Design Guidelines Wanted for Group Service IdP

Kaoru Maeda <kaorumaeda.ml@gmail.com>

Background

Large companies have been developing several services under the same brand. Traditionally they prepared a user account database per service. These days, demands are arising to unify these account bases into a single one under the brand, utilizing OpenID Connect and OAuth2. Design guidelines and best practices are wanted for such use cases.

Use Cases

Here are some common use cases:

- Create a new brand-wide IdP to hold all the user accounts for all the services the company provides. Such an IdP works as a client, in turn, to accommodate users who want to use social login, e.g. Google, Facebook, or Twitter. OpenID Connect takes place here. Each service under the brand provides its own set of APIs. An authorization mechanism is necessary per-service to allow different types of clients (browser, mobile apps) for the APIs. OAuth2 is used for this authorization. Authentication and basic user attributes are taken care of by the brand IdP.
- Redirection-based user authentication requires password authentication of the user. This is done by the brand IdP, however, each service wants to have better integration of the password form into their service themed screen. Sometimes a user-id/password pair is grabbed by the client and sent to the IdP with Resource Owner Password Credential Flow.
- For mobile, an app is provided to support the brand IdP workflows like password change. Each service provides an app for its own purpose. Single-sign-on among co-branded apps are wanted.

Problems

Following are some common problems found under the developments of such brand-wide IdPs. Comprehensive guidelines are wanted.

- A server behaves both as a provider and a client at the same time. Authorization dance requires multiple redirections between the user browser, the brand IdP, and external IdPs.
- Security validations must happen at both OIDC level and OAuth2 level. These are different set of documents that describe security considerations.
- New attacks/vulnerabilities are reported and corresponding mitigations are proposed as extensions to the core protocols. Though they are standardized as "extensions" in the spec, employing such mitigations are considered must. E.g. pkce, mitigations against the mix-up vulnerability.
- How a self-issued IdP in the mobile device is effectively used in this picture?
- Migration paths of the existing user account to the new brand unified ID is complicated. There are users who are completely new, who has some accounts on a subset of services under unification, who has the unified ID and an account on one service but not for others, etc.
- User interface design needs considerations. What are good user consent screen? How to prevent users from confusing among different (existing) IdPs? What should be avoided? When is the best time for a user to agree on a new privacy policy?
- How to protect against CSRF when multiple redirection endpoints are involved, and when a password dialog is provided from a service, not IdP.

Suggestions

Compiling a design guidelines to realize an intermediary IdP/Authorization server. Security considerations should be explained per brand IdP level, service AS level, and clients level. Requirements from both OpenID Connect and OAuth should be glanced at once.