Keycloak - future feature ideas

Marek Posolda
Keyconf 2023

Disclaimer

- These are just possible ideas for the future features in Keycloak

- No guarantee when (or if ever) they would be implemented and if implemented in same way proposed on these slides

Client types

Motivation - example use-case

- You want to create many different **service account** clients in your deployment

- All **service account** clients are usually very similar in some aspects

- Possibly repeating of filling same attributes for all **service account**

- And then I figured I want to change some attribute for my service accounts ...
  Need to update my 100 service-account clients :-(

Client types - idea

- Ability to define **type** when creating client in Keycloak
- Type can be for instance **single page application** or **service account**
- Type describes some client attributes are read-only, some writable, some not applicable
- Add hardcoded values (for read-only attributes) or default values (for writable attributes)

Other capabilities

- Inheritance (EG. "service-account" client type may inherit from type "oidc-client")

- Versioning (Can simplify migration)

- Support also more complex structures (protocol mappers, roles, role scopes)

- Definition of client types in admin console as well as in JSON editor (similar to client policies)

- Default (built-in) client types (may not be editable)

Other sources

- Discussions
  - https://github.com/keycloak/keycloak/discussions/8497
  - https://github.com/keycloak/keycloak/discussions/9066

User Profile

User profile - intro

- Ability to define what user attributes should be present on forms (registration form etc)

- Similar for administrators

- Some attributes might be read-only for users, some even for administrators

- Validations

- Support in UI

Progressive profiling

- Different client applications need specific attributes
- Don't ask user for all attributes during his initial registration
- Instead iteratively ask him for attributes needed for specific clients

Progressive profiling - use-case

- Client application **calendar** needs attribute **birthdate** of user
- User registers to Keycloak for login to application my-app
  - Keycloak will ask user just for basic attributes (username, email, first name, last name)
- Then user decides that he needs to login to client “calendar”
- OP (Keycloak) will ask user to fill his birthdate attribute during first login to calendar client

User profile & progressive profiling

- Progressive profiling not yet supported. One of the main gaps in user profile

Multitenancy

KEYCLOAK

Motivation

- Keycloak Realm is container for users, clients, roles and everything
- Each realm should have unique **issuer** to be used in tokens
- What if I want to share same **issuer** across multiple organizations?
- What if I want to have different organizations to be able to authenticate to same clients? And use same authentication flows?

Multitenancy - target

- Ability to define multiple **organizations** inside realm

- Or maybe have something like realm inheritance (**Realm of realms**) for certain aspects

- Multiple organizations can have EG. different users, but same issuer and same authentication flows

- Design is not yet polished and not 100% clear when this become a priority

Authentication policies

Motivation - example use-case

- I want all my users to be able to login with (password OR google IDP) and 2nd-level authentication (WebAuthn, OTP or Recovery codes)

- And I want users to be able to reset their password

- How to model it in current Keycloak?

- Require custom flows of "browser", "reset-password" and "post broker login" (for Google IDP)

Authentication policies

- Aim to simplify the "authentication requirements"

- Directly specify the requirements with “policies” as described above instead of a need to edit 3 different authentication flows

- More possible use-cases like:
  - Users with email domain "@keycloak.org" are required to use 2nd factor authentication (for other users it is optional)
  - Users with email domain "@keycloak.org" are required to login with `my-keycloak-idp`

Community

Keycloak OAuth SIG - possible future contributions?

- FAPI 2.0 - New version of FAPI (Keycloak already supports FAPI 1)
  - New requirements/restrictions on OAuth2 / OpenID Connect flows
  - https://github.com/keycloak/keycloak/issues/20708

- DPoP (Demonstrating proof-of-possession)
  - Alternative to MTLS Certificate-bound access token
  - MTLS is preferred, but cannot be always used
  - https://github.com/keycloak/keycloak/pull/8895
  - https://github.com/keycloak/keycloak-community/pull/254

- Reference tokens
  - https://github.com/keycloak/keycloak/discussions/19649/
  - https://github.com/keycloak/keycloak/issues/19650

Your contribution!

- See “contributors” section: https://www.keycloak.org/community

- Github discussions for contribution of new ideas
    - https://github.com/keycloak/keycloak/discussions

Store upgrade

Cross-datacenter replication

- Preview in Keycloak since 2018
- Aim was to fully support it with new store
- Current priority is to make cross-site replication supported with old store (first in active/passive mode, then active/active as a follow-up)
- New store, with the support of zero-downtime upgrade, might be delayed

Q & A

KEYCLOAK