

TI Incubator cycle 11 half-time demo

05 February 2026



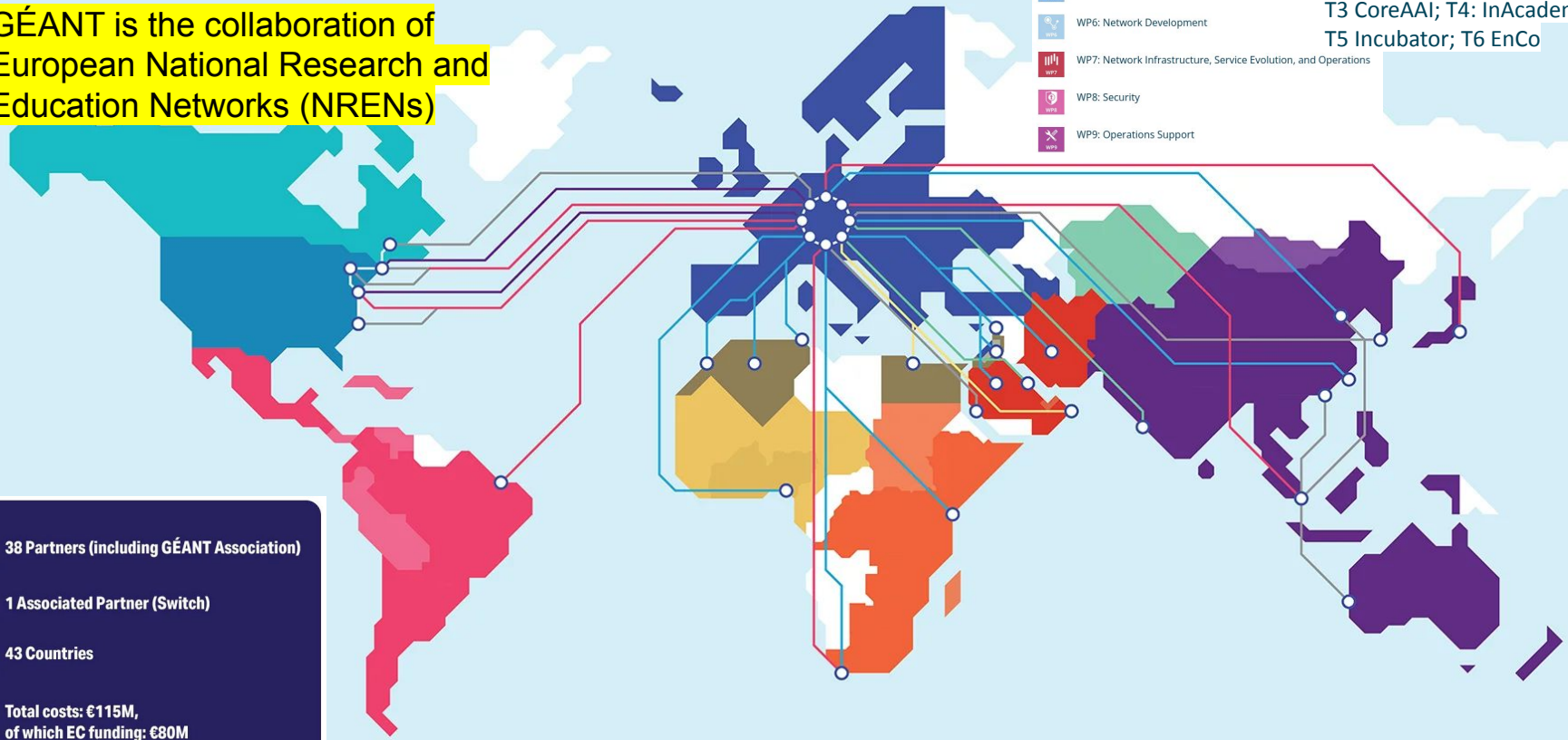


GÉANT is the collaboration of European National Research and Education Networks (NRENs)

GNS-2 is arranged around nine Work Packages covering 42 Tasks:

- WP1: Project Management
- WP2: Marketing Communications, Events, and Policy Engagement
- WP3: User and Stakeholder Engagement
- WP4: Above-the-Net Services
- WP5: Trust & Identity
- WP6: Network Development
- WP7: Network Infrastructure, Service Evolution, and Operations
- WP8: Security
- WP9: Operations Support

WP5
T1: EDUROAM; T2 eduGAIN;
T3 CoreAAI; T4: InAcademia;
T5 Incubator; T6 EnCo



- 38 Partners (including GÉANT Association)
- 1 Associated Partner (Switch)
- 43 Countries
- Total costs: €115M, of which EC funding: €80M

Introductions

TI Team

- Alexandr Petrunin
- Alexandru Cacean
- Andrijana Todosijevic
- Amin Mahnamfar
- Amineh Akhavan Saraf
- András Bodor
- Boro Jakimovski
- Febri Kazazi
- Gabriel Zachmann (on parental leave)
- Halil Adem
- Harm Roukema
- Janne Lauros
- Jovan Simonoski
- Kushal Das
- Marko Ivančić
- Mihály Héder
- *Niels van Dijk*
- Peter Bolha

WP5 Leads

- *Irina Sidorova*
- *Maarten Kremers*
- *Marina Adomeit (on parental leave)*

Now running: cycle 11:
October 25-May 26

Some topics in cycles (2019-present)

1. Trust Marks -> SIRTIFI+JAGGER
SeamlessAccess trials at GEANT
Cryptech HSM
2. eduGAIN status reporting
Push MDQ
3. eduLNK
4. SSP OIDC OP
OIDC agent for windows
DID for researchers
5. DID+EBSI+IRMA wallet
6. SSH certificates on federated login
IdP user profile
7. geteduroam linux client
SSP OI DFed
8. Scalable testing for Insecure XMLsig
Trust fabric for wallets ARF
9. eduGAIN OI DFed PoC
OIFED in Shib and SSP (part2)
10. GO OP, TA, RP implementation
SeamlessAccess OI DFed PoC
VC issuer IDPs
Bona Fide researcher vetting



Past cycles on
the dashboard:



TIM NEEDS

YOU



Géant
Innovation
Programme





Sweet spot: reducing risk with limited cost

Cost

TRL

Thinking over a beer -
Research on paper
(low cost)

Proof of concept
(increased cost)

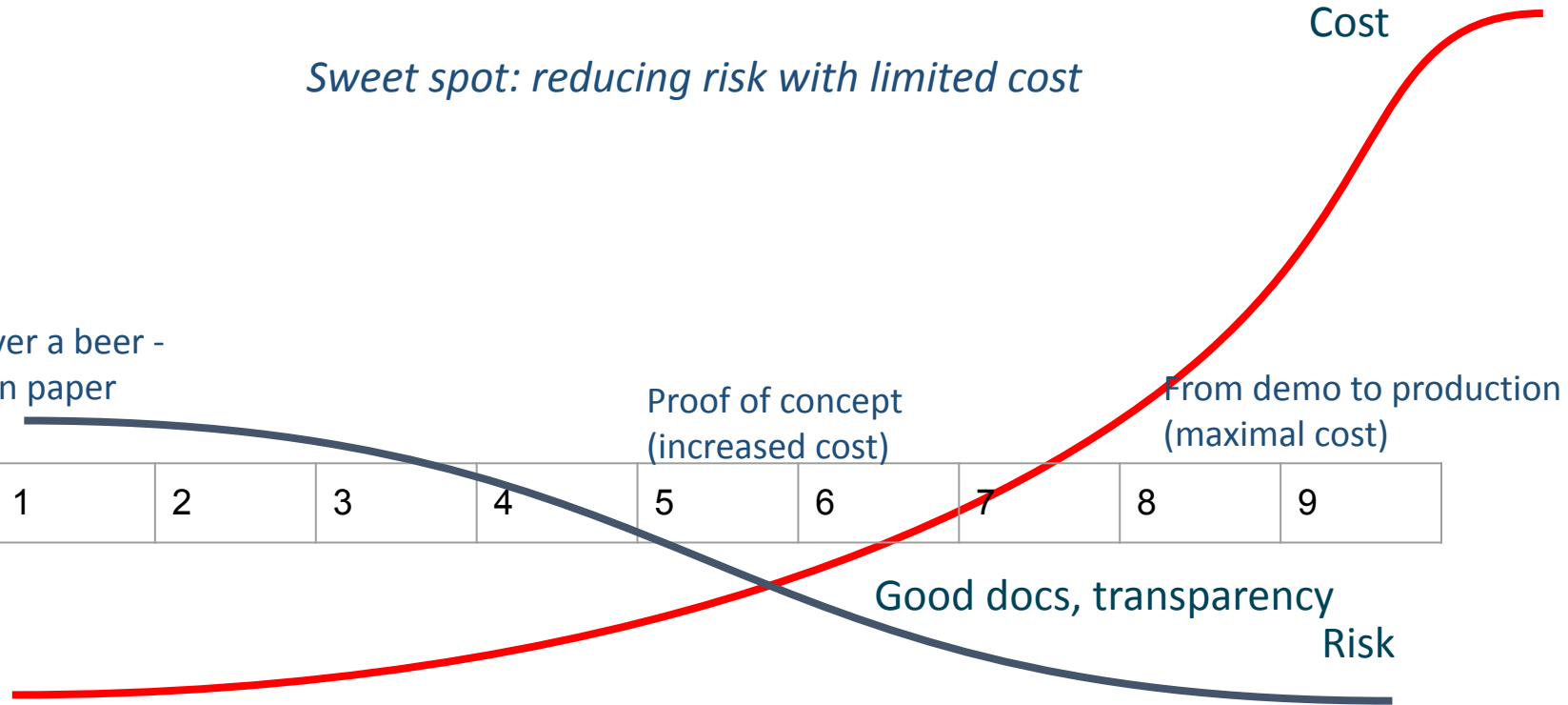
From demo to production
(maximal cost)



Good docs, transparency

Risk

Risk



Shibboleth and SSP Verifiable Credential Issuers' DIIP compliance

Janne Lauros
Marko Ivancic
Mihály Héder



Previous cycle was about having Shibboleth IdP and SimpleSAMLPHP issue credential to wallet - any credential to any wallet and by any means.

This cycle we aim for interoperability conformance.

Our named goal is DIIP, <https://fidescommunity.github.io/DIIP>

There is also HAIP,

https://openid.net/specs/openid4vc-high-assurance-interoperability-profile-1_0.html

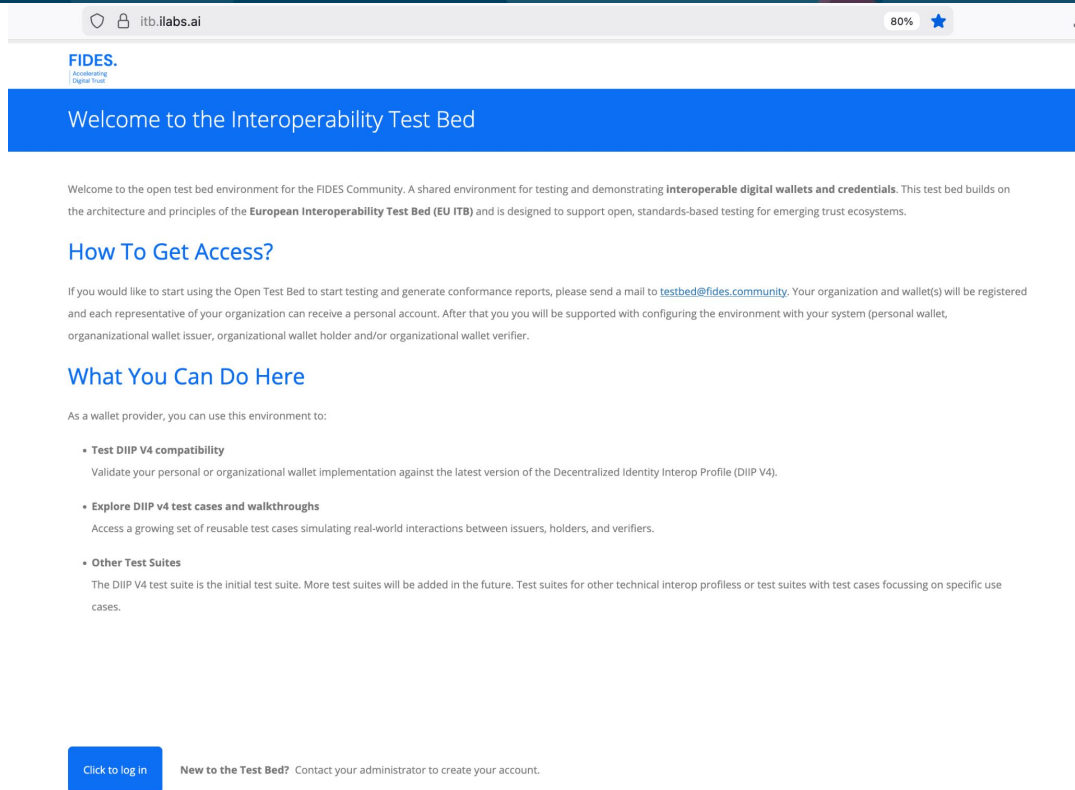
Full conformance will not be reached in this cycle but we are closing

Conformance Tools - DIIP <https://itb.ilabs.ai/>

Should be our main tool for running interoperability tests

Has only one test for Issuer (that I am able to locate)

We pass that test



The screenshot shows a web browser window with the address bar displaying "itb.ilabs.ai". The page header includes the FIDES logo with the tagline "Accelerating Digital Trust". A blue banner at the top of the page reads "Welcome to the Interoperability Test Bed". Below this, a welcome message states: "Welcome to the open test bed environment for the FIDES Community. A shared environment for testing and demonstrating **interoperable digital wallets and credentials**. This test bed builds on the architecture and principles of the **European Interoperability Test Bed (EU ITB)** and is designed to support open, standards-based testing for emerging trust ecosystems." A section titled "How To Get Access?" explains that users should email testbed@fides.community to get started. Another section, "What You Can Do Here", lists three main activities: "Test DIIP V4 compatibility" (validating wallet implementations), "Explore DIIP v4 test cases and walkthroughs" (accessing reusable test cases), and "Other Test Suites" (initial test suite with more to be added). At the bottom, there is a "Click to log in" button and a link for "New to the Test Bed?" which directs users to contact their administrator.

itb.ilabs.ai 80% ★

FIDES.
Accelerating
Digital Trust

Welcome to the Interoperability Test Bed

Welcome to the open test bed environment for the FIDES Community. A shared environment for testing and demonstrating **interoperable digital wallets and credentials**. This test bed builds on the architecture and principles of the **European Interoperability Test Bed (EU ITB)** and is designed to support open, standards-based testing for emerging trust ecosystems.

How To Get Access?

If you would like to start using the Open Test Bed to start testing and generate conformance reports, please send a mail to testbed@fides.community. Your organization and wallet(s) will be registered and each representative of your organization can receive a personal account. After that you will be supported with configuring the environment with your system (personal wallet, organizational wallet issuer, organizational wallet holder and/or organizational wallet verifier).

What You Can Do Here

As a wallet provider, you can use this environment to:

- **Test DIIP V4 compatibility**
Validate your personal or organizational wallet implementation against the latest version of the Decentralized Identity Interop Profile (DIIP V4).
- **Explore DIIP v4 test cases and walkthroughs**
Access a growing set of reusable test cases simulating real-world interactions between issuers, holders, and verifiers.
- **Other Test Suites**
The DIIP V4 test suite is the initial test suite. More test suites will be added in the future. Test suites for other technical interop profiles or test suites with test cases focussing on specific use cases.

[Click to log in](#) [New to the Test Bed?](#) Contact your administrator to create your account.

<https://itb.ilabs.ai/> - pre-authorized flow without tx code

The screenshot shows a web browser window with the address bar displaying "itb.ilabs.ai". The page content includes the FIDES logo, a blue header with the text "Welcome to the Interoperability Test Bed", and several sections of text and links. At the bottom, there is a "Click to log in" button and a link for new users to contact their administrator.

FIDES.
Interoperability
Experimentation

Welcome to the Interoperability Test Bed

Welcome to the open test bed environment for the FIDES Community. A shared environment for testing and demonstrating **interoperable digital wallets and credentials**. This test bed builds on the architecture and principles of the **European Interoperability Test Bed (EU ITB)** and is designed to support open, standards-based testing for emerging trust ecosystems.

How To Get Access?

If you would like to start using the Open Test Bed to start testing and generate conformance reports, please send a mail to testbed@fides.community. Your organization and wallet(s) will be registered and each representative of your organization can receive a personal account. After that you will be supported with configuring the environment with your system (personal wallet, organizational wallet issuer, organizational wallet holder and/or organizational wallet verifier).

What You Can Do Here

As a wallet provider, you can use this environment to:

- **Test DIIP V4 compatibility**
Validate your personal or organizational wallet implementation against the latest version of the Decentralized Identity Interop Profile (DIIP V4).
- **Explore DIIP v4 test cases and walkthroughs**
Access a growing set of reusable test cases simulating real-world interactions between issuers, holders, and verifiers.
- **Other Test Suites**
The DIIP V4 test suite is the initial test suite. More test suites will be added in the future. Test suites for other technical interop profiles or test suites with test cases focussing on specific use cases.

[Click to log in](#) [New to the Test Bed?](#) Contact your administrator to create your account.

Privacy note: By continuing you consent to the use of cookies to manage your session.

Define a credential as per

https://openid.net/specs/openid-4-verifiable-credential-issuance-1_0.html#name-credential-issuer-metadata `#credential_configurations_supported`

The definition is used by:

Shibboleth itself as a instruction what credentials it supports and the specifics of them like format.

Wallets as they read Issuer configuration for which credentials that are being offered have suitable specifics like format.

`/opt/shibboleth-idp/metadata/verifiable-credentials.json`

The party that is allowed to initiate pre-authorized flow i.e. generate the url/qr code is considered in this flow the actual client

```
openid-credential-offer://?credential_offer={"grants":{"urn:ietf:params:oauth:grant-type:pre-authorized_code":{"pre-authorized_code":"AAD..6dw"}}, "credential_issuer":"https://geant-vci.2.rahtiapp.fi", "credential_configuration_ids":["GeantIncubatorDiploma"]}
```

Wallet not presenting any client identifier is referred with identity of the party that initiated the flow

/opt/shibboleth-idp/conf/[relying-party.xml](#)

Conformance Tools - OpenID

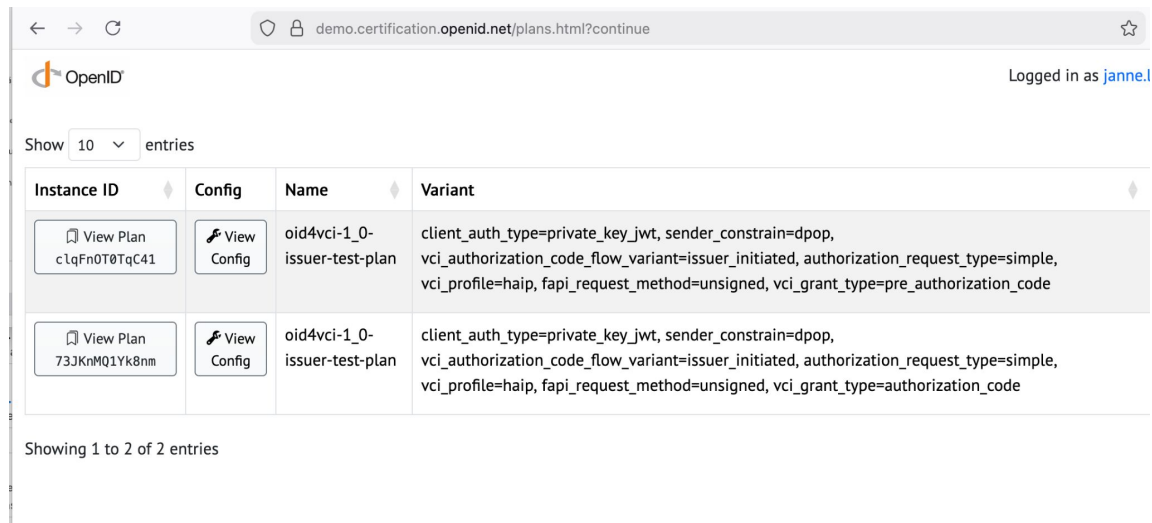
<https://demo.certification.openid.net/login.html>

Also still work in progress

Only “happy flow” cases exist for both pre-authorized and code flows

We pass both tests!

Excellent tooling even in this draft stage!



The screenshot shows a web browser window with the URL `demo.certification.openid.net/plans.html?continue`. The page displays the OpenID logo and a user logged in as `janne.l`. Below the header, there is a table with 4 columns: Instance ID, Config, Name, and Variant. The table contains 2 entries. Each entry has a 'View Plan' button and a 'View Config' button. The variant text for both entries is: `client_auth_type=private_key_jwt, sender_constrain=dpop, vci_authorization_code_flow_variant=issuer_initiated, authorization_request_type=simple, vci_profile=haip, fapi_request_method=unsigned, vci_grant_type=pre_authorization_code`.

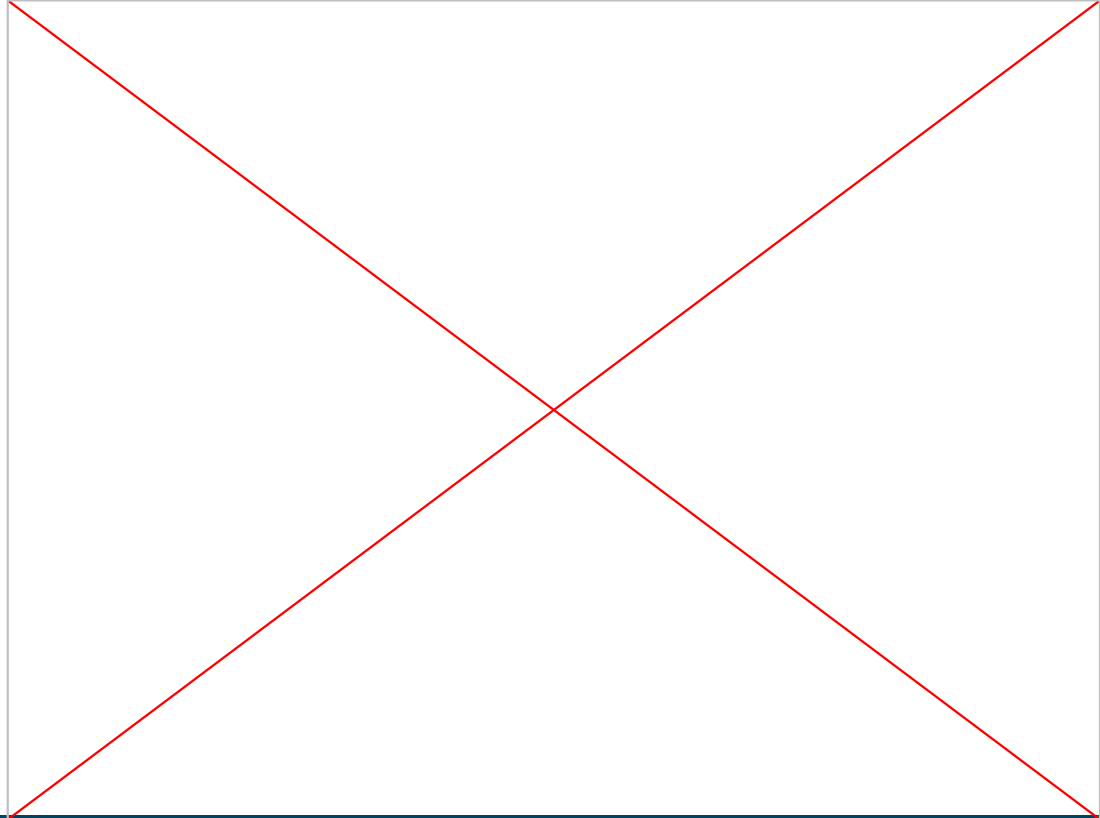
Instance ID	Config	Name	Variant
View Plan cLqFn0T0TqC41	View Config	oid4vci-1_0-issuer-test-plan	client_auth_type=private_key_jwt, sender_constrain=dpop, vci_authorization_code_flow_variant=issuer_initiated, authorization_request_type=simple, vci_profile=haip, fapi_request_method=unsigned, vci_grant_type=pre_authorization_code
View Plan 73JKnM01Yk8nm	View Config	oid4vci-1_0-issuer-test-plan	client_auth_type=private_key_jwt, sender_constrain=dpop, vci_authorization_code_flow_variant=issuer_initiated, authorization_request_type=simple, vci_profile=haip, fapi_request_method=unsigned, vci_grant_type=authorization_code

Showing 1 to 2 of 2 entries

<https://demo.certification.openid.net/login.html> - pre-authorized flow

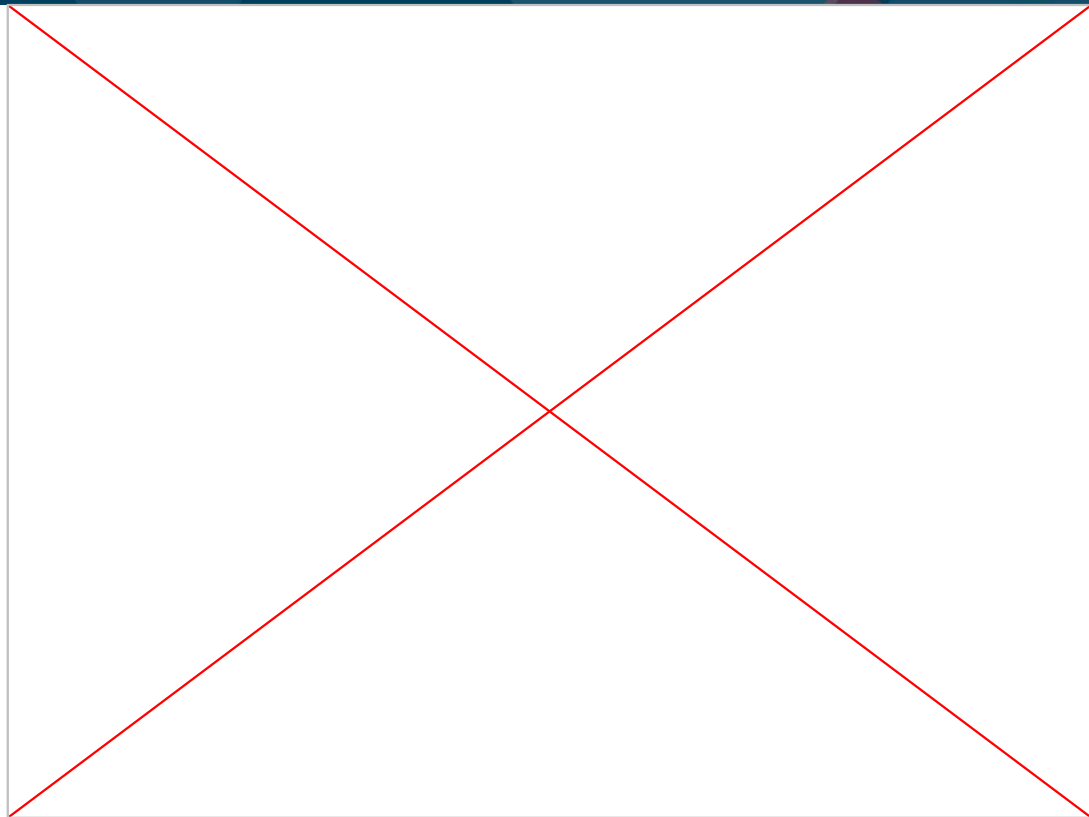
We run here all available pre-authorization flow tests.

Excellent tooling! Notice the number of specific checks and explanations.



We run here all available code flow tests.

Still Excellent tooling! I wish it had already the tricky tests that try to fool my implementation.



“Normal” known relying parties vs. DIIP case. No inherited client identifiers.
DPOP configured, PAR configured for Code flow.

`/opt/shibboleth-idp/conf/relying-party.xml`

Shib Admin Task - Code flow requires attribute, for now like this:

```
<AttributeDefinition id="GeantIncubatorDiploma" xsi:type="ScriptedAttribute">
  <InputAttributeDefinition ref="mail" />
  <InputAttributeDefinition ref="eduPersonPrincipalName" />
  <InputAttributeDefinition ref="givenName" />
  <InputAttributeDefinition ref="sn" />
  <InputAttributeDefinition ref="eduPersonAffiliation" />
  <Script><![CDATA[
    var credential = '{"requestedCredential":[\
      {"path":["diploma"],"value":"Scrum Master"},\
      {"path":["mail"],"value":"' +mail.getValues().get(0)+'"},\
      {"path":["eppn"],"value":"' +eduPersonPrincipalName.getValues().get(0)+'"},\
      {"path":["givenName"],"value":"' +givenName.getValues().get(0)+'"},\
      {"path":["familyName"],"value":"' +sn.getValues().get(0)+'"},\
      {"path":["affiliation"],"value":"' +eduPersonAffiliation.getValues().get(0)+'"}]\
    }';
    GeantIncubatorDiploma.addValue(credential.replaceAll("\\s+", ""));
  ]]></Script>
  <AttributeEncoder xsi:type="oidc:OIDCString" name="GeantIncubatorDiploma"/>
</AttributeDefinition>
```

Compare to `/opt/shibboleth-idp/metadata/verifiable-credentials.json`

Features still missing to meet DIIP (v4) on paper.

- “did:jwk” and “did:web” identifiers, analysis what responsibilities comes to Issuer from them and possible implementations. Undersigned has no clue yet.
- IETF Token Status List as token status list mechanism i.e. revoking credentials. Undersigned has no clue yet.
- Some other more minor details.

*To reach maturity have first public version would require still a lot more than merely implement the features above. There are missing validations, not every corner case has been implemented and configurations need lot of work still. What is good we seem to be able to write credentials to several wallets without much pain. **Luckily Incubator is about incubating and not birthing.***

Fuzzing SSO implementations (TIM student topic)

Harm Roukema
Amineh Akhavan Saraf



- Why look into security of SSO implementations?
 - Protocols are secure, but can be complex to implement
 - Bugs can lead to severe vulnerabilities like account takeover
- Fuzzing: automatically randomize inputs to uncover bugs
- Focus on OIDC
 - SimpleSAMLphp OIDC module
- Later: Shibboleth, possibly SAML

- Example OIDC authentication request:

```
https://server.example.org/authorize?response_type=code&scope=openid%20profile%20email&client_id=s6BhdRkqt3&state=af0ifjsldkj&redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
```

- Example OIDC authentication response:

```
HTTP/1.1 302 Found  
Location: https://client.example.org/cb?  
code=Sp1x10BeZQQYbYS6WxSbIA  
&state=af0ifjsldkj
```

Example: redirect_uri open redirect

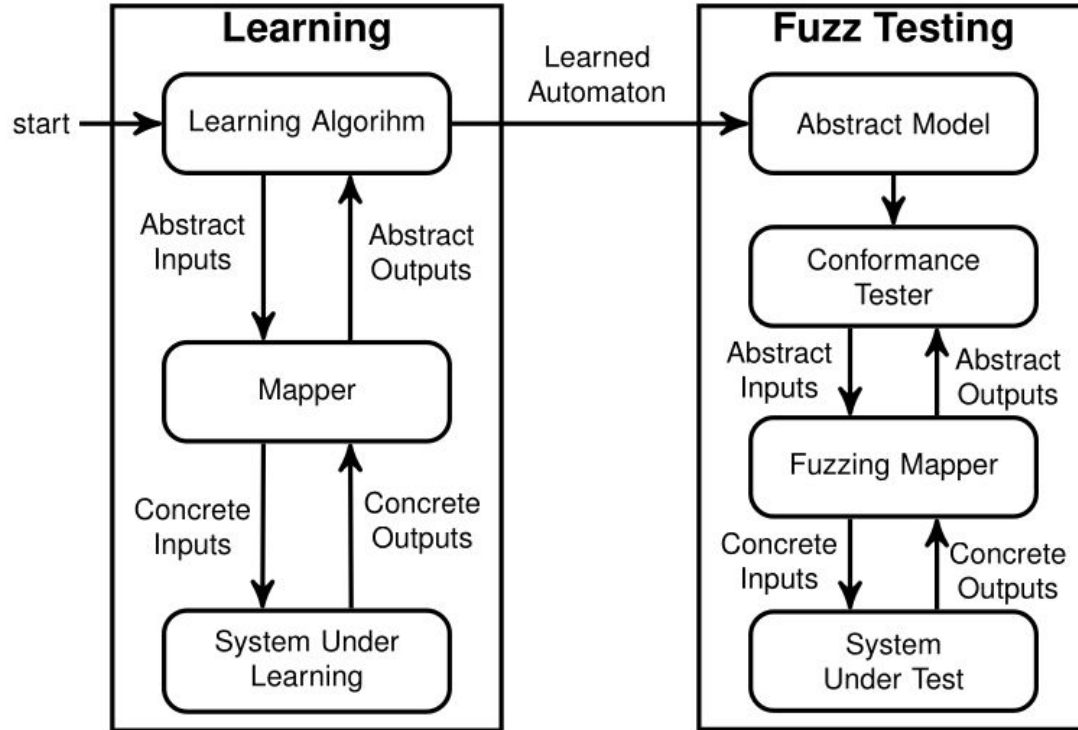
- Example implementation error:

```
if redirect_uri.startswith("https://client.example.org"):  
    ...
```

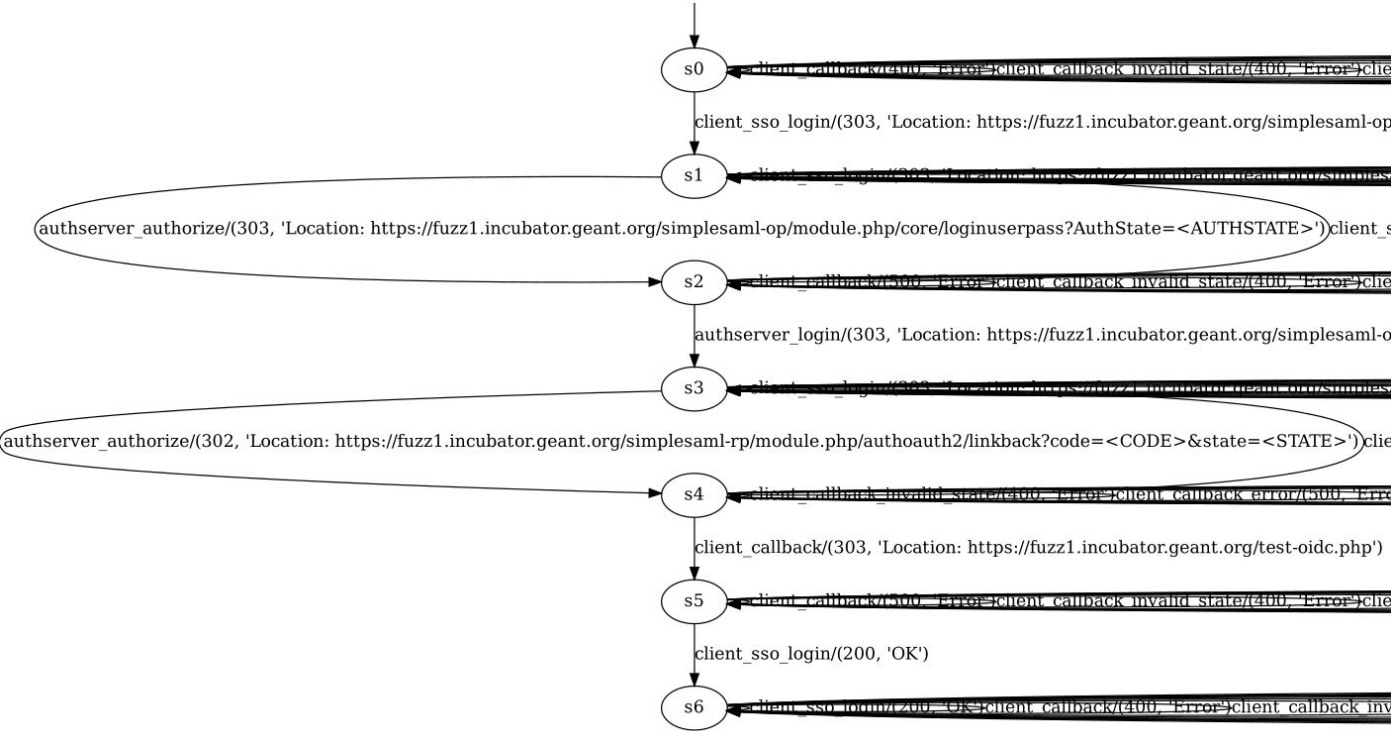
- Bypass:

```
https://server.example.org/authorize?response_type=code&scope=openid%20profile%20email&client_id=s6BhdRkqt3&state=af0ifjsldkj&redirect_uri=https%3A%2F%2Fclient.example.org.evil.com%2Fcb
```

- Impact: one-click account takeover (authorization code is sent to evil.com!)



SimpleSAMLphp OIDC authorization code flow



- Examine learnt state machines (manually)
 - Differences between SimpleSAMLphp and Shibboleth?
- Fuzz using state machines
 - Random walks through states with fuzzed inputs
- Fuzzing parameters: define properties of parameters [2]
 - Constant/variable, mandatory, used once or multiple, user-specific, session-specific
 - E.g. `state`: session-specific -> get new session and substitute
 - URLs (`redirect_uri`): use known specific tricks, e.g. redirect-fuzzer [3]

- Uses previously learned state machine
- Walks through state machine using fuzzed inputs
- Finds **difference in outputs** between fuzzed inputs and expected behavior:
 - Redirect to
`https://client1-oauth-ctf.harmroukema.nl.evil.com`
`?code=<CODE>&state=<STATE>`
- Shows concrete traces for reproducibility

- Implement more fuzzing techniques and fuzz SSP OIDC
 - JWT fuzzing
 - Learn more flows
- Fuzz Shibboleth
 - Compare state machine with SSP OIDC

Stretch goals:

- Fuzz SAML implementations

- [1] Bernhard K. Aichernig, Edi Muškardin, and Andrea Pferscher. Learning-based Fuzzing of IoT Message Brokers.
- [2] Ronghai Yang, Guanchen Li, Wing Cheong Lau, Kehuan Zhang, and Pili Hu. Model-based security testing: An empirical study on OAuth 2.0 implementations.
- [3] Wang, X., Lau, W. C., Yang, R., & Shi, S. (2019). Make redirection evil again: URL parser issues in OAuth.

Accessibility testing of wallet ecosystem

Febri Kazazi
Esther Ruiz Ben



- Departing from **DUxDI-Project's results** (GÉANT Innovation Programme 2024/25 - University of Málaga and DFN) related to Accessibility of Digital Wallets in education and research contexts.
- Target in T&I Incubator cycle 11:
 - **Improve accessibility of digital wallets prototypes for education and research.**
- Methods currently used:
 - **Tests of digital wallets with available tools, including those AI-based** (i. e. WAVE, Lighthouse, AInspector). Considering selected scenarios (Wallet Registration, VC Issuance, VC Presentation, Selective Disclosure, Interoperability with assistive tools such as NVDA).
- Methods to be further applied: **digital wallets UX-accessibility tests by persons with disabilities** active in education and research contexts.

Lighthouse report



Accessibility

Desktop score



Accessibility

Mobile score

WAVE tool report

AIM Score: 4.2 out of 10



Lighthouse report



Performance

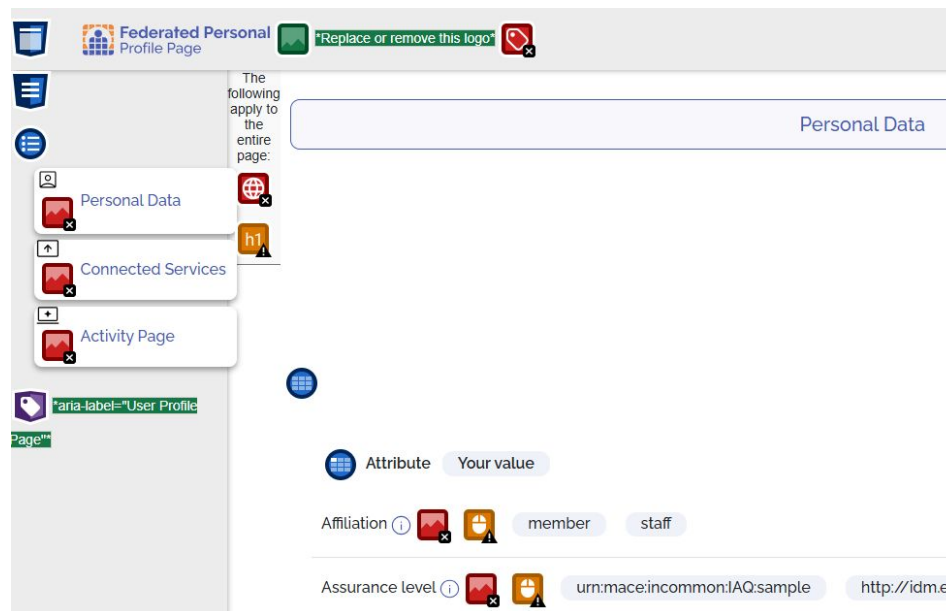
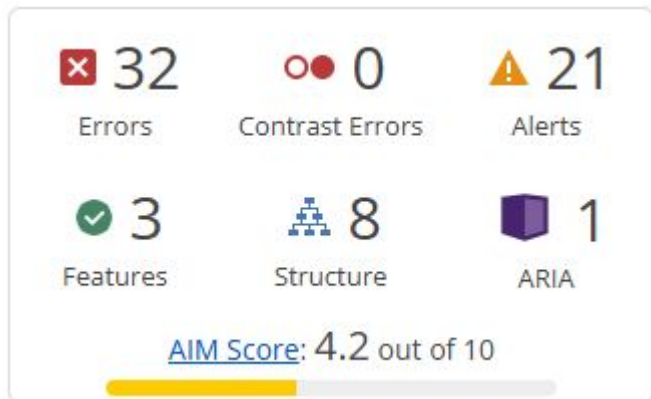
Desktop score



Performance

Mobile score

WAVE tool report



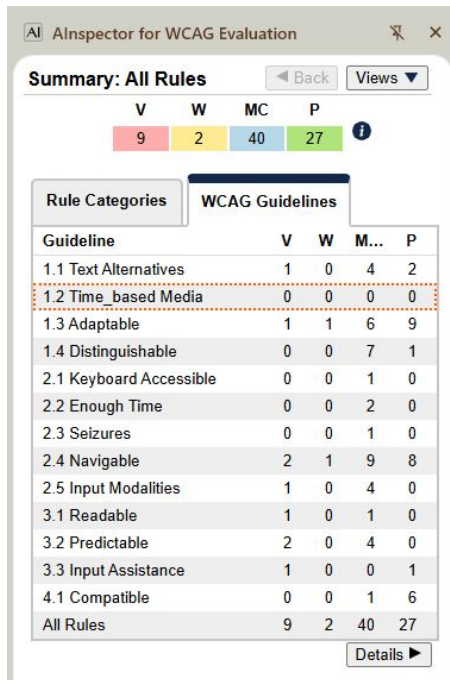
The screenshot shows a "Federated Personal Profile Page" with several accessibility issues highlighted:

- "Replace or remove this logo" (red X icon)
- "The following apply to the entire page:" (text box)
- "Personal Data" (red X icon)
- "Connected Services" (red X icon)
- "Activity Page" (red X icon)
- "aria-label='User Profile'" (green checkmark icon)
- "Page!" (green checkmark icon)

The page content includes a "Personal Data" section with an "Attribute" table:

Attribute	Your value
Affiliation	member staff
Assurance level	urn:mace:incommon:IAQ:sample http://idme

University of Illinois AInspector report



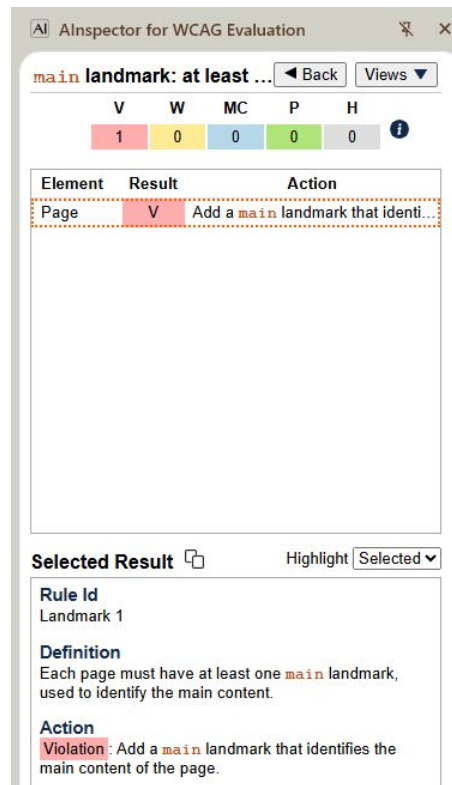
Summary: All Rules

V	W	MC	P
9	2	40	27

Rule Categories: WCAG Guidelines

Guideline	V	W	M...	P
1.1 Text Alternatives	1	0	4	2
1.2 Time_based Media	0	0	0	0
1.3 Adaptable	1	1	6	9
1.4 Distinguishable	0	0	7	1
2.1 Keyboard Accessible	0	0	1	0
2.2 Enough Time	0	0	2	0
2.3 Seizures	0	0	1	0
2.4 Navigable	2	1	9	8
2.5 Input Modalities	1	0	4	0
3.1 Readable	1	0	1	0
3.2 Predictable	2	0	4	0
3.3 Input Assistance	1	0	0	1
4.1 Compatible	0	0	1	6
All Rules	9	2	40	27

Details ▶



main landmark: at least ...

V	W	MC	P	H
1	0	0	0	0

Element	Result	Action
Page	V	Add a main landmark that identi...

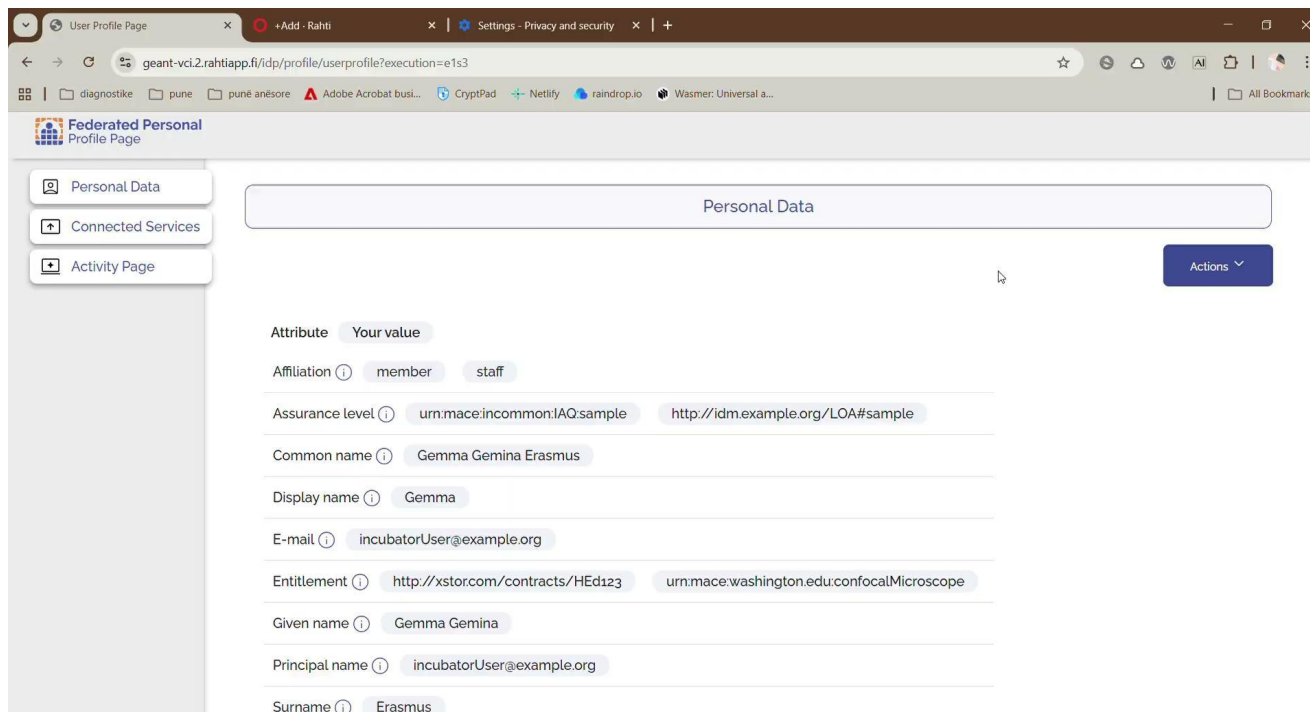
Selected Result

Rule Id
Landmark 1

Definition
Each page must have at least one main landmark, used to identify the main content.

Action
Violation: Add a main landmark that identifies the main content of the page.

NVDA screen reader testing



The screenshot displays a web browser window with the following details:

- Browser Tabs:** User Profile Page, +Add - Rahti, Settings - Privacy and security.
- Address Bar:** geant-vci.2.rahtiapp.fi/idp/profile/userprofile?execution=e1s3
- Page Title:** Federated Personal Profile Page
- Sidebar Menu:** Personal Data (selected), Connected Services, Activity Page.
- Main Content Area:**
 - Personal Data** (Section Header)
 - Actions** (Dropdown Button)
 - Attribute:** Your value
 - Affiliation:** member, staff
 - Assurance level:** urn:mace:incommon:IAQ:sample, http://idm.example.org/LOA#sample
 - Common name:** Gemma Gemina Erasmus
 - Display name:** Gemma
 - E-mail:** incubatorUser@example.org
 - Entitlement:** http://xstor.com/contracts/HEd123, urn:mace:washington.edu:confocalMicroscope
 - Given name:** Gemma Gemina
 - Principal name:** incubatorUser@example.org
 - Surname:** Erasmus

Issues

Severe

- Image elements do not have [alt] attributes.
- <html> element does not have a [lang] attribute.
- Document does not have a main landmark.
- Missing [aria*] attributes.
- Missing and/or empty form labels.
- Relying on device dependent event handlers only.

Light

- Incorrect navigation order.
- Improper use of HTML semantics.

- Apply the fixes to Shibboleth and SimpleSAMLphp wallets.
- Perform another quality test to identify any other.
- Run a “user flow” simulation with people with disabilities.
- Smartphone APP testing - how?

OIDFed topics



OpenID Federation topics:

- PHP OIDFed library and RP
- Local discovery for OIDFed
- Apache mod_oidfed (TIM)
- OpenID Federation Registry

PHP OI DFed library and RP

Marko Ivančić
Peter Bolha
Jovan Simonoski



- PHP OpenID Library
 - <https://github.com/simplesamlphp/openid>
 - Contains various abstractions / tools related to OpenID specs
 - Related to Federation spec: has tools to build Entity Statements, resolve Trust Chains, fetch and Validate Trust Marks... with auto-cache-until-expiration feature
- Generic Relying Party in PHP with federated capabilities
 - <https://github.com/cicnavi/oidc-client-php/tree/v3.x>
 - Proof-of-concept generic PHP RP with federated capabilities
 - Automatic client registration and authorization code flow
 - Uses [simplesamlphp/openid](https://github.com/simplesamlphp/openid) tools under the hood
 - Demo app: <https://fed-rp.mivanci.incubator.hexaa.eu/>

- SimpleSAMLphp OP module (`oidc`)
 - OAuth 2.0 Token Introspection (Jovan, Peter)
 - Support for defining and using multiple signature algorithms and keys (prepared for v7 which is WIP)
- Next steps
 - Federation capabilities for SimpleSAMLphp RP module
<https://github.com/cirrusidentity/simplesamlphp-module-uthoauth2>
 - SimpleSAMLphp as a Trust Anchor / Intermediate :
<https://github.com/simplesamlphp/simplesamlphp-module-oidanchor>
 - If we have time:
 - Pushed Authorization with automatic client registration
 - Dynamic / Explicit client registration

PHP Federated RP

This is a sample demo showcasing PHP Federated RP capabilities.

OIDC Client PHP repo URL: <https://github.com/cicnavi/oidc-client-php>

Configuration Overview

Entity Configuration: [/_well-known/openid-configuration/ \(JSON\)](#)

Other options (that are not visible in Entity Configuration):

- Trust Anchors:

```
array (  
  0 => 'https://oidfed-ta-demo.incubator.geant.org',  
)
```
- Entity Statement Duration:
`'00mon 01day 00hou 00min 00sec'`
- Max Cache Duration:
`'00mon 00day 06hou 00min 00sec'`
- Timestamp Validation Leeway:
`'00mon 00day 00hou 01min 00sec'`
- Max Trust Chain Depth:
`9`
- Default Trust Mark Status Endpoint Usage Policy:
`'RequiredIfEndpointProvidedForNonExpiringTrustMarksOnly'`
- Include Software ID:

Authentication

Login

Local discovery for OI DFed

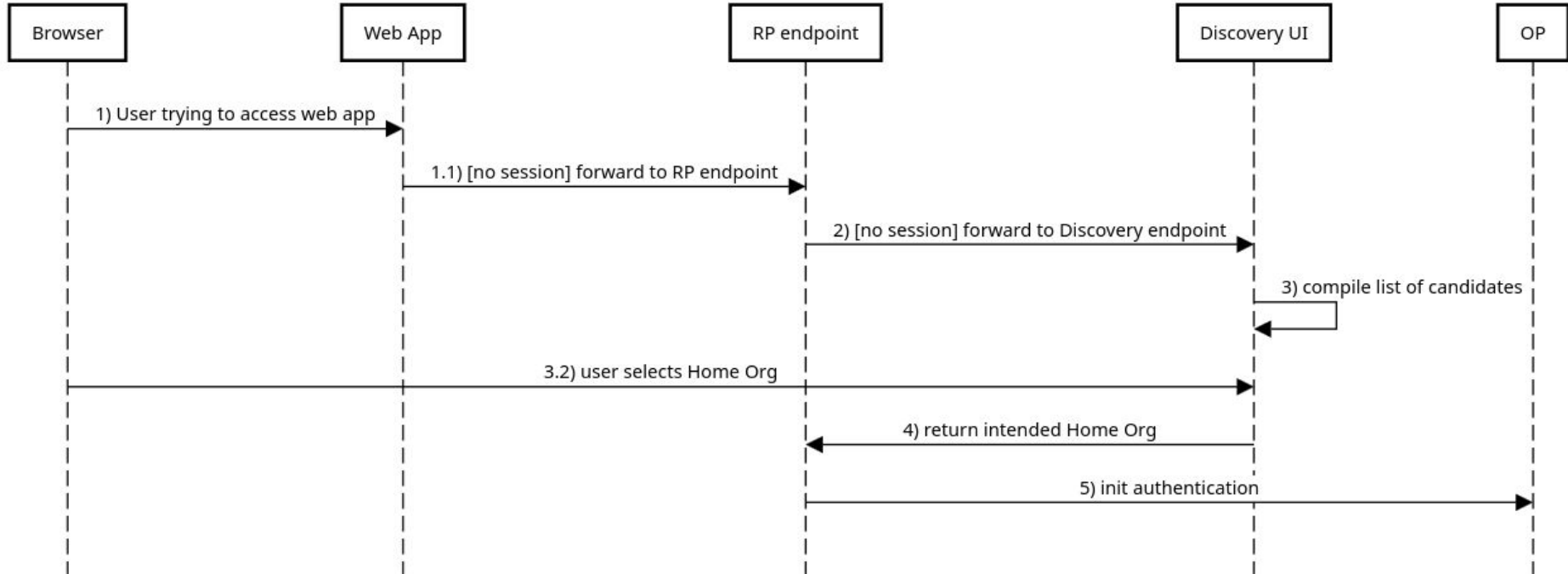
Mihály Héder



- Motivation
 - In a typical RP deployment, OP selection is the first GUI element the user experiences
 - Often this is the only visible part of the RP stack itself
 - Deployers will want to have this branded & nice UX
 - Deployers will want their own caching and even business logic
 - Separation of concern, modularisation
 - Let's not make this the burden of the RP implementer necessarily
 - Let's use the **same flow for centralized discovery** (like SeamlessAccess, as seen in cycle 10), and **local**
- **More on that at TIIME 13 Friday at Discovery section**

Suggested flow

Discovery



Apache mod_oidfed (TIM student topic)

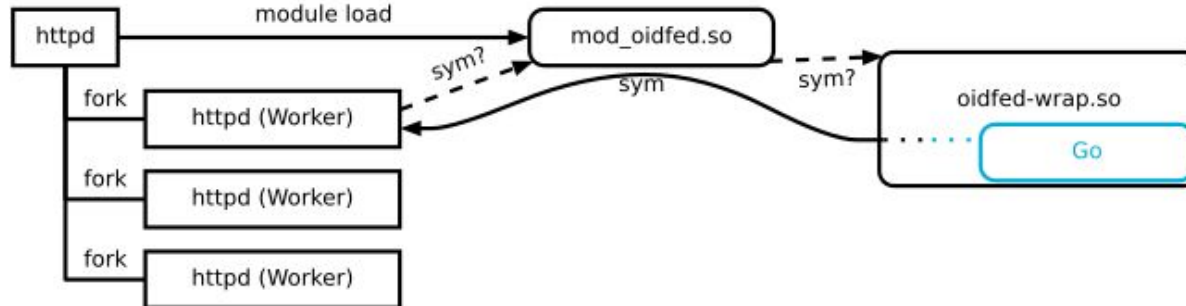
András Bodor



- Different approach from auth_memcookie (was demoed: Cycle 10)
 - Apache httpd is widely used
 - Reverse proxy, HTTP server, etc.
 - Built-in module to act as RP in a federation
 - Anything behind httpd can implicitly enter the federation
- => mod_oidfed

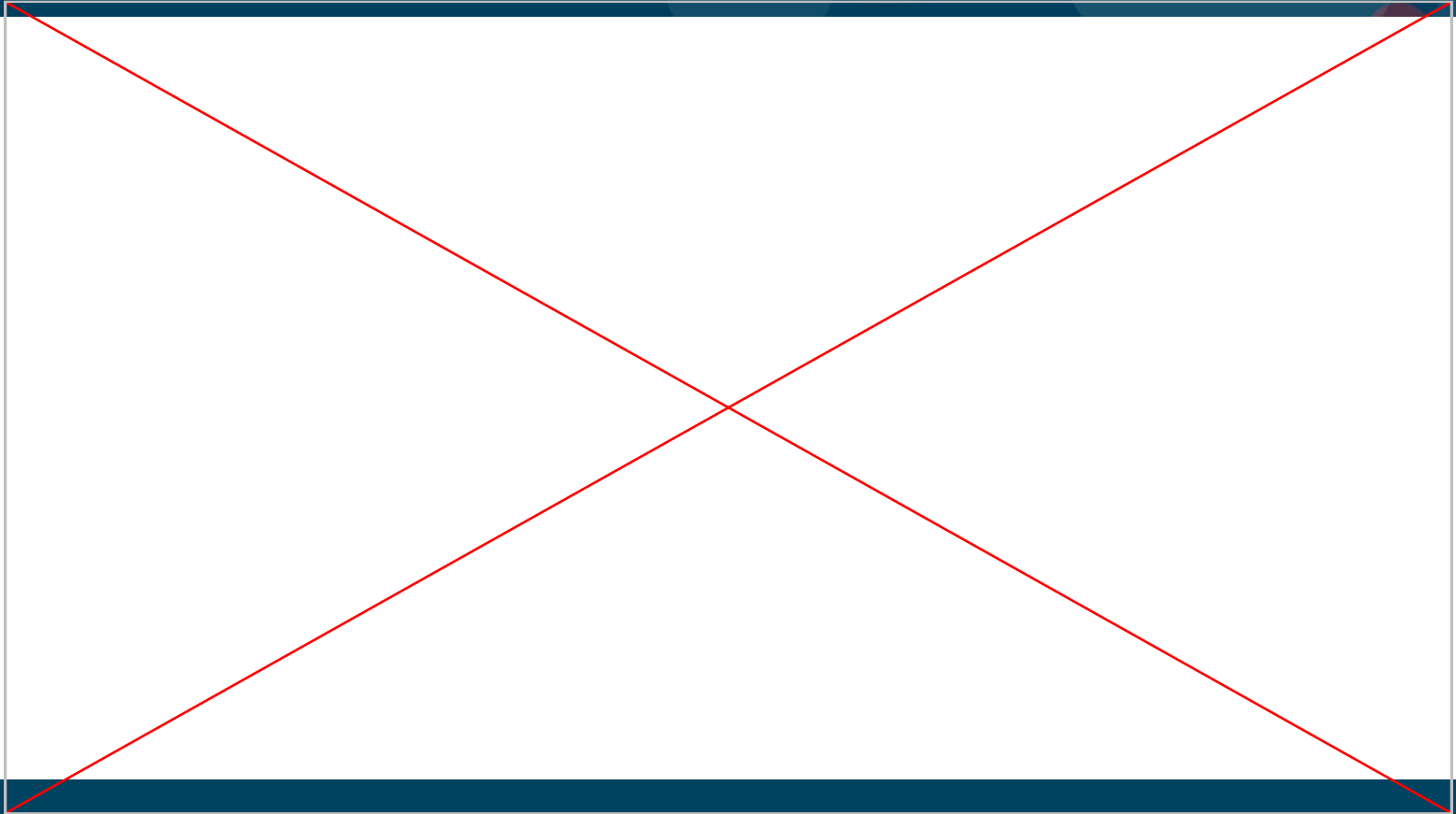
- Wrapping the existing Go library
 - Allows reusing of existing federation logic
 - Considerably faster development
 - Compatibility for free
 - Slightly larger resource usage than pure-native (~15 MiB/Worker)
 - Have to be especially careful with runtime clashes (Go vs Apache)

=> Each worker manually handles symbol resolution



- Federation entity configuration from Apache httpd.conf
 - e.g. OidfedEntityId <https://rp.example.com>
- /.well-known/openid-federation endpoint
 - Module automatically registers and handles this path
- Apache authn module integration
- *Work in progress:*
 - *Login page (OP selection)*
- *Demo!*

Apache mod_oidfed - Demo



OpenID Federation Registry

Gabriel Zachmann

Amin Mahnamfar

Niels van Dijk

Alexandr Petrunin

Halil Adem

Kushal Das



- OpenAPI spec:
https://gitlab.software.geant.org/TI_Incubator/federation-admin-api/-/blob/main/federation_admin_openapi.yaml?ref_type=heads
 - Fairly complete, but not considered final
- The API covers relevant actions a federation administrator wants to do.
- Usage:
 - [nerds] from the command line (e.g. curl)
 - Swagger UI
 - Proper frontend

- Management of the Federation Authority (itself)
 - Authority Hints
 - Management of Signing Keys
 - Metadata
 - Published Trust Marks
 - Allowed Trust Marks, Trust Mark Issuers & Trust Mark Owners
 - Additional Entity Configuration Claims
 - Entity Configuration Lifetime
- Subordinates
 - General and individual Metadata Policies
 - General and individual Constraints
 - General and individual additional claims for subordinate statements
 - Management of Subordinate's JWKS
 - Metadata
 - Lifecycle Management of Subordinates
- Trust Mark Issuance

- New “major” version of LightHouse
 - Major changes
 - Version jump to 0.20.0
- SQL Databases instead of document storage
- HTTP API to manage LightHouse
- New capabilities, like individual Metadata Policies per entity
- There will be breaking changes, but a smooth upgrade path will be provided
 - Config file migration
 - DB migration

- Management of the Federation Authority (itself)
 - Authority Hints
 - Management of Signing Keys
 - Metadata
 - Published Trust Marks
 - Allowed Trust Marks, Trust Mark Issuers & Trust Mark Owners
 - Additional Entity Configuration Claims
 - Entity Configuration Lifetime
- Subordinates
 - General and individual Metadata Policies
 - General and individual Constraints
 - General and individual additional claims for subordinate statements
 - Management of Subordinate's JWKS
 - Metadata
 - Lifecycle Management of Subordinates
- Trust Mark Issuance

Done

TODO

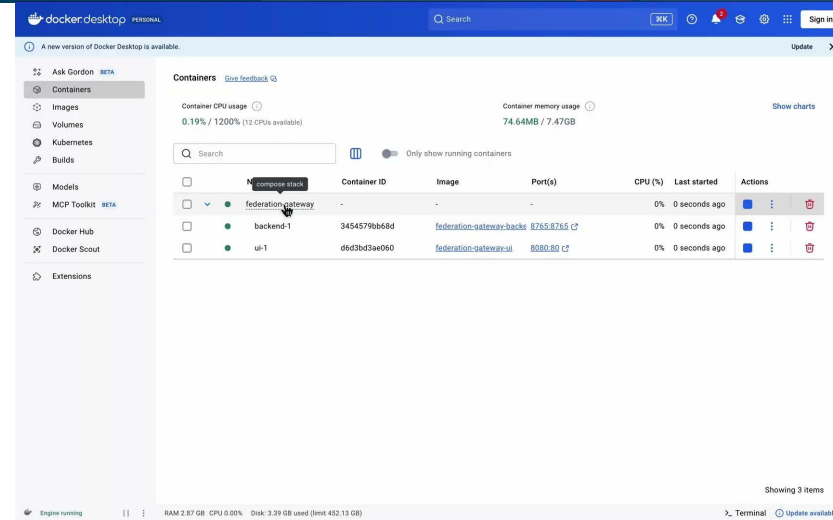
Ongoing

Usability prototype:

- Trust Anchors & Intermediates - CRUD
 - National level
 - Superior level
 - Subordinate level
- Leaf Entities - registration flow

Future work:

- Wire to Admin API backend implementation
- Add fine-grained RBAC
- Support Multi-tenant contexts



OIDFed Federation Admin API - Inmor implementation

- Inmor Admin API Reference

<https://inmor.readthedocs.io/en/latest/api/admin.html>

TI Incubator Call for Ideas:

<https://wiki.geant.org/spaces/G52W5/pages/927596665/TII+call+for+Ideas>

Final Demo: in May

TIM, TIIME

Thank you

Any questions?

