Public Key management with Signed JWKS

# Introduction

This specification outlines how entities within the *Finnish Trust Network* (FTN) publish public keys used for signing and encrypting *OpenID Connect* protocol messages.

The purpose is to provide an additional layer of security and integrity protection for management of public keys.

This is a profile of *OpenID Connect Federation* specification focusing on the entity statement and signed jwks features.

# Entity Statement

<https://openid.net/specs/openid-connect-federation-1_0.html#section-3.1>

An Entity Statement contains the information needed for the Entity that is the subject of the Entity Statement to participate in federation(s). An Entity Statement is always a signed JWT. The subject of the JWT is the Entity itself. The issuer of the JWT is the party that issued the Entity Statement, which will often be the Entity itself.

Mandatory parameters

* iss
  + REQUIRED. The Entity Identifier of the issuer of the statement.
* sub
  + REQUIRED. The Entity Identifier of the subject.
  + FTN: Subject MUST be same as issuer.
* iat
  + REQUIRED. The time the statement was issued. Its value is a JSON number representing the number of seconds from 1970-01-01T0:0:0Z as measured in UTC until the date/time.
* exp
  + REQUIRED. Expiration time on or after which the statement MUST NOT be accepted for processing. Its value is a JSON number representing the number of seconds from 1970-01-01T0:0:0Z as measured in UTC until the date/time.
* jwks
  + REQUIRED. A JSON Web Key Set (JWKS) representing the public part of the subject Entity's signing keys. The corresponding private key is used by leaf entities to sign Entity Statements about themselves, and intermediate entities to sign statements about other entities. The keys that can be found here are intended to sign Entity Statements and SHOULD NOT be used in other protocols.
* metadata
  + REQUIRED Conditional. JSON object including protocol specific metadata claims that represent the Entity's metadata.
  + FTN: Possible metadata type identifiers: openid\_provider and openid\_relying\_party

# Signed JWKS

<https://openid.net/specs/openid-connect-federation-1_0.html#section-4.1>

Signed JWKS is a signed JWT having the entity's JWK Set as payload. The JWT is signed with a key that was included in the JWK set that the entity published in its (self-signed) entity statement.

Mandatory parameters

* iss
  + REQUIRED. The "iss" (issuer) claim identifies the principal that issued the JWT.
* sub
  + REQUIRED. This claim identifies the owner of the keys.
  + FTN: Subject MUST be same as issuer.
* keys
  + REQUIRED. List of JWKs.

## signed\_jwks\_uri

A URI pointing to a signed JWT having the entity's JWK Set as payload. This parameter MUST exist in metadata embedded in Entity Statement.

# Processing

## Publish entity statement

1. Generate current and next key pair
2. Build JWK set from current and next key pair
3. Build entity statement JWT (iss, sub, iat, exp, jwks and metadata claims)
4. Sign JWT with current key

## Publish protocol public keys

1. Fetch protocol JWK set
2. Build signed jwks JWT (iss, sub and keys claims)
3. Sign JWT with current key

## Validate entity statement

1. Decode entity statement JWT
2. Make sure iss and sub are same and match known entity
3. Read exp claim and validate statement has not expired
4. Read JWK set from entity statement and validate signature of entity statement JWT

## Consume protocol public keys

1. Validate entity statement
2. Read signed\_jwks\_uri from entity statement metadata and fetch signed jwks JWT
3. Read JWK set from entity statement and validate signature of signed jwks JWT
4. Make sure iss and sub are same and match entity statement values

# Notes

## Federation Operator

In Finnish Trust Network there is no Federation Operator or trusted third party. All entities are acting as Leaf Entities. Each entity will publish a self-signed Entity Statement.

## Lifecycle of keys

The self-signed entity statement and signed jwks are signed with keys that have fairly long expiration (years or several months). This long expiration also allows out-of-band sharing of the entity statement.

The keys within signed jwks are used for signing and encrypting OIDC protocol messages. These keys have shorter expiration (months or weeks).

# Examples

## Entity Statement

### JWT

eyJhbGciOiAiUlMyNTYiLCAia2lkIjogImVudGl0eS1zdGF0ZW1lbnQta2V5LTEiLCAidH  
lwIjogIkpXVCJ9.ew0KICAiZXhwIjogMTcyNDM3MTIwMCwNCiAgInN1YiI6ICJodHRwczo  
vL2V4YW1wbGUuY29tIiwNCiAgImlzcyI6ICJodHRwczovL2V4YW1wbGUuY29tIiwNCiAgI  
m1ldGFkYXRhIjogew0KICAgICJvcGVuaWRfcHJvdmlkZXIiOiB7DQogICAgICAic2lnbmV  
kX2p3a3NfdXJpIjogImh0dHBzOi8vaWRwLmV4YW1wbGUuY29tL3NpZ25lZC1qd2tzIiwNC  
iAgICAgICJpc3N1ZXIiOiAiaHR0cHM6Ly9pZHAuZXhhbXBsZS5jb20iDQogICAgfQ0KICB  
9LA0KICAiandrcyI6IHsNCiAgICAia2V5cyI6IFsNCiAgICAgIHsNCiAgICAgICAgImUiO  
iAiQVFBQiIsDQogICAgICAgICJraWQiOiAiZW50aXR5LXN0YXRlbWVudC1rZXktMSIsDQo  
gICAgICAgICJrdHkiOiAiUlNBIiwNCiAgICAgICAgIm4iOiAidmwtWnNBdzVpVFFPOGIyZ  
kJPZ1dMZGlKYjlLcWtsalR0c29UaEhRaUk4QzdGQWZwaENwSmVZUHNIczhxeGpLNDFvWnR  
wNGwzdjZvLURnMVRKbkpxLUNFR2F3eElzazBBYVhTQ2ZpOEtVOHJQZXpESS15a3FLT1QxY  
VEwX29MTW9XZmlEMHM3UFNvMHJuUXQ0S3hsY1Q0S3dZSERudWMzczY0WWx6aFdpc0JFb0Z  
NeVQ1RU1kRS1XR0ttRFhhaGVJa1dseHVMbjkzVEFSQVdwTjNVVTh2dmY4akgtekROa0VmT  
zRlbnJnWFMwRDdzNTZ5dWp6MXVIVUN1S1VXb3NmbGdrLW52YWMzV0djVGpNOXI4RU9GX05  
sZUFNMm1haGRhY001UGQwWFo5dmZlWVVXakxsSGFFMmVTbDluOFhVX19peDFLZmxiNEFac  
E1LT25ic0VuSTl3IiwNCiAgICAgICAgInVzZSI6ICJzaWciDQogICAgICB9LA0KICAgICA  
gew0KICAgICAgICAiZSI6ICJBUUFCIiwNCiAgICAgICAgImtpZCI6ICJlbnRpdHktc3Rhd  
GVtZW50LWtleS0yIiwNCiAgICAgICAgImt0eSI6ICJSU0EiLA0KICAgICAgICAibiI6ICJ  
tamhoTEdhNVFUSHI0RXdGcXFZbmFhMm1mbkJ5YURzVXVlRlk2dFFvSEJ3OXBhUkdWM3ZzN  
1JjTHY5RHNPZXFlaERCSkZlZFF5ek9iQ0xFT0pacnk3WjRfM2VoWGNjZl9KMlpMU3dFTGh  
CcDVpTWxCZkZHQ1kxTmNHemUyY21wYWVNS3FrdUc2TTdMM3k1RzJVZGx5dHlYQmI1bGFiT  
k5BRUNUbVhrb01Jai1qNEg1b2V4OFpJMVBhRFJGTG5hS2xCY2VxTmViZ0NqYW43U2tTa0d  
ScHZuSWZqaVo3N1lDc2Y5Z0VHQVBKRjZwZDNTSGF5UWJtb3hVQXlNNU9qdUN0dmhBeVhKb  
kl2RzdJblVlSThReGx3Yjh2anhVajZ2X0JGVTJ3QzVtQnZvbVdVVlRMWmhYV3VXT2VTMkp  
jdHBwQkNIVURZcWdLc3lqSHpoZVRoYmVIalEiLA0KICAgICAgICAidXNlIjogInNpZyINC  
iAgICAgIH0NCiAgICBdDQogIH0sDQogICJpYXQiOiAxNjYxMjEyODAwDQp9.vEW02VDNpO  
NTc-Nk8fXzPHG-giPqjyZzLrSu2SWouDVZhrB0dUsuzpSsJRssutD6\_8\_aeIpAaN\_NgG\_h  
jLOqfRFrX5NrHNwcMqIrIij2Lr-TIcz4ViAud6tR7JQs2I\_1xkNaXOBgE39d29nD31us9H  
khqTmL-hFBylI1ibXmYDxaDRQfOHE29fLkAF69fNkStMF3NQszm5yp9ZuFowj6xtDSc3NQ  
JELb2\_PPvwN90gCcfTLeJyGY5HTQi8OFUR5qBuGCZvc9MFgfqC-dI-bG1gowppbOa\_OP5G  
ROl\_i6DeG9TL2vhMajOErY7ru\_j4EGtg2CnYOPkKc0w6q-KZk82w

### JWT header

{"alg": "RS256", "kid": "entity-statement-key-1", "typ": "JWT"}

### JWT body

{  
 "exp": 1724371200,  
 "sub": "https://example.com",  
 "iss": "https://example.com",  
 "metadata": {  
 "openid\_provider": {  
 "signed\_jwks\_uri": "https://idp.example.com/signed-jwks",  
 "issuer": "https://idp.example.com"  
 }  
 },  
 "jwks": {  
 "keys": [  
 {  
 "e": "AQAB",  
 "kid": "entity-statement-key-1",  
 "kty": "RSA",  
 "n": "vl-ZsAw5iTQO8b2fBOgWLdiJb9KqkljTtsoThHQiI8C7FAfphCpJeYPsHs8qxjK41oZtp4l3v6o-Dg1TJnJq-CEGawxIsk0AaXSCfi8KU8rPezDI-ykqKOT1aQ0\_oLMoWfiD0s7PSo0rnQt4KxlcT4KwYHDnuc3s64YlzhWisBEoFMyT5EMdE-WGKmDXaheIkWlxuLn93TARAWpN3UU8vvf8jH-zDNkEfO4enrgXS0D7s56yujz1uHUCuKUWosflgk-nvac3WGcTjM9r8EOF\_NleAM2mahdacM5Pd0XZ9vfeYUWjLlHaE2eSl9n8XU\_\_ix1Kflb4AZpMKOnbsEnI9w",  
 "use": "sig"  
 },  
 {  
 "e": "AQAB",  
 "kid": "entity-statement-key-2",  
 "kty": "RSA",  
 "n": "mjhhLGa5QTHr4EwFqqYnaa2mfnByaDsUueFY6tQoHBw9paRGV3vs7RcLv9DsOeqehDBJFedQyzObCLEOJZry7Z4\_3ehXccf\_J2ZLSwELhBp5iMlBfFGCY1NcGze2cmpaeMKqkuG6M7L3y5G2UdlytyXBb5labNNAECTmXkoMIj-j4H5oex8ZI1PaDRFLnaKlBceqNebgCjan7SkSkGRpvnIfjiZ77YCsf9gEGAPJF6pd3SHayQbmoxUAyM5OjuCtvhAyXJnIvG7InUeI8Qxlwb8vjxUj6v\_BFU2wC5mBvomWUVTLZhXWuWOeS2JctppBCHUDYqgKsyjHzheThbeHjQ",  
 "use": "sig"  
 }  
 ]  
 },  
 "iat": 1661212800  
}

## Signed JWKS

### JWT

eyJhbGciOiAiUlMyNTYiLCAia2lkIjogImVudGl0eS1zdGF0ZW1lbnQta2V5LTEiLCAidH  
lwIjogIkpXVCJ9.ew0KICAic3ViIjogImh0dHBzOi8vZXhhbXBsZS5jb20iLA0KICAiaXN  
zIjogImh0dHBzOi8vZXhhbXBsZS5jb20iLA0KICAia2V5cyI6IFsNCiAgICB7DQogICAgI  
CAiZSI6ICJBUUFCIiwNCiAgICAgICJraWQiOiAiaWQtdG9rZW4ta2V5LTEiLA0KICAgICA  
gImt0eSI6ICJSU0EiLA0KICAgICAgIm4iOiAidUNvRzdfeW5UYTdScndZZ0JqRlZBM3hsY  
WFBU2Z1Q053LWlYQTNidGRTSFN2N19fLWtXeWtwSFpfOUFGVGVMWnJFcFQ5MU80WlFZM19  
HNXZDREVaU2cwV1ZwSlBoM3FDR0NDQURCbUV3bmlLQllMQkM0Ml8xVS1fU1pKYmRud0gyU  
nNDTUtCWHNmamdvRUFGUVRWMkE5UHJDX0pQT0pDSTlkNWROLUFyNmdQMU9KaFlKY2RaeGh  
CV0lkTXZkVV9uQUlpZ3J2MjNUSEtHMTdTaDF1cTBKSW5fOUx0RVBxVE9Uc1J2NlVRWXRNY  
XhON1VFcWpxZ0pocXBTYV8wZTJFeEJvamhtdnY3eEZQV0JMTkFaZWhhcUIwajZQZWNCem9  
vWWdkemdJRE9OYlVxdzZ0TEU3LTdIcHRNN2pKc1NpaEV0OXlWTTEzanl1VkxrVTFRcGhxU  
EFRIiwNCiAgICAgICJ1c2UiOiAic2lnIg0KICAgIH0sDQogICAgew0KICAgICAgImUiOiA  
iQVFBQiIsDQogICAgICAia2lkIjogImlkLXRva2VuLWtleS0yIiwNCiAgICAgICJrdHkiO  
iAiUlNBIiwNCiAgICAgICJuIjogIjFBamdaTWtKcVVQczVmd0dicE9Fa2o3ZGF4SE85bU1  
lc0Y3ODIweXUxTG0tS2R2V2VCZG5weXc3M1BwQ29aYXRtLVhyMUtEeXo0Vi1VWjJSalpSR  
GRDSkhTVFRXaUVOS3cwS2xSNnZjaEJpNmlMeVNFU25sQkEwN1dwckoxX0xjZTNpRldVRVB  
hVEZHaGZWSFRmVHVfcS01TVpBTFVDOU0tdW1tUGxJdWVpMm9Fck1xTzJPSFFqYXkwd0JFS  
ml3U0FjNUxIejdCYlFwOVhzTVBPMGpOQW40T2tPdC1HazYtOHBLNnExSXR1c1gxVUFiekF  
vZW9kNDVMN2N4TmZBWER6OXFNRkJYeHNmOU44N3NhZDBSd0EyRUVmR2N3TUFXQ3oyQy1na  
jU0V2ZZMXlOc1dZRy1yMEhUbnhPZWxTOGJOM0dRT3Flc052N3VycU0yaXNld1F6dyIsDQo  
gICAgICAidXNlIjogInNpZyINCiAgICB9DQogIF0NCn0.GvoYwYAxw8G0o9iz6TeRtg1wy  
w1r9JXDN7lL94BKSBRKCSw7Fx3xO11PvNpcyNx97aemouXeVrDMeHArmIgOv8fm7bjBvbe  
017vAf-EH6iXEljabICyMzGh6vELcGSa5fJw6twfc6sPNmTJkkV96BLeaUikr6otPa7Vzw  
i3uYeiUt9E2qmqYjZHsDGwoXuss9WrvFAN6TAcsbjOC9buglkRd7LagAmNHhv4KY6mCXH-  
k1Wg3OJTFFCNqFm79DNnii3fo0PWolaFBkbJdVdtRpZGBjLdcRdtj1pB4G233ziVyrFZXj  
y4i3dzAKP4fnXW6Qbsj4SbEDbVGRHGvx6oe0g

### JWT header

{"alg": "RS256", "kid": "entity-statement-key-1", "typ": "JWT"}

### JWT body

{  
 "sub": "https://example.com",  
 "iss": "https://example.com",  
 "keys": [  
 {  
 "e": "AQAB",  
 "kid": "id-token-key-1",  
 "kty": "RSA",  
 "n": "uCoG7\_ynTa7RrwYgBjFVA3xlaaASfuCNw-iXA3btdSHSv7\_\_-kWykpHZ\_9AFTeLZrEpT91O4ZQY3\_G5vCDEZSg0WVpJPh3qCGCCADBmEwniKBYLBC42\_1U-\_SZJbdnwH2RsCMKBXsfjgoEAFQTV2A9PrC\_JPOJCI9d5dN-Ar6gP1OJhYJcdZxhBWIdMvdU\_nAIigrv23THKG17Sh1uq0JIn\_9LtEPqTOTsRv6UQYtMaxN7UEqjqgJhqpSa\_0e2ExBojhmvv7xFPWBLNAZehaqB0j6PecBzooYgdzgIDONbUqw6tLE7-7HptM7jJsSihEt9yVM13jyuVLkU1QphqPAQ",  
 "use": "sig"  
 },  
 {  
 "e": "AQAB",  
 "kid": "id-token-key-2",  
 "kty": "RSA",  
 "n": "1AjgZMkJqUPs5fwGbpOEkj7daxHO9mMesF7820yu1Lm-KdvWeBdnpyw73PpCoZatm-Xr1KDyz4V-UZ2RjZRDdCJHSTTWiENKw0KlR6vchBi6iLySESnlBA07WprJ1\_Lce3iFWUEPaTFGhfVHTfTu\_q-5MZALUC9M-ummPlIuei2oErMqO2OHQjay0wBEJiwSAc5LHz7BbQp9XsMPO0jNAn4OkOt-Gk6-8pK6q1ItusX1UAbzAoeod45L7cxNfAXDz9qMFBXxsf9N87sad0RwA2EEfGcwMAWCz2C-gj54WfY1yNsWYG-r0HTnxOelS8bN3GQOqesNv7urqM2isewQzw",  
 "use": "sig"  
 }  
 ]  
}