

@Web Conference  
4 September 2024

# OAuth SIG Community 15<sup>th</sup> Meeting (60<sup>th</sup> from ex-FAPI-SIG)

# Table of Contents

## Ongoing

1. OID4VCI

## Standing Still

1. DPoP
2. Token Exchange
3. Passkeys

## Keeping Watch

1. OpenID Federation 1.0
2. SSF
3. OpenID Connect Native SSO for Mobile Apps 1.0
4. OIDC4IDA

## Community Event

Keyconf 24

# Ongoing

# OpenID Connect for Verifiable Credentials (OID4VCs)

■ Goal: Keycloak can work as an issuer of VCs. (currently it is an **experimental** feature)

- Phase 1: supported as an experimental feature
- Phase 2: supported as a preview feature
- Phase 3: supported officially

Completed by KC 25

In Progress

■ Epic issue

- <https://github.com/keycloak/keycloak/issues/25936>

■ Status: 23 out of all 25 issues were resolved. (no progress)

Keycloak **25** starts supporting passkeys as an **experimental** feature.

■ Discussion

- <https://github.com/keycloak/keycloak/discussions/17616>

■ Design

- <https://github.com/keycloak/keycloak-community/blob/main/design/OID4VCI.md>

# OpenID Connect for Verifiable Credentials (OID4VCs)

## ■ Current focus points

- Where we define the credentials:

Per client: client attributes (KC25 took this option)

Per realm: protocol mapper's configuration (from the FUNKE challenge), realm attributes

- “Protocol” attribute of a client for OID4VCI:

Different protocol: oidc client for OAuth2/OIDC while oid4vc client for OID4VCI (KC25 took this option)

Same protocol: oidc client for both OAuth2/OIDC and OID4VCI (from the FUNKE challenge)

Note: OID4VCI seems to be an extension of OAuth2 like OIDC.

So, it might be better for OID4VCI spec to have dedicated scope value for OID4VCI the same as OIDC.

If so, it is ease for us to distinguish whether a request from a client is for OIDC or OID4VCI

E.g., OIDC - scope=openid , OID4VCI - scope=oid4vc

## ■ Guide: [adorsys/keycloak-ssi-deployment](https://adorsys.github.io/keycloak-ssi-deployment/)

## ■ Breakout sessions

24<sup>th</sup> : 14 Aug (<https://cloud-native.slack.com/archives/C05KR0TL4P8/p1723709866534169>)

25<sup>th</sup> : 21 Aug (<https://cloud-native.slack.com/archives/C05KR0TL4P8/p1724304792655579>)

26<sup>th</sup> : 28 Aug (<https://cloud-native.slack.com/archives/C05KR0TL4P8/p1724852714357779>)

# Standing Still

# Demonstrating Proof-of-Possession (DPoP)

- Goal: Officially supported (currently it is a **preview** feature)

- Status: In progress

Keycloak **23** starts supporting DPoP as a **preview** feature.

- Epic issue

- <https://github.com/keycloak/keycloak/issues/22311>

- Status: 2 of 4 issues were resolved (+1 issue added)

- Remaining issue: Fully decouple DPoP from TokenEndpoint and TokenManager if possible

- Issue: <https://github.com/keycloak/keycloak/issues/21921>

Due to other contributions, DPoP is decoupled from TokenEndpoint, and

There is the only one remaining part of DPoP in TokenManager.

# Token Exchange

- Goal: Officially supported (currently it is a **preview** feature)

- Status: In progress

Keycloak (unknown version) supported token exchange as a **preview** feature.

- Epic issue

- <https://github.com/keycloak/keycloak/issues/31546>

- Status: 1 out of all 8 issues were resolved. (no progress)

- Discussion

- <https://github.com/keycloak/keycloak/discussions/26502>

- Resources:

- Specification: RFC 8693 OAuth 2.0 Token Exchange
- Keycloak's implementation
  - [https://www.keycloak.org/docs/latest/securing\\_apps/index.html#\\_token-exchange](https://www.keycloak.org/docs/latest/securing_apps/index.html#_token-exchange)



# Passkeys (Multi-Device FIDO Credentials)

■ Status: Goal: Officially supported (currently it is a **preview** feature)

■ Status: In progress

Keycloak **23** starts supporting passkeys as a **preview** feature.

■ Discussion

- <https://github.com/keycloak/keycloak/discussions/16201>

■ Epic Issue: 6 of 11 issues were resolved (no progress)

- <https://github.com/keycloak/keycloak/issues/23656>

■ Resources

- FIDO Alliance : <https://fidoalliance.org/white-paper-multi-device-fido-credentials/>
- Apple : <https://developer.apple.com/passkeys/>
- Google : <https://developers.google.com/identity/passkeys/>
- passkeys.dev : <https://passkeys.dev/>
- passkeys.io : <https://www.passkeys.io/>
- passkeys futures detection: <https://featuredetect.passkeys.dev/>

# Keeping Watch

# OpenID Federation 1.0

## ■ Overview

Establishing trust between Entities like OPs and RPs by means of a trusted third party called a Trust Anchor, whose main purpose is to issue statements about Entities.  
(Ex. even if an RP has not yet been registered to an OP)

## ■ Motivation

- Enabling Automatic Registration of Clients within the European ecosystem:  
Payment Service Directive (PSD), Payment Services Regulation (PSR), Financial Data Access (FIDA)
- In research & education field, eduGAIN's OpenID Federation profile support.

## ■ Specification (Implementer's Draft)

- [https://openid.net/specs/openid-federation-1\\_0.html](https://openid.net/specs/openid-federation-1_0.html)

## ■ Discussion

- <https://github.com/keycloak/keycloak/discussions/31027>

## ■ Adopters

- Italian government (Sistema Pubblico di identità Digitale (SPID) / Carta di Identità Elettronica (CIE))  
<https://docs.italia.it/italia/spid/spid-cie-oidc-docs/it/versione-corrente/index.html>
- Global Assured Identity Network (GAIN) PoC  
<https://openid.net/cg/gain-poc/proof-of-concept/>

PROPOSED DRAFT

# OpenID Federation 1.0

## ■ Reference Implementation:

- Keycloak for the European Open Science Cloud:

<https://github.com/eosc-kc/keycloak-oidc-federation>

- Thomas Darimont (forked eosc-kc implementation):

<https://github.com/thomasdarimont/keycloak-oidc-federation/tree/update/keycloak-25.0.x>

## ■ Resources:

Connect2id:

<https://connect2id.com/learn/openid-federation>

<https://connect2id.com/learn/openid-federation-metadata-policies>

Sphereon: API specifications

<https://app.swaggerhub.com/apis/SphereonInt/OpenIDFederationAPI/1.0.0-d35>

OpenID Foundation: conformance suite (conformance test development in progress)

<https://gitlab.com/openid/conformance-suite/>

# Shared Signals Framework (SSF)

## ■ Overview

Framework for sharing security events related to authenticated users among systems.

Currently, there are two applications using SSF:

- Continuous Access Evaluation Protocol (CAEP): Events about an authenticated user and its sessions
- Risk Incident Sharing and Coordination (RISC): Events about user accounts, credentials, identifiers

## ■ Motivation

Some major platform vendors supported/required it:

- [Apple Business Manager, Apple School Manager](#)
- [Google Cross-Account Protection \(RISC\)](#)

## ■ Standardization Body

[OpenID Foundation - Shared Signals Working Group](#)

## ■ Specification (Implementer's Draft)

- SSF: [https://openid.net/specs/openid-sharedsignals-framework-1\\_0-ID3.html](https://openid.net/specs/openid-sharedsignals-framework-1_0-ID3.html)
- CAEP: [https://openid.net/specs/openid-caep-interoperability-profile-1\\_0-ID1.html](https://openid.net/specs/openid-caep-interoperability-profile-1_0-ID1.html)
- RISC: [https://openid.net/specs/openid-risc-profile-specification-1\\_0.html](https://openid.net/specs/openid-risc-profile-specification-1_0.html)

# Shared Signals Framework (SSF)

## ■ Discussion

[Support RISC and CAEP events / Shared Signals and Events #14217](#)

## ■ Resources

- [Shared Signals](#): explaining SSF by Cisco
- [caep.dev](#): playground for testing SSF implementation

# OpenID Connect Native SSO for Mobile Apps 1.0

## ■ Overview:

SSO in mobile apps: Share user authentication information among mobile applications installed on the same device.

Current BCP recommends to use a session cookie on a system browser, but some problems:

- The session cookie could be deleted by a user.
- The session cookie could not be shared if a user uses private browsing.

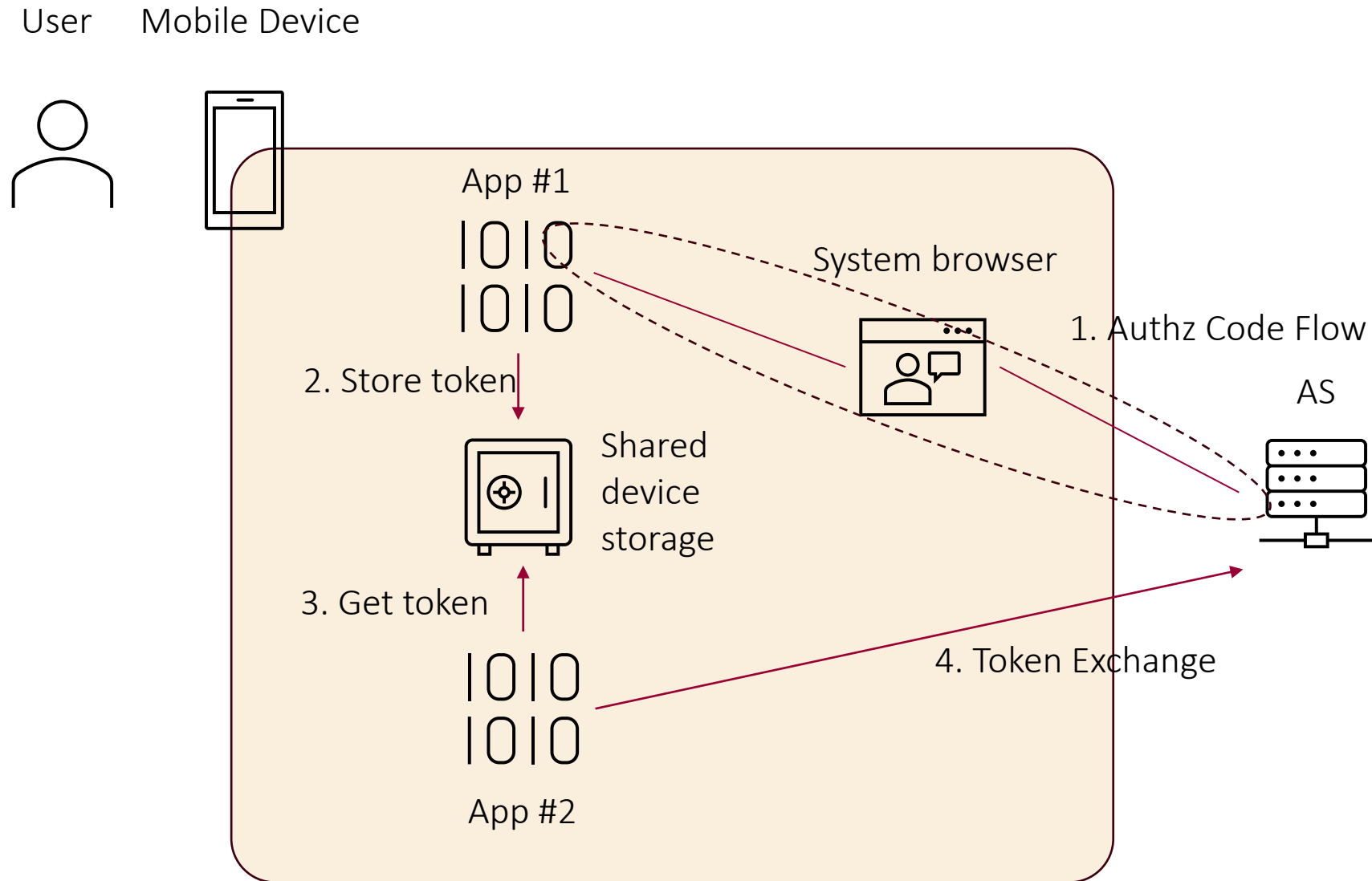
To avoid using the session cookie, use token exchange.

It could be one of use-cases of Token Exchange.

## ■ Specification (Implementer's Draft)

- [https://openid.net/specs/openid-connect-native-sso-1\\_0.html](https://openid.net/specs/openid-connect-native-sso-1_0.html)

# OpenID Connect Native SSO for Mobile Apps 1.0





# OpenID Connect for Identity Assurance 1.0 (OIDC4IDA)

## ■ Status: In progress

Waiting for discussion and review.

## ■ Specification

- OpenID Connect for Identity Assurance 1.0

Draft

[https://openid.net/specs/openid-connect-4-identity-assurance-1\\_0.html](https://openid.net/specs/openid-connect-4-identity-assurance-1_0.html)

## ■ Discussion

- Support for OIDC extensions: OIDC4IDA

<https://github.com/keycloak/keycloak/discussions/21270>

## ■ Implementation

- implement oidc4ida

Draft PR sent


Draft PR: <https://github.com/keycloak/keycloak/pull/21309>

# Community Event

# KeyConf 24

**Tickets sold out!**

**Live Stream is available**

- Date and Time: 10 AM to 4 PM, September 19th Thursday
- Venue: ARCOTEL Kaiserwasser, Vienna/Austria 
- Website : <https://keyconf.dev/>



END