `HETU`, using the corresponding OIDs for the claim names.

`response_type`	`code`	REQUIRED
`client_id`	varies	REQUIRED. Identifier of the party initiating the authentication (FTN Broker), assigned to the FTN Broker by the FTN IdP and

Parameter	(Example) Value	Comments
		RECOMMENDED to consist of an alphanumeric string (A-Z, a-z, 0-9).
`redirect_uri`	`https://...`	REQUIRED, URI for returning the authentication response to. MUST match what is configured at the IdP for the corresponding `client_id`
`state`	varies	REQUIRED and MUST contain at least 128 bits of entropy (for example at least 22 random characters A-Z, a-z, 0-9)
`nonce`	varies	REQUIRED and MUST contain at least 128 bits of entropy (for example at least 22 random characters A-Z, a-z, 0-9)
`acr_values`	list of URIs	REQUIRED, space separated list of requested FTN authentication context class reference values
`ui_locales`	list of language tags	REQUIRED, end user language preference tags (BCP 47) coding according to https://en.wikipedia.org/wiki/List_of_**ISO_639-1**_two_letter_codes. If the preference tag is not set, the default will be "fi"
`prompt`	`login`	RECOMMENDED, the FTN Broker SHOULD use this parameter to make sure the end user is reauthenticated in case the end user already had an existing session at the IdP

FTN specific OIDC authentication request parameters are specified below:

Parameter	(Example) Values	Comments
`ftn_spname`	`Esimerkkikauppa Oy`	REQUIRED, Human readable name of the Service Provider the user is authenticating to. This is carried in the request so that the IdP or Broker MUST show in its user interface the end user service being authenticated to. The name is RECOMMENDED to be in the same language as user's preferred user interface language (parameter `ui_locales`).
`ftn_sptype`	`private`	OPTIONAL, Type of the Service Provider the user is authenticating to. This string can be either: • `public` — services provided by the public sector • `private` — services provided by private organizations/individuals

`ftn_idp_id`	`fi-xyz-abc`
	`fi-tuvwxyz-1234`
	`fi-x1yz-a2bcd5`
	`fi-xyz`
		OPTIONAL, lower case alphanumeric (a-z, 0-9, plus '-' acting as separator) ASCII identifier for the FTN IdP that SHOULD be used for end-user authentication. This parameter is primarily meant to be used by Service Providers to signal their Broker the IdP to use for authenticating the

Parameter	(Example) Values	Comments
		user, if the user has indicated the IdP service to use to the Service Provider. This makes it possible for the Broker to seamlessly transfer the user to the chosen FTN IdP, without the Broker having to display an IdP selection user interface to the end user. The first "`fi`" part is constant within the FTN. The second "`xyz`" part in the example value is allocated by FICORA and listed in the registry of strong identification service providers[iv]. The third part is OPTIONAL and MAY be used by the IdP to differentiate between their authentication services/methods. The third part value is allocated by the corresponding FTN IdP. It MUST consist of lower case alphanumeric characters only (a-z, 0-9). Maximum length of each part is 20 characters, which means the maximum length of the whole identifier including two separators is 62 characters.

Other parameters defined by OIDC Core 1.0 for authentication requests or agreed by the parties MAY also be used. For example the parameter `max_age` may be useful in SSO use cases and PKCE (RFC 7636) also uses additional request parameters.

If the IdP determines the request to be valid following OIDC Core 1.0 section 3.1.2.2 validation rules and the table above, the IdP starts the end-user authentication process. The exact process used for end user authentication is out of scope for this document.

4.2.1 Chained authentication tokens/means creation request

If this recommendation is used as the interface for creating new chained strong authentication tokens/means between two FTN IdPs, the following extra parameter is defined ("Laki vahvasta sähköisestä tunnistamisesta ja sähköisistä luottamuspalveluista" 617/2009 17 §). This parameter MUST NOT be used for any other purpose.

Parameter	(Example) Values	Comments
`ftn_chain_level`	one URI, see Section 3.2	REQUIRED, when new chained authentication tokens/means are created. Presence of this parameter in an authentication request to an FTN IdP from another FTN IdP signals that this authentication is being done to issue a new chained strong authentication token/means to an end user.

Parameter	(Example) Values	Comments
		The value of this parameter MUST be a URI describing the Level of Assurance (LoA) of the authentication token/means being issued. It MUST be of the same LoA level or lower than is used by the user to perform the authentication. LoA URIs that begin with `http://ftn.ficora.fi/` MUST be used exclusively. An authentication request using this parameter MUST always result in full (re)authentication, i.e. the `prompt` authentication request parameter MUST be set to value `login`. The request MUST be cryptographically signed according to Section 4.2.2 and SHOULD NOT be relayed via 3rd party FTN Brokers. If a Broker is used, the Broker MUST meet the requirements of the requested LoA. See Section 4.5.3 for the corresponding ID Token response attribute.

4.2.2 Authentication request signing

OIDC authentication requests themselves are not cryptographically authenticated by default. It is RECOMMENDED to use signing of authentication requests as described in OIDC Core 1.0 section 6.1 (`request` parameter). FTN IdP and FTN Broker may agree to use fully signed requests. In such case all parameters using OAuth 2.0 request syntax other than the mandatory ones SHOULD be ignored. Fully signed request MUST have parameters client_id, response_type and scope presented in OAuth 2.0 request syntax and their processing MUST follow OIDC Core 1.0 section 6.1

In any case the FTN IdP authenticates the FTN Broker later, when a request for an ID token is made. It is RECOMMENDED that billing etc accounting is only based on cryptographically signed ID Token requests/responses.

4.3 Authentication response

If the end-user authentication completes successfully, the FTN IdP sends an authentication response to the FTN Broker according to OIDC Core 1.0 section 3.1.2.5 via redirect.

A successful response MUST include the following parameters:

Parameter	Comments
`code`	REQUIRED and MUST contain at least 128 bits of entropy (for example at least 22 random characters A-Z, a-z, 0-9)
`state`	REQUIRED, copied from the authentication request as is

A successful authentication response MUST be validated according to OIDC Core 1.0 section 3.1.2.7.

If the end-user authentication fails or some other cause leads to an error condition at the IdP, the error SHOULD be communicated back to the FTN Broker according to OIDC Core 1.0 section 3.1.2.6. The corresponding `state` parameter SHOULD, if possible, be included in such a response, and a `code` parameter MUST NOT be included.

4.3.1 Authentication error response

In addition to the authentication error response defined in the OIDC Core 1.0 section 3.1.2.6 FTN defined error response parameters MUST be used.

error	**error_description (example)**	**Comments**
`invalid_request_object`	`"missing request object"`	REQUIRED. Use when request object is required
`access_denied`	`"User cancel at IDP"` `"User cancel at broker"`	REQUIRED. User cancel – descibe where cancel occurs

4.4 ID Token request

To complete the *authorization code flow*, the `code` received in the previous phase of the protocol needs to be exchanged for an ID token. The ID token contains the actual identity information of the person authenticating. An access token MUST also be returned in addition to the ID Token due to OAuth 2.0 requirements, even if an access token is not used later. A refresh token SHOULD NOT be returned.

The ID token request is made directly from the FTN Broker to the FTN IdP as an HTTP POST request to the token endpoint, without going through the end user's browser/device. The following table details the parameters of the ID token request (OIDC Core 1.0 sections 3.1.3.1 and 9). The FTN Broker identifies itself using the private_key_jwt authentication method.

Parameter	**(Example) Values**	**Comments**
`grant_type`	`authorization_code`	REQUIRED, fixed value
`code`	varies	REQUIRED, the `code` from authentication response (section 4.3)
`client_id`	varies	REQUIRED, see section 4.2
`redirect_uri`	`https://...`	REQUIRED, see section 4.2
`client_assertion_type`	`urn:ietf:params:oauth:client-assertion-type:jwt-bearer`	REQUIRED, signals that the assertion is a signed JWT

Parameter	(Example) Values	Comments
client_assertion	JWT contents, MUST include claims iss, sub, aud, jti, and exp	REQUIRED, signed JWT (see also RFC 7523)

The FTN IdP SHOULD keep a record of jti claim values it has seen in client assertions that have not yet expired (exp claim value) in order to prevent assertion replay. The exp value MUST NOT be set more than 10 minutes into the future by the ID Token request sender and MUST be validated by the receiver. Also iss, sub, and aud values of the client assertion MUST be validated.

The IdP MUST validate the request according to OIDC Core 1.0 section 3.1.3.2. Specifically replays of the same authorization code MUST be prevented.

4.5 ID Token response

The structure of a successful ID token response is detailed in OIDC Core 1.0 section 3.1.3.3. Before returning the ID token to the FTN Broker, the FTN IdP MUST first sign the token with the private key of the FTN IdP. After signing it MUST be encrypted with the public key of the FTN Broker to create a nested JWT.

Parameters	(Example) Values	Comments
access_token	varies	REQUIRED, MUST contain at least 128 bits of entropy (for example at least 22 random characters A-Z, a-z, 0-9). access_token is not necessarily used in the typical FTN use case, but it is required by OAuth 2.0.
token_type	Bearer	REQUIRED
expires_in	number of seconds	OPTIONAL, number of seconds access_token is valid for
id_token		REQUIRED, first signed and then encrypted JWT containing information about the end user that authenticated

The format of an error response to an ID token request is described in OIDC Core 1.0 section 3.1.3.4.

4.5.1 Token error response

In addition to the token error response defined in the OIDC Core 1.0 section 3.1.2.6 FTN defined error response parameters MUST be used.

error	**error_description (example)**	**Comments**
`invalid_request`	`"invalid 'aud' parameter"`	REQUIRED. Use when "aud", "exp" or "jti" claims in JWT payload have a problem, and describe the erroneous claim.
`invalid_client`	`Leave empty to prevent fishing of invalid/valid client_ids.`	REQUIRED. Use when token request "iss" claim is not recognized, or when JWT signature validation fails on server side.

4.5.2 ID Token

The ID Token is a signed and encrypted JWT that contains details about the authentication event. The FTN Broker MUST validate the ID Token in the token response as described in OIDC Core 1.0 section 3.1.3.5. The fields in the following table are further described in OIDC Core 1.0 section 2:

Claim	**Comments**
`iss`	REQUIRED. Case sensitive issuer identifier of the FTN IdP, typically the https URL of the IdP service.
`sub`	REQUIRED (by the OIDC specification). Subject Identifier. This identifier is meant to be transient in the FTN and MUST NOT be relied upon when the user authenticates the next time. The user is identified based on the attributes defined in section 3.1.1.1.
`aud`	REQUIRED. Audience(s) this ID Token is intended for. It MUST contain the client_id of the FTN Broker as an audience value.
`exp`	REQUIRED. Expiration time on or after which the ID Token MUST NOT be accepted for processing. The processing of this parameter requires that the current date/time MUST be before the expiration date/time listed in the value. The expiration time MUST be 10 minutes or less into the future from the timestamp specified in `iat`. All FTN participants SHOULD be configured with a reliable UTC time source. Its value is a JSON number representing the number of seconds from 1970-01-01T00:00:00Z as measured in UTC until the date/time. See RFC 3339 [RFC3339] for details regarding date/times in general and UTC in particular.
`iat`	REQUIRED. Time at which the JWT was issued, number of seconds since the beginning of 1970 UTC.

`auth_time`	REQUIRED. Time when the end-user authentication occurred, number of seconds since the beginning of 1970 UTC.
`nonce`	REQUIRED. Case sensitive string from the authentication request to associate an end-user with an ID token and to mitigate replay attacks (see section 4.2). The FTN Broker MUST verify that the nonce claim value is equal to the value of the nonce parameter sent in the authentication request.
`acr`	REQUIRED. The Authentication Context Class Reference string for this authentication transaction, see section 3.2.
`at_hash`	OPTIONAL. Recommended to be included and validated when an `access_token` is also returned.

The ID token returned by the FTN IdP MUST also always contain the requested/agreed minimum personal attribute set described in section 3.1.1.1.

4.5.3 Chained authentication tokens/means response claim

FTN IdP to FTN IdP specific ID Token claim for creating a new chained strong authentication token/means is specified below ("Laki vahvasta sähköisestä tunnistamisesta ja sähköisistä luottamuspalveluista" 617/2009 17 §):

Claim Name	**(Example) Values**	**Comments**
urn:oid: 1.2.246.575.1.17	one URI, see Section 3.2	REQUIRED, when successfully issuing an ID Token based on a Section 4.2.1 request. The presence of this claim in an ID Token signals that the FTN IdP sending the response performed the authentication successfully in response to an `ftn_chain_level` authentication request parameter, and that the IdP receiving the ID Token MAY thus issue a new chained authentication token/means of the specified Level of Assurance to the end user. This claim MUST NOT be included in an ID Token if the authentication request did not include the `ftn_chain_level` parameter. The value of this parameter MUST be the same Level of Assurance (LoA) URI as specified in the request and the authentication of the end user MUST have been performed at the indicated LoA or higher.

4.6 User Info Endpoint

If user info endpoint (OIDC Core 1.0 section 5.3) is used in the FTN, the responses SHOULD be processed similarly to the ID token endpoint. I.e. all responses must first be signed and then encrypted to form a signed-and-encrypted JWT. An access token MUST be returned in the ID token response to be able to access the user info endpoint.

5 Other considerations

5.1 Public clients

This profile has been written with "confidential clients" in mind, i.e. protocol exchanges happening between trusted servers that can be trusted with secrets (private keys). If a public client is used, for example any application that runs on the end-user's phone or computer (apps, browsers), the security of the system MUST NOT rely on the client being able to keep pre-bundled secrets like private keys. The client should either work without secrets, or the secrets should be client-specific and thus configured/created/enrolled at application installation/runtime and be stored diligently.

In environments like smartphone clients the *authorization code flow* needs additional controls to be secure, for example PKCE (RFC 7636[v]) SHOULD be implemented. RFC 8252[vi] also lists things to take into account when dealing with native applications in OAuth environments.

5.2 OIDC provider metadata

It is RECOMMENDED for OIDC providers to publish metadata of their services according to the OIDC Connect Discovery 1.0 specification[vii].

Notes

[i] OpenID Connect Core 1.0 incorporating errata set 1 http://openid.net/specs/openid-connect-core-1_0.html

[ii] RFC 6819 OAuth 2.0 Threat Model and Security Considerations https://tools.ietf.org/html/rfc6819

[iii] FICORA Regulation 72 on electronic identification and trust services https://www.viestintavirasto.fi/en/steeringandsupervision/actsregulationsdecisions/regulations/regulation72onelectronicidentificationandtrustservices.html

[iv] FICORA Register of strong identification service providers https://www.viestintavirasto.fi/kyberturvallisuus/sahkoinentunnistaminenjaallekirjoitus/rekisteritunnistamispalveluntarjoajista.html

[v] RFC 7636 Proof Key for Code Exchange by OAuth Public Clients https://tools.ietf.org/html/rfc7636

[vi] RFC 8252 OAuth 2.0 for Native Apps https://tools.ietf.org/html/rfc8252

[vii] OpenID Connect Discovery 1.0 incorporating errata set 1 https://openid.net/specs/openid-connect-discovery-1_0.html