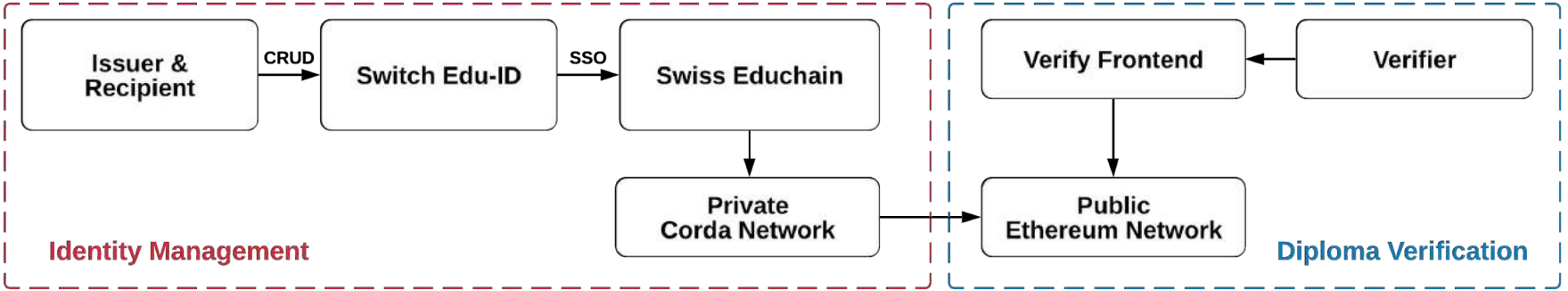
Identity Management for a Blockchain-based Certificate Issuance

Final Master Thesis Presentation
Vasileios Koukoutsas

16.03.2020

Swiss Educhain Architecture



Requirements

Requirement	Description
RQ1	Only authorized individuals are allowed to issue diplomas.
RQ2	Diploma data should be confidential to its `Recipients`.
RQ3	Process of issuing and verifying diplomas should abstract technical complexities.
RQ4	Multiple diplomas should be processable in batch.
RQ5	Verification capabilities should be accessible to anyone.
RQ6	Diplomas should be verified autonomously.
RQ7	Graduates should receive their diplomas in a digital format.

Table 4.1: Initial Educhain Requirements based on [1]

Requirement	Description
RQ8	`Recipients` should have a unique identification.
RQ9	`Recipients` should be the only ones that have the right to disclose issued credentials.
RQ10	`Recipient's` account should persist over time and be independent of any association with an Issuing Organization.
RQ11	Registration needs identity verification.
RQ12	`Issuers` should be able to revoke diplomas.
RQ13	The governance model of the Swiss Educhain system must be defined.
RQ14	Issuing should create an unchangeable audit trail.
RQ15	Data owner is responsible for data backup. System should provide an option for a participants' data to be exported.
RQ16	Multisig transactions should be possible.
RQ17	System processes data in a text-based format.
RQ18	Allow for identity details to change (e.g. name, address).
RQ19	The process to onboard Issuing Organisations to the Swiss Educhain service needs to be examined and defined.
RQ20	User accounts need to be associated with one or more Issuing Organizations.

Table 4.2: Swiss Educhain Functional Requirements

Requirement	Description
RQ21	Verifier must be able to verify the diploma even when the private environment is not available.
RQ22	Easy to use from a user perspective, with a simple UX/UI and straightforward functionality.
RQ23	Easy to install, configure, deploy, operate, monitor and maintain from a System Administrator's perspective.
RQ24	Uses technologies that are freely available, popular, well-established and mature.
RQ25	Has as few as possible technology requirements and dependencies both in terms of hardware and software.
RQ26	Can be easily integrated with existing IT infrastructure and is cross-platform compatible.
RQ27	Is not dependent on state-of-the-art technologies such as Containers, Cloud etc.
RQ28	System should support multiple issuing organizations.
RQ29	High-level access control must be defined for the different kind of identities participating in the system.
RQ30	Can be modularly enhanced by existing functionality.
RQ31	Data that are disclosed peer-to-peer should not be broadcasted.
RQ32	All transactions in the system should be signed and the identity of any action initiator should be verifiable.

Table 4.3: Swiss Educhain Non-Functional Requirements

MVP Identity Functionality

- Two types of roles supported, **Issuers** and **Recipients**.
  - Rules defined for access control.
- Account creation processes.
  - Mapping student details to Educhain accounts.
  - Educhain account automatically updates upon detail changes.

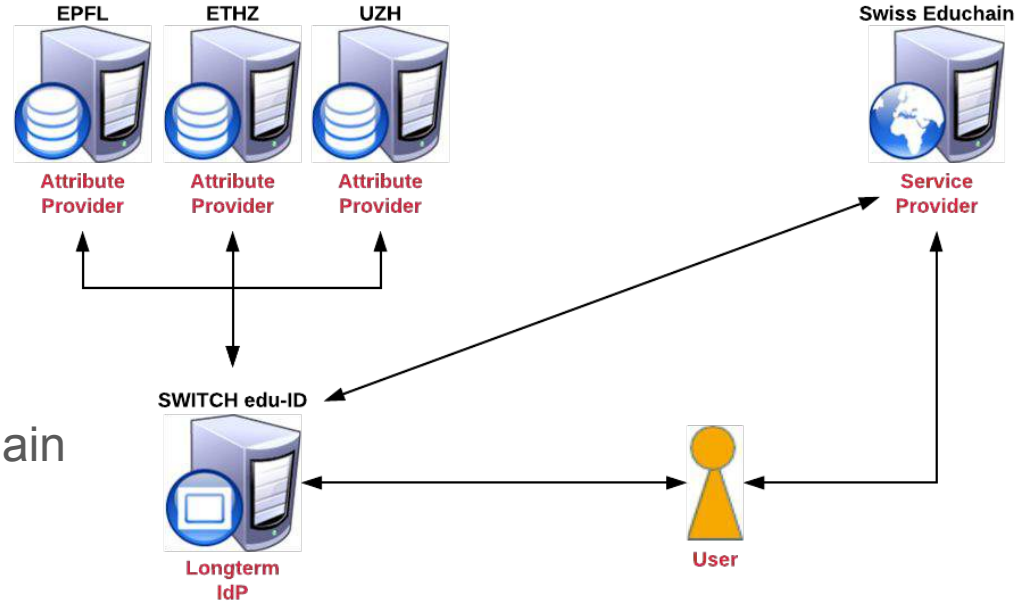
Candidate Identity solutions

- Custom IAM solution using Corda.
  - High development effort (re-inventing the wheel)
- Use an existing CorDapp Identity solution.
  - Tight dependency with third party CorDapp and IdP
- Integrate with a federated IdP service.
  - Security, authentication and data verification is provided as a service

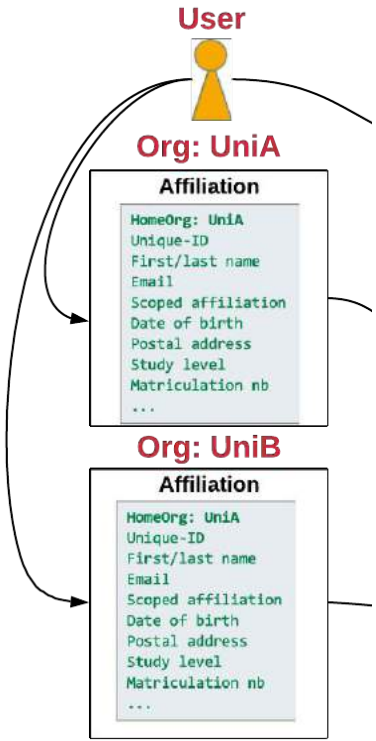
Identity Solution

SWITCH Identity Federation

- **Identity Provider**: edu-ID
  - Users: Issuers & Recipients
- **Attribute Provider:** UZH
- **Service Provider:** Swiss Educhain



Information flow and identity mapping



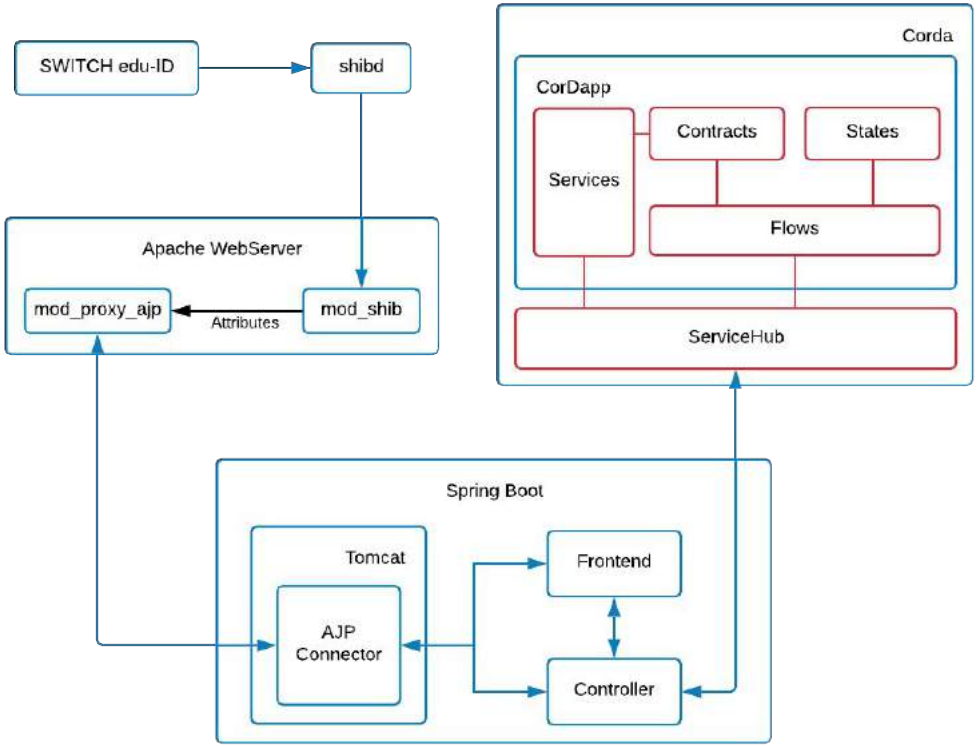
Swiss Educhain Components

Shibboleth (shibd)
- Single sign-on software

Apache Web Server (mod_*)
- Protects resources
- Exposes attributes

Spring Boot (ajp, controller, frontend)
- Fetches attributes
- Serves Frontend
- Calls Corda backend

CorDapp (States, Flows, Contracts)
- Hosts Educhain accounts



Authorization Policy

Service-Level

- ❑ Valid session
- ❑ Matriculation Nr. exists
- ❑ Linked affiliation exists

```
<Location /app/>
  AuthType shibboleth
  ShibRequestSetting requireSession true
  ShibUseEnvironment On
  <RequireAll>
        Require shib-attr swissEduIDLinkedAffiliation ~ .*@.*
        Require shib-attr matriculationNumber ~ .*
  </RequireAll>
</Location>
```

Application-Level

- ❑ Recipient
- ❑ Issuer

```
val issuersAffiliation=listOf<String>("faculty@uzh.ch","staff@uzh.ch")
val hardCodedIssuers=listOf<String>("16718991", "12715389", "12345678")
var isAllowedToIssue=false
for (it in issuersAffiliation) {
 if (swissEduIDLinkedAffiliation.contains(it)) {
  isAllowedToIssue = true }
}
if(hardCodedIssuers.contains(matriculationNumber)) isAllowedToIssue=true
```

DEMO

Evaluation

- Access to target audience (150k+ users)
- Integration with edu-ID for service-level access control
  - Backwards compatibility
  - Onboarding of verified users
  - Always up-to-date user data
- Corda for application-level access control
  - Verifiable audit trail
  - Data disclosed on a need-to-know basis

Limitations

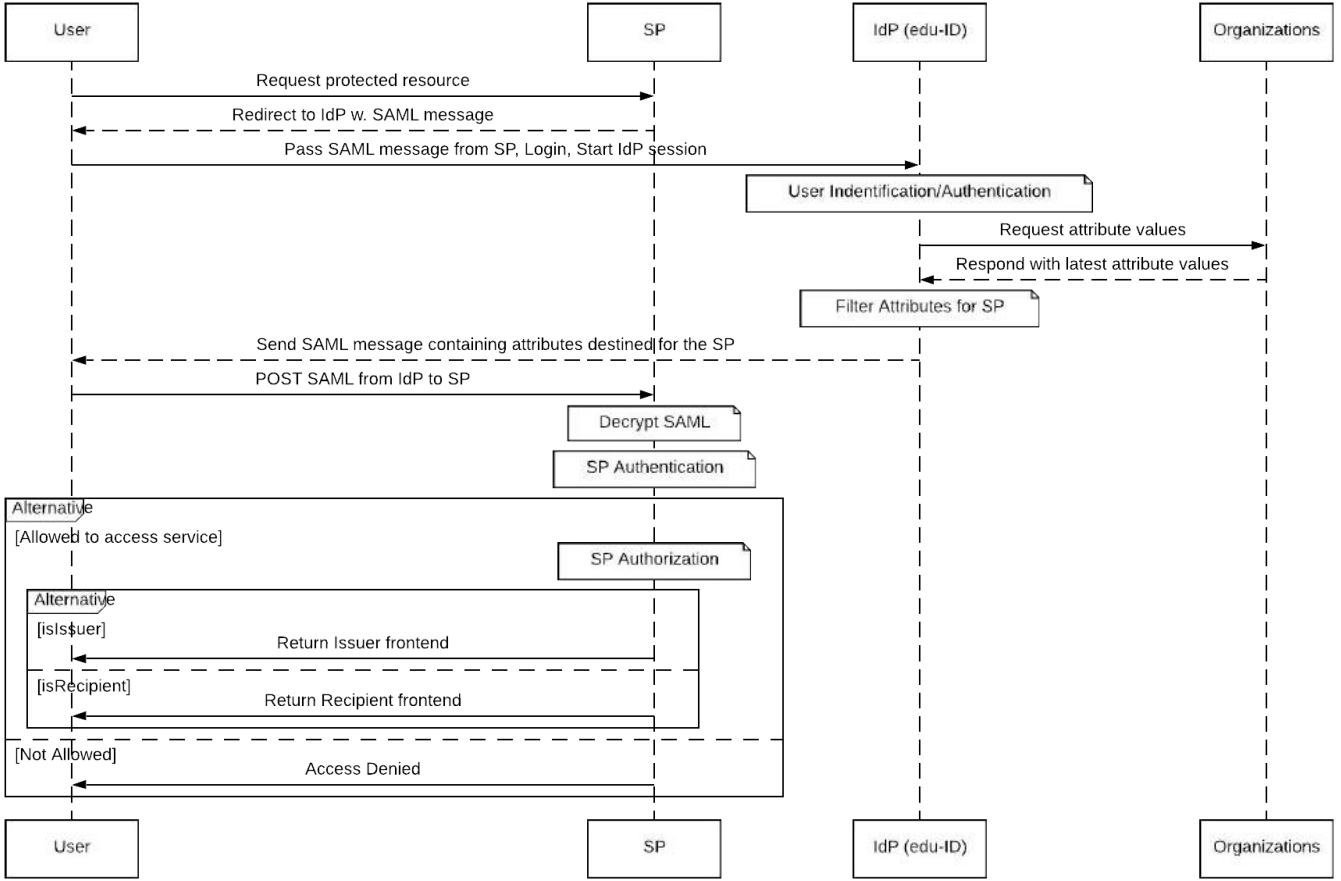
- UZH onboarding to edu-ID
- **Issuer** value for the affiliation attribute

Future Work

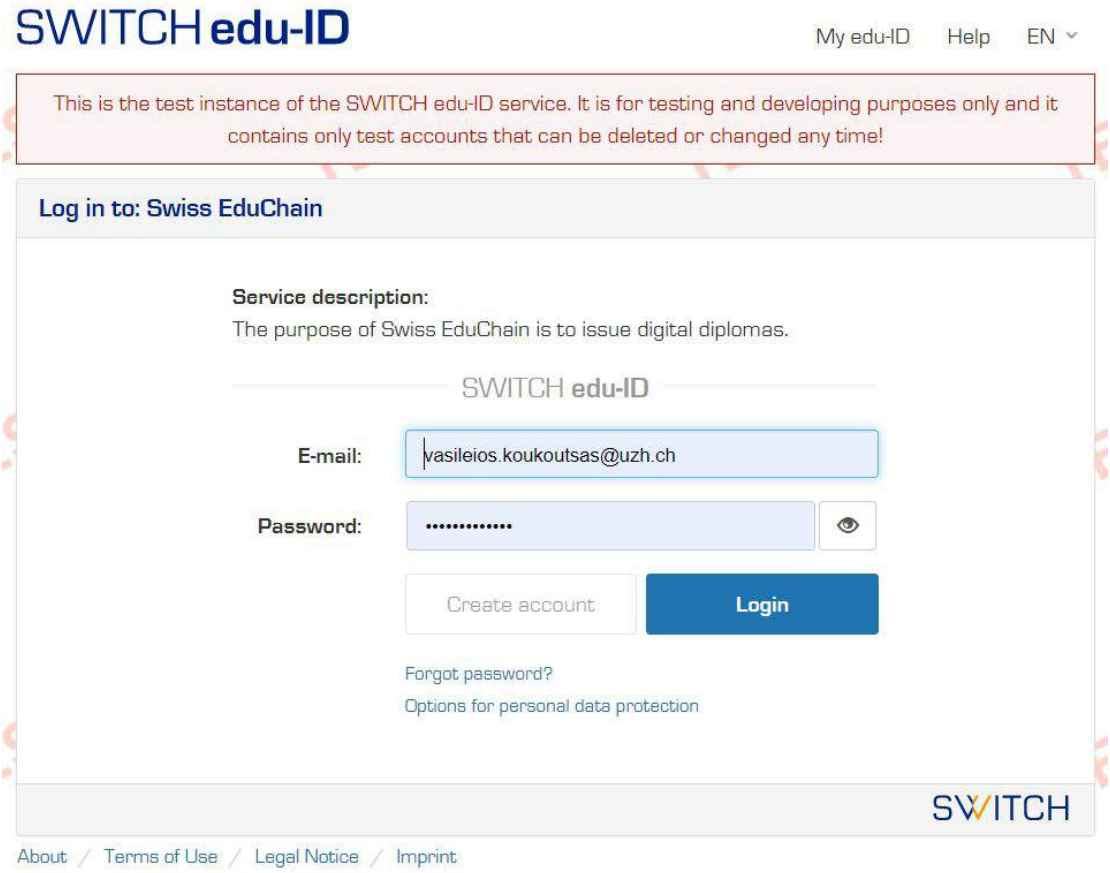
- ❏ Four eye principle
- ❏ Issuance on behalf of a specific organization
- ❏ Improved audit trail
- ❏ MFA enforcement

DISCUSSION

Login Flow


User
SP
IdP (edu-ID)
Organizations
Request protected resource
Redirect to IdP w. SAML message
Pass SAML message from SP, Login, Start IdP session
User Indentification/Authentication
Request attribute values
Respond with latest attribute values
Filter Attributes for SP
Send SAML message containing attributes destined for the SP
POST SAML from IdP to SP
Decrypt SAML
SP Authentication
Alternative
[Allowed to access service]
SP Authorization
Alternative
[isIssuer]
Return Issuer frontend
[isRecipient]
Return Recipient frontend
[Not Allowed]
Access Denied
User
SP
IdP (edu-ID)
Organizations

Demo Backup Slide 1/3



Demo Backup Slide 2/3

Corda Accounts:

- Account name: https://test.eduid.ch/idp/shibboleth!https://educhain.csg.uzh.ch/shibboleth!so3WNOJLA+Fp/SuC8E7KbK6wgFc=
- Host: O=PartyA, L=Zurich, C=CH
- ID: d4d1070a-e671-4d6d-a5a5-885e8c9a8fb7

- Account name: https://test.eduid.ch/idp/shibboleth!https://educhain.csg.uzh.ch/shibboleth!Ypu9XG2Hn4MQqGCScdMBoeWSkiE=
- Host: O=PartyA, L=Zurich, C=CH
- ID: 8b025bd3-1995-4099-a34a-d8955ce0458a

- Account name: identityService
- Host: O=PartyA, L=Zurich, C=CH
- ID: e166aa63-ecdf-4f4b-9fea-97502b71ff13

EduChain Accounts:

- Persistent-ID: https://test.eduid.ch/idp/shibboleth!https://educhain.csg.uzh.ch/shibboleth!so3WNOJLA+Fp/SuC8E7KbK6wgFc=
- CN: Vasileios Koukoutsas
- Mail: vasileios.koukoutsas@uzh.ch
- Matriculation Number: 16718991
- swissEduIDLinkedAffiliation: member@uzh.ch;student@uzh.ch
- isAllowedToIssue: true

- Persistent-ID: https://test.eduid.ch/idp/shibboleth!https://educhain.csg.uzh.ch/shibboleth!Ypu9XG2Hn4MQqGCScdMBoeWSkiE=
- CN: Simon Müller
- Mail: simondo.mueller@bluewin.ch
- Matriculation Number: 12715389
- swissEduIDLinkedAffiliation: member@uzh.ch;student@uzh.ch
- isAllowedToIssue: true

Demo Backup Slide 3/3