FAPI-SIG Community 42nd Meeting

@Web Conference

1 Feb 2023

PROPOSED DRAFT

Table of Contents

Status: EdDSA support

Status: EdDSA support

Schedule: send PR by 28 Feb at the latest.

Current problems:

- Keycloak Guides Maven Plugin build fails when building keycloak

```
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-plugin-plugin:3.6.0:descriptor (default-descriptor) on project keycloak-guides-maven-plugin: Execution default-descriptor of goal org.apache.maven.plugins:maven-plugin-plugin:3.6.0:descriptor failed: Unsupported class file major version 61
```

It seems that org.apache.maven.plugins:maven-plugin-plugin:3.6.0:descriptor does not support Java 17 (major version 61)

EdDSA related classes were introduced from Java 15 (major version 59)

Keycloak Guides project is for generating guides document so that the project is not related to keycloak itself.

Enlarging a scope of FAPI-SIG and renaming FAPI-SIG

Enlarging a scope of FAPI-SIG and renaming FAPI-SIG

Schedule (proposed):

- Continue proposing new scope and new name on 42nd meeting (1 Feb).
- Vote new scope and new name on 43rd meeting (1 Mar)
- Change scope and name from 44th meeting.

Determined issues:

- Change SIG to WG

Enlarging a scope of FAPI-SIG and renaming FAPI-SIG

New scope and name candidates:

- Security for … :

  Security WG, IAM Security WG, API Security WG, Secure Digital Identity WG,

- Specifications for … :

  OIDC WG, OpenID WG, OAuth WG, Identity standards WG,

- Keycloak community for … :

  Keycloak Governance WG, Keycloak Alignment WG, Keycloak Community WG,

  Keycloak Authorization WG, Keycloak Identity WG

Keyconf’ 23

Keyconf’ 23

Keycloak conference held on 2019

Keyconf’ 19

Host: UK Research and Innovation, Science and Technology Facilities Council, Hartree Centre

Date: 12 and 13 Jun 2019 (2 days)

Venue: STFC Hartree Centre, Sci-Tech Daresbury, Warrington, United Kingdom

Web site: https://www.hartree.stfc.ac.uk/Pages/KeyConf.aspx (dead link)

Participants: about 20 people (including Stian, Marek and me)

Registration Fee: nothing

Program: 12 talks and 4 unconferences

Keyconf’ 23

Current candidate:

Scale of the conference: the same as Keyconf’ 19 (small size, ~50 participants?)

Venue: London, UK

Date: run along side a major identity event/conference in London

1 day or 2 day?

Audience: TBD (developers, users / novice, intermediate, professional?)

Theme: TBD (sessions, workshops?)

Proposal: Handle type token

Handle type token

Two types of token:

Assertion type token: including information.

Handle type token: not including information.

An authorization server provides information in return for the token

Current situation:

Keycloak supports an assertion type.

Motivation for a handle type token:

Privacy: Some use cases require not including sensitive data like PII in an access token.

Handle type token: current situation

How to realize a handle type access token in a current keycloak:

Lightweight access token: discussed but not yet realized (and not a handle type strictly).

Discussion: https://github.com/keycloak/keycloak/discussions/9713

PR (Draft) : https://github.com/keycloak/keycloak/pull/8914

Access token encryption: tried but not realized.

PR (Closed) : https://github.com/keycloak/keycloak/pull/6796

https://github.com/keycloak/keycloak/pull/7340

https://github.com/keycloak/keycloak/pull/7341

Handle type token: proposal

How to realize a handle type access token in a current keycloak:

Converting an assertion token to a handle token and vise versa.

Concept:

Keycloak continues treating an access token as an assertion type.

When creating an access token in token endpoint:

1. creating an assertion type access token as usual.
2. creating a handle type access token referring the assertion type access token.
3. storing both types of access tokens and their relationship somewhere.
4. returning a handle type access token to a client.

Handle type token: proposal

Concept:

Current keycloak continues treating an access token as an assertion type.

When receiving an access token in introspection endpoint:

1. retrieving an assertion type access token from some store by a handle type token.
2. processing and returning the result of introspection based on the assertion as usual.

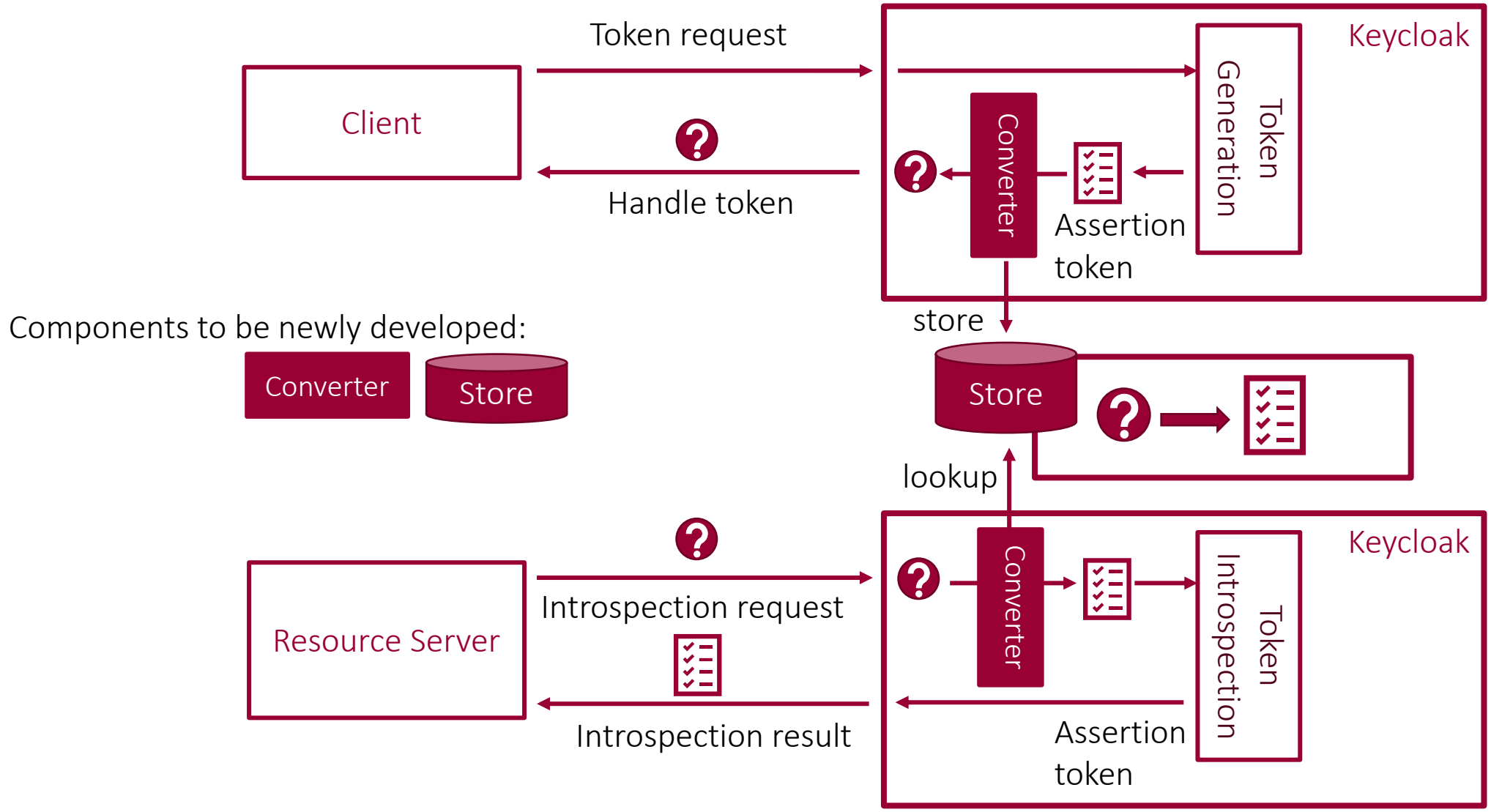
When receiving an access token in revocation endpoint:

1. retrieving an assertion type access token from some store by a handle type token.
2. processing and returning the result of revocation based on the assertion as usual.

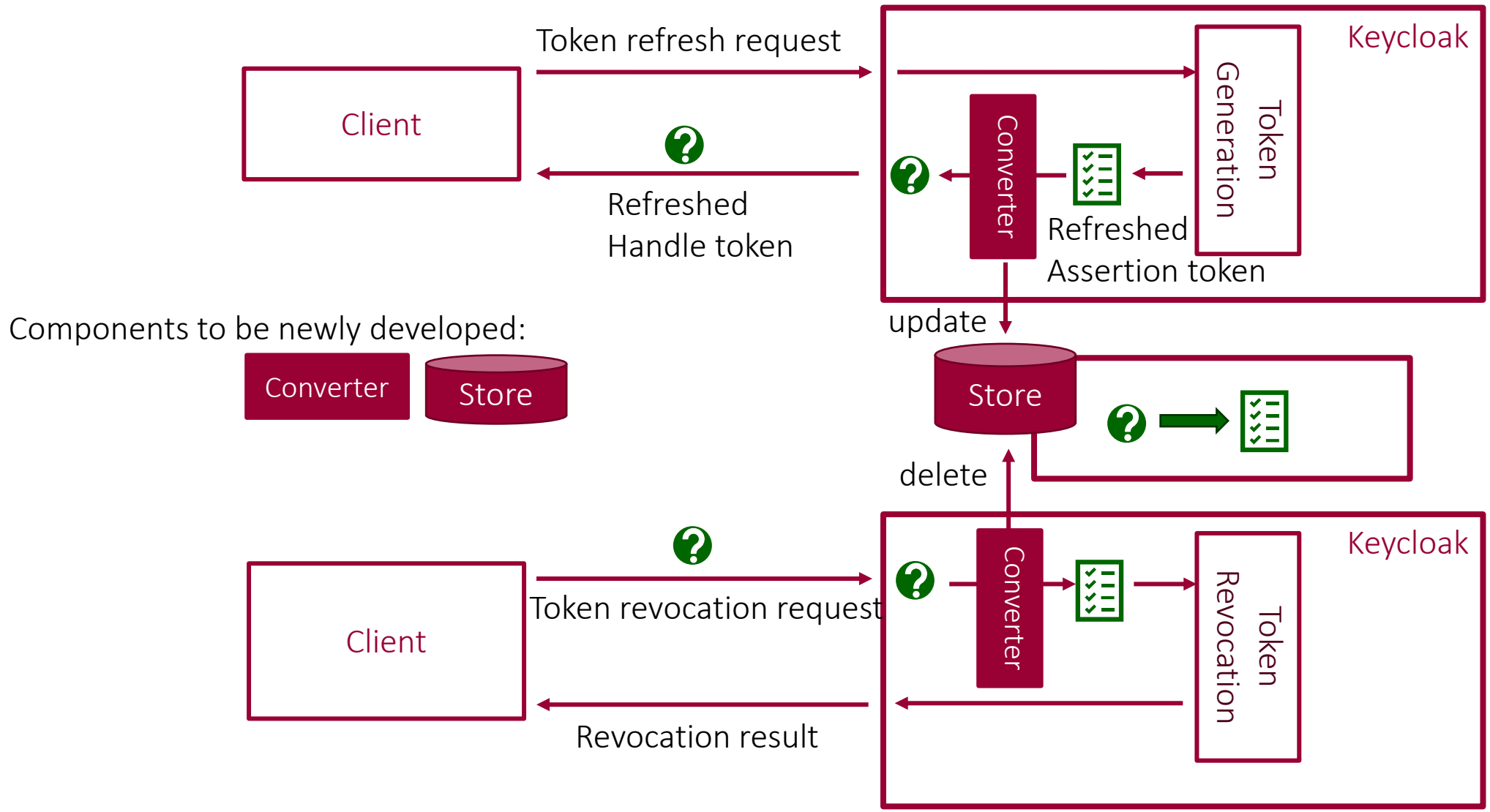
When receiving an access token in token endpoint for refresh:

1. retrieving an assertion type access token from some store by a handle type token.
2. processing and returning the result of token refresh based on the assertion as usual.

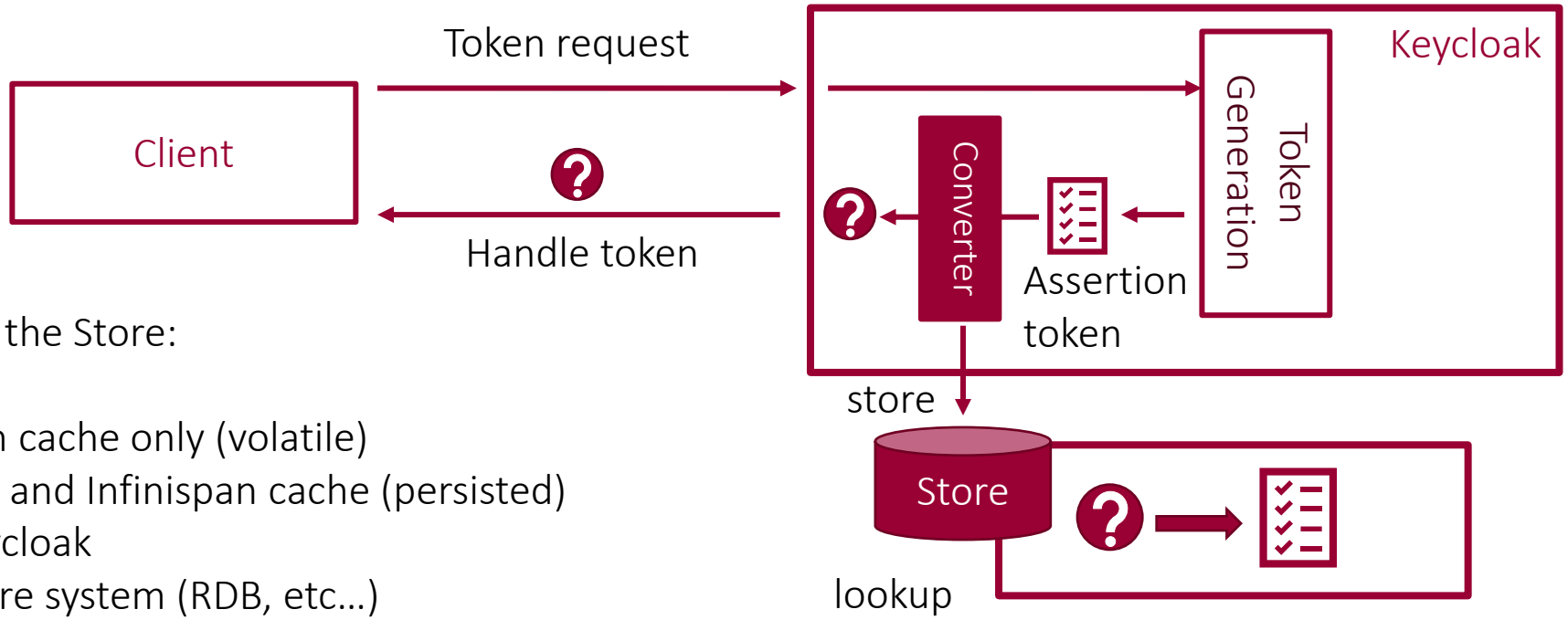
Handle type token: proposal



Handle type token: proposal



Handle type token: proposal



How to realize the Store:

1. In keycloak
   1. Infinispan cache only (volatile)
   2. Database and Infinispan cache (persisted)
2. Outside keycloak
   1. Some store system (RDB, etc…) accessing it via APIs

Challenges:

Sync lifecycle of an access token with stored data (CRUD)

TBD:

Need a handle type refresh token?

Proposal: sigstore signing for keycloak

sigstore

The Linux Foundation Open SSF project for signing, verifying and protecting software

How about adopting sigstore to keycloak?

https://www.sigstore.dev/

Merit :

[1] It is expected that CNCF projects be required to support sigstore.

E.g., kubernetes, istio are signed by sigstore

https://blog.sigstore.dev/new-sigstore-landscape-add-your-signed-project-dda0517723b6

[2] It is expected that customers require them supply chain security as part of requirements for procurement in the future.

END