Group call minutes 05/07/2024

Present:
Rolf Brugger (Switzerland), Marlies Rikken (Netherlands), ...

Agenda:
- Group management

Notes:

Group Management

How to do group management in a federated environment.

Initial Situation - Problem Statement

- users originate from many different organizations
  - consequence: we can’t use organizational attributes to build groups
    - (in Switch edu-ID we currently have ~70 organizations)
- users have no organizational affiliation at all
  - (in Switch edu-ID we currently have ~500'000 users without organizational affiliation)
- access to some services has to be restricted
  - but services don’t have sufficient capabilities to restrict access for individuals
    - (no waiting room, no invitation mechanism for group administrators, ...)
- services share the same access restrictions

Use Cases

Some known use cases (for Switch edu-ID).

Private library customers accessing resources on the site of an international publisher:
- mostly private users without org affiliation
- restricted access required to licensed contents of publisher

University members sharing a project space in a cloud environment:
- mainly university members with org affiliation
- project space with expensive resources (CPU, storage)
- access restricted to project members

Further education courses at university:
- course participants are not fully onboarded as official university members
- access to classes needs to be restricted to enrolled individuals

Switch procurement reselling user-based licenses:
- Switch resells (expensive) licences to DeepL translation service
- Universities define list of university members who are entitled to use the service

University operates examination environment for non academic educational institution:
- University service is to be opened to non-academic users
- users registration takes place at non-academic institution that sells certifications

Solution approaches

Restricting access to service
Further extensions:

- add provisioning engine to authorization proxy that creates accounts in the target service

Building groups

By invitation:

- group administrator sends personalized invitation to users
- invitation contains a group voucher, and has to be confirmed by user

By user request:
users request group membership on a web page (provided by the group management system)
group administrator has to confirm membership

Via API:

* groups are built elsewhere
* API to group management

By uploading user lists:

* registration process exists elsewhere
* group administrator manages list of group members
* group members are identified by
  * uniqueID
  * email address
  * ...

Also, define processes to remove groups and group members.

**Current discussion at Switch edu-ID**

We currently develop the solution approach of group management based on lists of mail addresses

* A group corresponds to a list of email addresses
* The list is managed by a group administrator
* When a user accesses a service the IdP checks if one of the user's email addresses in the edu-ID account is matches an email address in one of the groups
* For each matching group, a group-attribute (Entitlement or isMemberOf) is added to the SAML assertion / OIDC claims
* We might need the authorization proxy functionality

open questions:

* Is there an existing solution or do we have to implement it on our own?