Extending OAuth2 to Join Local Services into a Federative SOA

M. Politze
IT Center RWTH Aachen University
Where are we now?

You are here!
Support the core processes: Teaching, Learning and Research

• Connect legacy systems with a single, consistent API

• Develop an SOA that fits to the processes at the university
  – Start with eLearning
  – Generalize and try to apply to other fields:
    ▪ Campus Management, Identity Management
    ▪ Research Data Management / eScience

• Security by design
  – Confidentiality
  – Integrity
  – Availability

• Protect personal and confidential data
OAuth2 at Commercial Service Providers

- Tightly coupled with their web services
  - Authorization for *local* scopes
  - Used for applications

- Applications using multiple services still require multiple logins
  - 1:1 mapping of services and logins
  - Hinders crossing system boundaries for process supporting application

- Authentication via authorization
  - Use *user info* supplied by a service to identify the user
  - Leads to possible security vulnerabilities [1]

Secure, device based Authorizations
- (De)Authorizations via Webinterface
- No credentials are passed to apps

OAuth2 as a service
- Integrates Shibboleth as authentication
- Possibility to provide a federative service (DFN, …)

Established at RWTH
- RWTHApp has ~20.000 active users
- Procedure scales across different applications
A Bit More Detail?

1. Access application
2. Authenticate
3. Grant access
4. Issue access token
5. Access services
6. Verify authorization

Application

Webservices

OAuth Token Service

Identity Provider
Security Implications

- The token service is the authority
- The token service is trusted
- Users are known
- Applications and web services are separated
Problem Statement

OAuth2 Workflows allow apps to cross system boundaries

• … because apps and systems are known to the OAuth2 server
• … because each user is known to the OAuth2 server
• … because systems trust the OAuth2 server to handle authorizations

Can we always assume this?

No
Extending OAuth2 to Join Local Services into a Federative SOA

Marius Politze

EUNIS 2017, 08.06.2017

Partially Solved!
Long Answer

- Federated services (SaaS)
  - Offered by one University
  - Members of other Universities may use
  - Likely each University has an OAuth2 server

- Suppose an app is using APIs from several services
  - User needs to log in multiple times
  - Application has to decide which are the correct servers
  - User likely has many places to manage authorizations

- Services need validate authorizations
  - May need to query multiple servers
  - Have to establish a trust relationship to all authorization servers
Security Implications

- The token service is the authority  
  -

- The token service is trusted    
  -

- Users are known      
  -

- Applications and web services are separated  
  -?
Goals

Always use the home institution

- Let users manage their authorizations at their home institution
- Let applications request authorizations from their home institution
- Let services validate authorizations in their home institution

Reuse existing technology for federated (web) applications

Build a federated OAuth infrastructure
OAuth2 Federated Workflow
Establishing Authority / Trust

- Local OAuth2 service remains authority
  - ... for apps
  - ... for services
  - ... for users

- Discover remote OAuth2 services

- Trust is established to local OAuth2 service
  - Local OAuth2 trusts remote services in the federation
  - Hides complexity of the federation from developers

```json
{
...
  "token_services" : {
    "https://oauth.example.com" : {
      "displayName" : "Example University",
      "namespace" : "example.com",
      "key" : "-----BEGIN PUBLIC KEY-----
      "endpoints" : {
        "authorize" : "https://oauth.example.com/authorize",
        "code" : "https://oauth.example.com/code",
        "token_info" : "https://oauth.example.com/token_info",
        "context" : "https://oauth.example.com/context"
      }
    },
    ...
  }
}
```
Knowing the User

- Transfer user information on validation
  - Reuse existing eduPerson scheme
  - Likely sufficient for many services

- Use namespaces to distinguish users
  - Reuse existing namespaces (scopes)
  - Tie user IDs to the ones delivered by authentication infrastructure

```json
{
  "isValid" : true,
  "application" : "ahcndwlsajcnalfejalsd@example.com",
  "mail" : "max.power@example.com",
  "displayName" : "Max Power",
  "eduPersonPrincipalName" : "anpqr7d@example.com",
  "eduPersonScopedAffiliation" : "student@example.com"
}
```
Conclusion

• Rising need to share services among Universities
  – Highly decentralized environments
  – Reuse of existing techniques is mandatory

• Rising demand among researchers and students
  – … to customize tools
  – … to combine existing systems

• Federated OAuth2 may satisfy some demands

• Currently evaluating proof-of-concept
  – Two OAuth instances operated at RWTH Aachen
  – In cooperation with Forschungszentrum Jülich
Thank you for your attention

Vielen Dank für Ihre Aufmerksamkeit