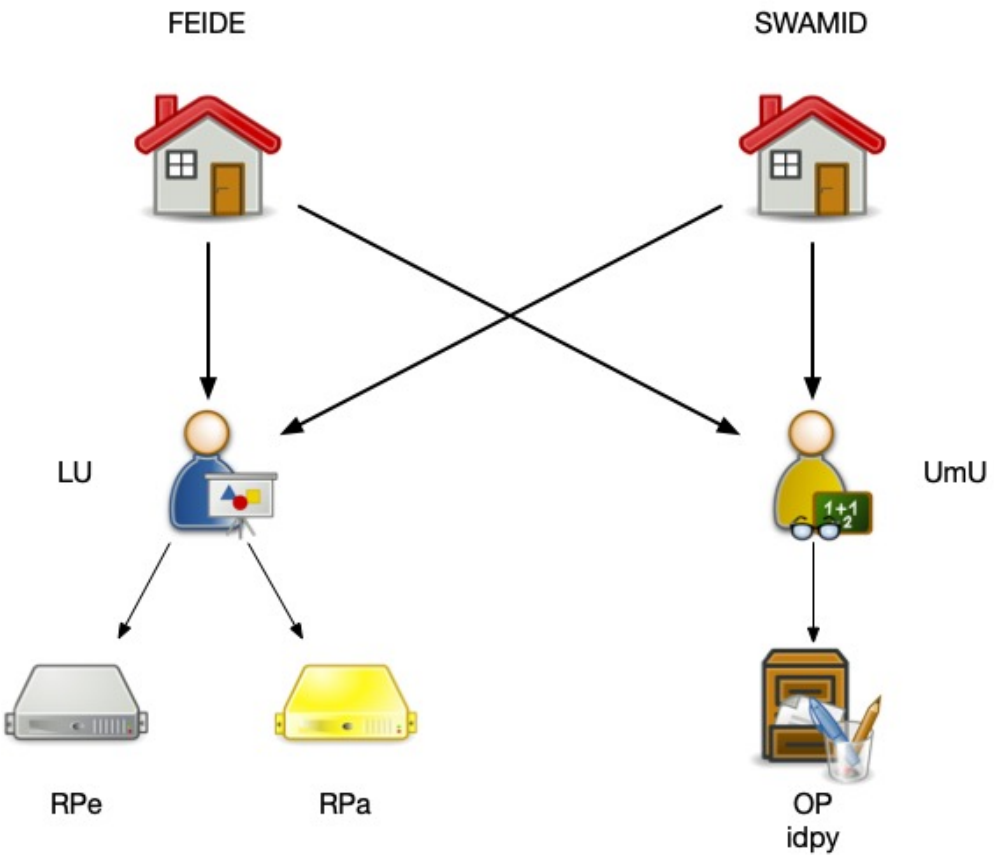
Interoperability testing setup

Participants

Henri Mikkonen Shibboleth/GEANT
Vladimir Dzhuvinov, Connect2id
Jouke Roorda, NIKHEF
Roland Hedberg, Catalogix/IdentityPython

Basic setup

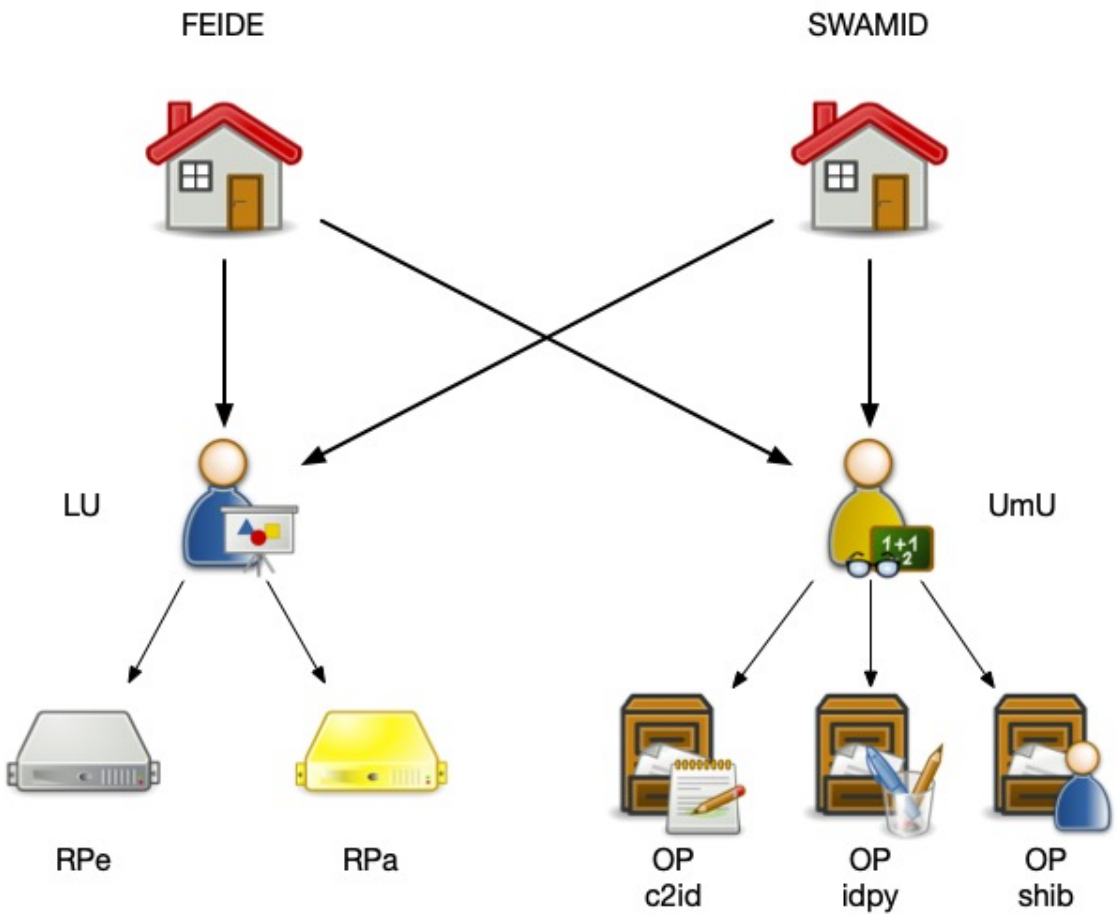
The basic setup (provided by me Roland Hedberg) contained one federation with the following entities:



- An RP that does explicit registration (RPe)
- An RP that does automatic registration (RPa)
- An OP that can handle explicit and automatic registration
- 2 intermediates representing 2 organisations (one owning the RPs and the other the OP)
- 2 federation operators (FO)

To this framework it was then possible to add the OPs and RPs the test participants provided. In this case 2 more OPs.

Which gave us this setup.



Which consisted of having both intermediates (organisations) belonging to both federations. The changes in configuration of the RPs and OPs what there now was a new trust anchor. For the organisations they now needed to publish another authority_hint. And that was it.

We where not able to find a second RP implementation.

Goal of the exercise

During interop event 1 and 2 we had tested and gotten the results according to this matrix:

	Connect2ID	IdentityPython	Shibboleth
Entity Statements	OK	OK	OK
Trust Chain collection	OK	OK	OK
Trust Chain validation	OK	OK	OK
Explicit Client registration	OK	OK	OK
Automatic Client registration	OK	OK	OK

What we now wanted to add understanding and applying metadata policies as described in section 4.1 of the specification. The policy we tested was this one:

```
{
  "openid_provider": {
    "contacts": {
      "add": "operations@feide.no"
    },
    "token_endpoint_auth_methods_supported": {
      "subset_of": ["client_secret_jwt", "private_key_jwt"]
    },
    "token_endpoint_auth_signing_alg_values_supported": {
      "default": ["ES256"],
      "subset_of": ["ES256", "ES384", "ES512"]
    },
    "userinfo_signing_alg_values_supported": {
      "default": ["ES256"],
      "subset_of": ["ES256", "ES384", "ES512"]
    },
    "id_token_signing_alg_values_supported": {
      "default": ["ES256"],
      "subset_of": ["ES256", "ES384", "ES512"]
    }
  },
  "openid_relying_party": {
    "id_token_signed_response_alg": {
      "default": "ES256",
      "one_of": ["ES256", "ES384", "ES512"]
    },
    "userinfo_signed_response_alg": {
      "default": "ES256",
      "one_of": ["ES256", "ES384", "ES512"]
    },
    "token_endpoint_auth_signing_alg": {
      "default": "ES256",
      "one_of": ["ES256", "ES384", "ES512"]
    }
  }
}
```

Basically the FEIDE federation demanded that only Elliptic curve cryptography was to be used. Note that this doesn't influence the federation protocol exchange that is still using RSA crypto.

Results

OP-c2id

We first verified that all the things that worked last time still worked, and they did. When testing one stumble stone was to get the OP to chose the federation we wanted to test. The OP software was designed to chose the shortest trust path but since both where of equal length one was chosen randomly. Once we got passed that doing explicit client registration and metadata policy was shown to work. We failed to get automatic client registration and metadata policy to work. One problem on the RP side for that setup was that the RP needed to apply the metadata policy to the OPs published metadata and then chose a signing algorithm that was supported by the

OPs published keys. Turns out the OP published support for a signing algorithm it was not publishing keys for.

OP-shib

During the second interop event we got the Shibboleth version to do standard explicit and automatic client registration. We where not able to get support for metadata policies to work. We will continue to work on that aspect.

OP-idpy

Explicit and automatic client registration within either of the federations (FEIDE and Swamid) worked as expected.

The OIDC federation specification

During the testing we got time to discuss parts of the specification we felt still needed some work. This lead to a rewrite of section 4.1 metadata policies and also changes to the text describing automatic client registration. These changes are part of version 0.14 which can be found at https://github.com/rohe/oidcfederation .

Next step

I will continue to keep the framework up and running to allow us to reach the goal of the exercise.

I have created a Slack channel (oidcfederation.slack.com) to allow for quick question/ reply feedback.