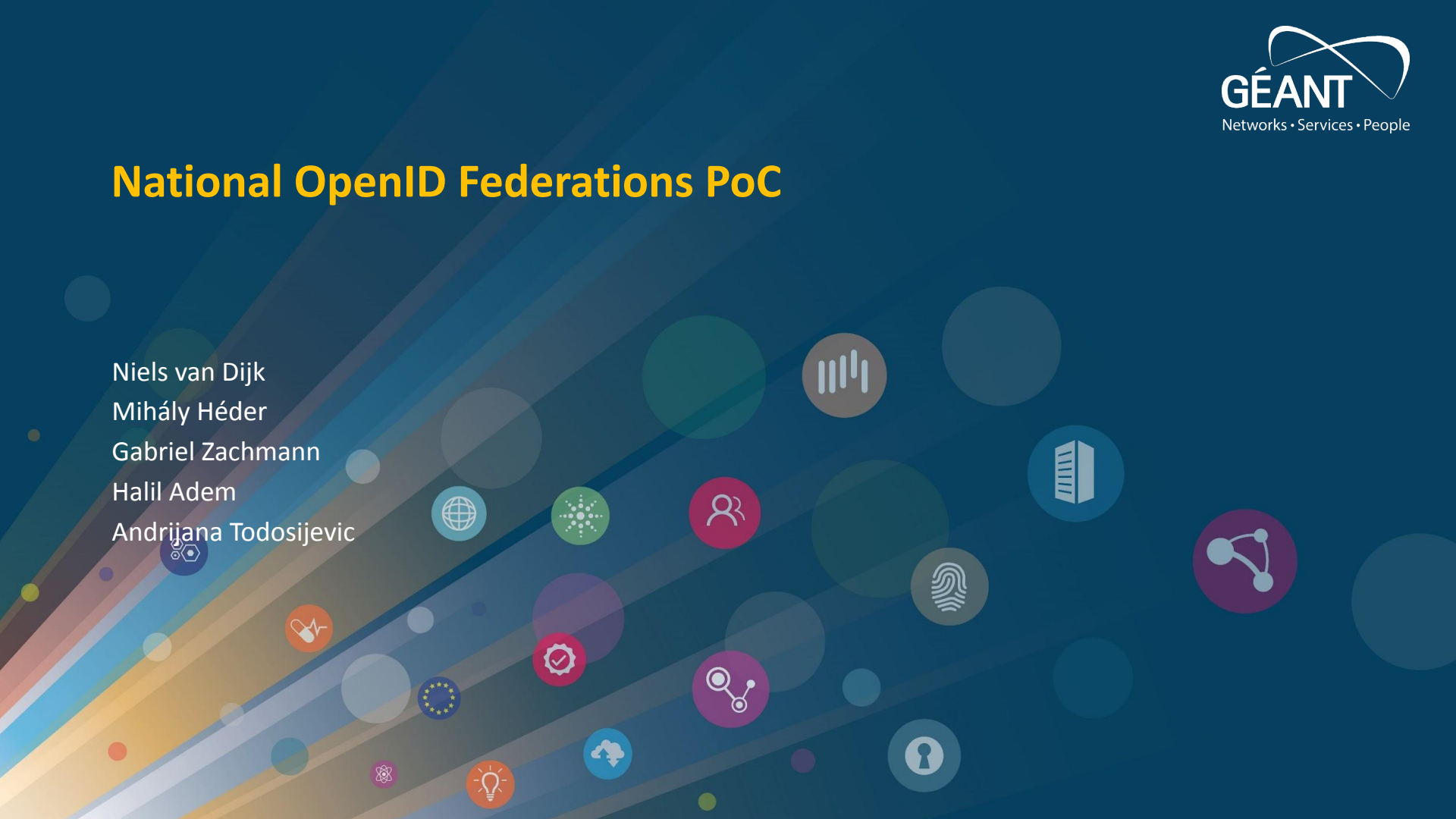
National OpenID Federations PoC

Niels van Dijk
Mihály Héder
Gabriel Zachmann
Halil Adem
Andrijana Todosijevic

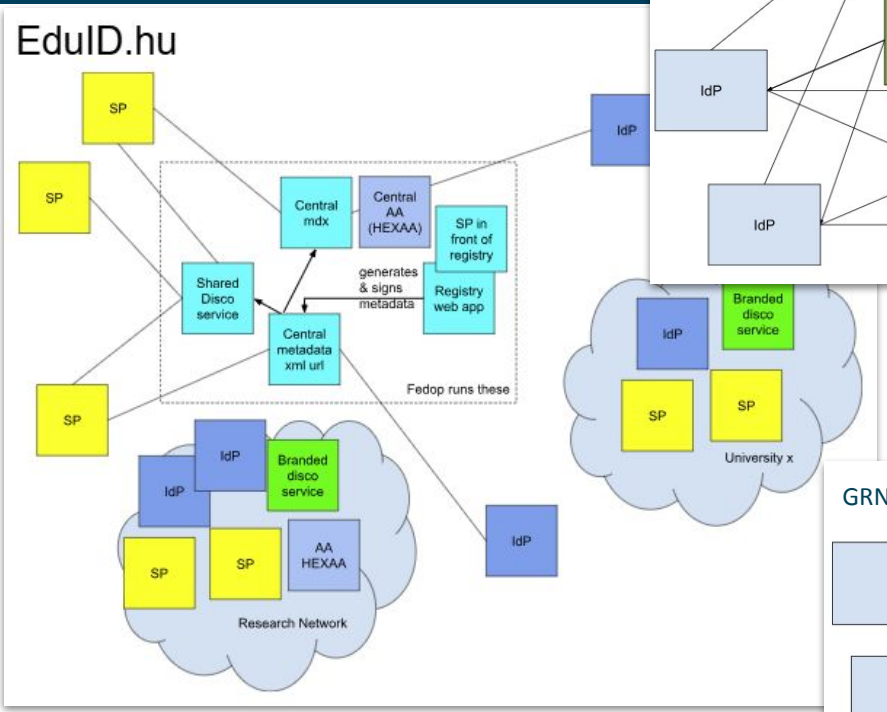


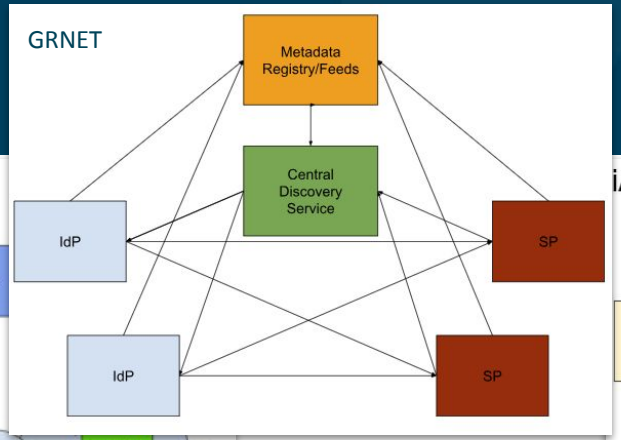
Motivation and initial idea

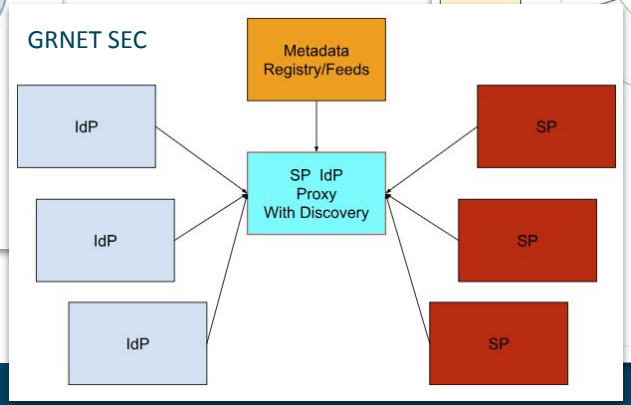
Simulate the migration from SAML Federations to OpenID Federation:

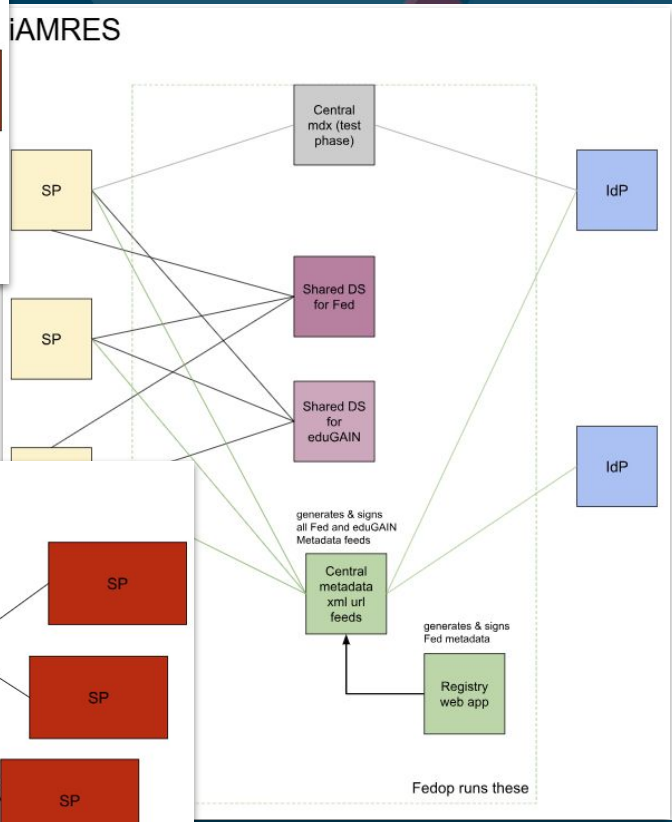
- Can FedOps replicate all functionalities that they already have in the SAML R&E Federations?
- Do the Federations have enough manpower to complete the process?
- Are all the tools ready?
- Are the necessary learning materials for FedOps available?

SAML Federation examples









Analysis and challenges

- Learning material and guides
- Parallel work within other working groups
- List of tools
  https://openid.net/developers/openid-federation-implementations/

Not enough!

Identified gaps

- DS
- RPs
- Registry

- OpenID R&E Federation Testbed
- AuthMemCookie module for Apache
- OFFA - Openid Federation Forward Auth

OpenID Federation R&E Testbed

Experimenting at scale

Ongoing work wrt OpenID Federation

- Establish OID Federation profile for eduGAIN
- REFEDs OID Federation profiling (National Federation)
- Shibboleth and SimpleSamlPHP (OP ‘done’, RP planned)
- Incubator activities on discovery

> But how do we bring all of this together?

> And how do we test and validate our assumptions, preferably at scale?

> How do we enable federation engagement
when we cannot overnight rollout OIDF?

OpenID Federation R&E Testbed

- A “copy” of eduGAIN:
  - All trust relations exist
  - But **no** authentications can flow
- Includes Trust Anchors, OPs & RPs and Trustmarks as their equivalent exist in eduGAIN today
- Entities may be added or left out to study impact

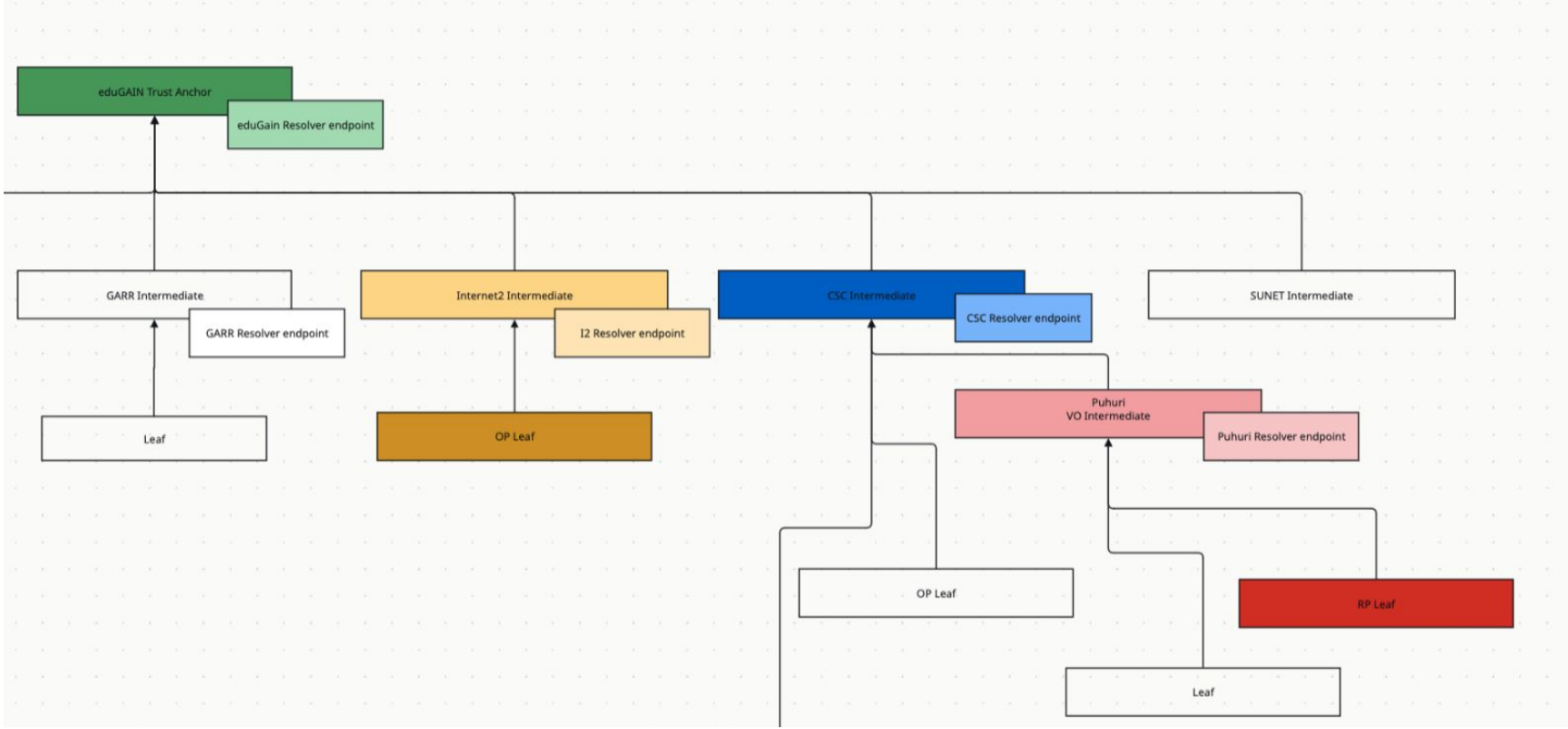
The eduGAIN OpenID Connect Profile - work in progress

TRUST is based on trust chains with eduGAIN as Trust Anchor, Federations as Intermediates and Entities as Leaves

ENTITY VALIDATION is based the eduGAIN Trust Mark. **Only validated entities can be part of trust chains with eduGAIN as Trust Anchor**

ENTITY RESOLUTION is provided by a resolver endpoint at federation and inter-federation level that provides metadata about entities

eduGAIN OpenID Federation Trust model (proposed)



Broader engagement with OpenID Federation R&E Testbed

- Test with chain resolution at scale
- OP and RP development
- Test Registry, TA, TMI and TMO tooling
- Discovery
- Interop testing
- Certification

OpenID Federation R&E Testbed - Technically

- Leverages go-oidfed (https://github.com/zachmann/go-oidfed)
- Approx 120 dockers for TAs, TMIs, TMOs
- Approx 10k Leafs
- Test OPs and RPs

OpenID Federation R&E Testbed - Demo (WIP!)

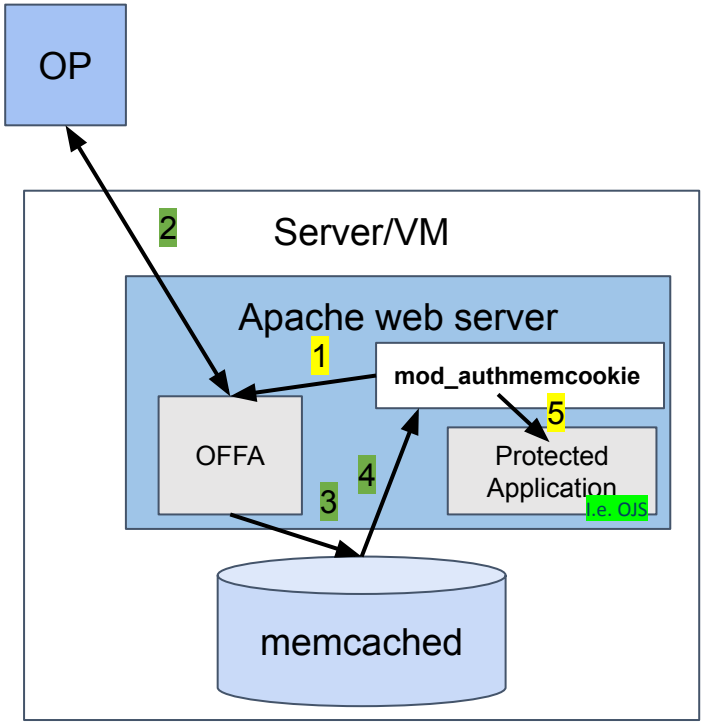
GÉANT
Networks • Services • People

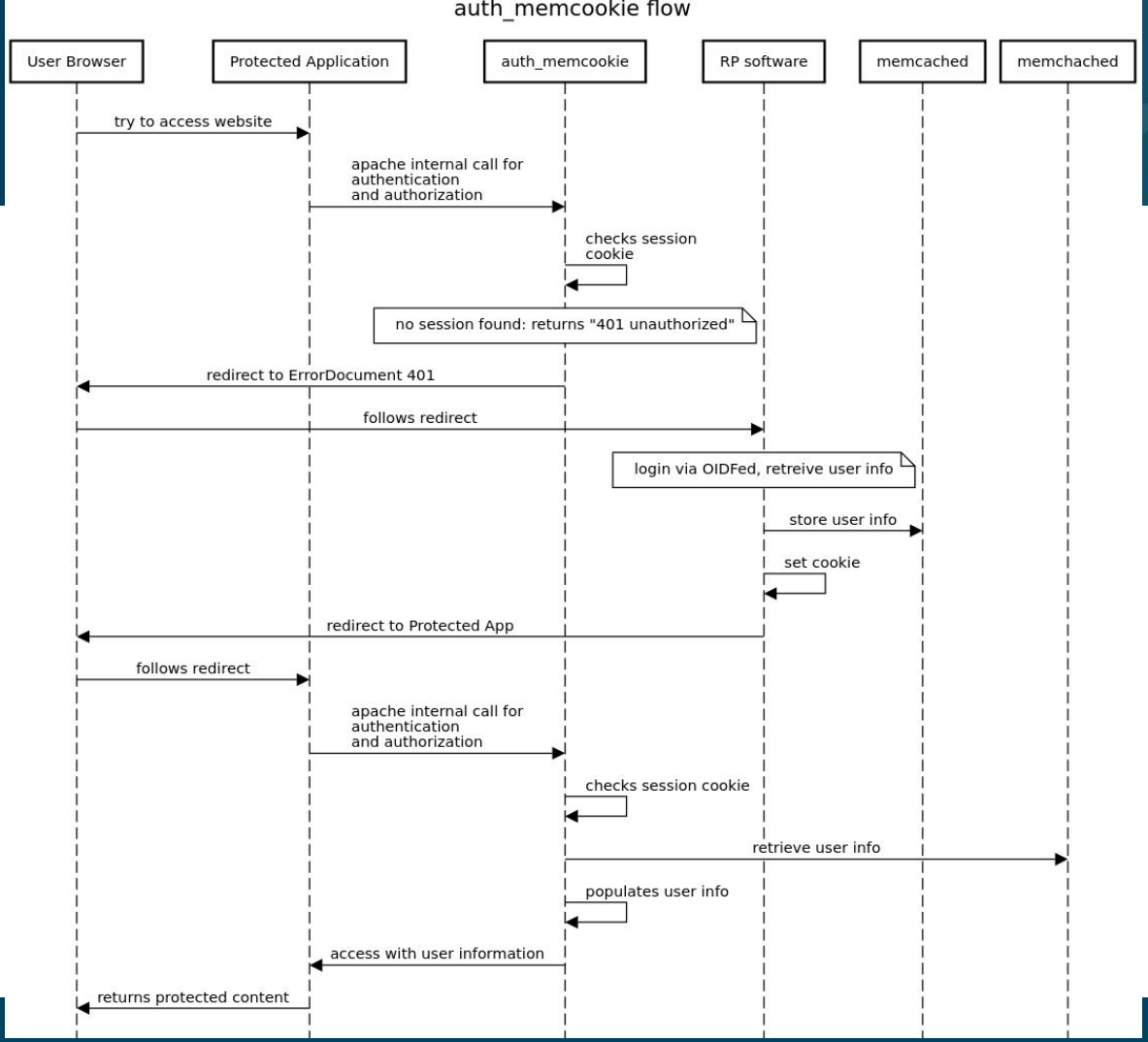
TRUST & IDENTITY
INCUBATOR

National federation engagement with OpenID Federation R&E Testbed

- Include national federation metadata into testbed
- Add test Ops and RPs
- Introduce TA Registries at national level
  - Rollout new (tested within Incubator)
  - Introduce OIDF in existing fed registry
- Start replacing dummy Entities with real ones

Instant integration via Apache module AuthMemCookie





OFFA - Openid Federation Forward Auth

- Enabling OpenID Federation SSO for “legacy” Services.
- Forward Authentication Service:
  - OFFA acts as a gatekeeper in front of services, handling authentication requests via a reverse proxy
  - Works with NGINX, Apache, Caddy
- Can also be used with the AuthMemCookie Apache Module
- Pass Userinfo to Service via HTTP Headers
- Easy to deploy with docker compose

- GitHub: https://github.com/zachmann/offa
- Documentation: https://zachmann.github.io/offa/
- Docker: https://hub.docker.com/r/oidfed/offa/tags
- Demo: https://hello.test.fedcloud.eu

OFFA - Openid Federation Forward Auth

GÉANT
Networks • Services • People

docker-compose.yaml

```yaml
services:
  caddy:
    image: caddy:latest
    restart: unless-stopped
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - ./caddy/Caddyfile:/etc/caddy/Caddyfile
      - ./caddy/data:/data
      - ./caddy/config:/config

  offa:
    image: oidfed/offa:main
    restart: unless-stopped
    volumes:
      - ./offa/config.yaml:/config.yaml:ro
      - ./offa:/data

  # This would be your service
  whoami:
    image: containous/whoami
    restart: unless-stopped
```

caddy/Caddyfile

```
offa.example.com {
  reverse_proxy offa:15661
}

whoami.example.com {
    forward_auth offa:15661 {
        uri /auth
        copy_headers X-Forwarded-User \
                     X-Forwarded-Groups \
                     X-Forwarded-Name \
                     X-Forwarded-Email \
                     X-Forwarded-Provider \
                     X-Forwarded-Subject
    }

    reverse_proxy whoami:80
}
```

offa/config.yaml

```yaml
server:

logging:
  access:
    stderr: true
  internal:
    level: info
    stderr: true

sessions:
  ttl: 3600
  cookie_domain: example.com

auth:
  - domain: whoami.example.com
    require:
      groups: users

federation:
  entity_id: https://offa.example.com
  trust_anchors:
    - entity_id: https://ta.example.com
  authority_hints:
    - https://ta.example.com
  logo_uri: https://offa.example.com/static/img/offa-
  key_storage: /data
  use_resolve_endpoint: true
  use_entity_collection_endpoint: true
```

OFFA - Openid Federation Forward Auth

Demo: https://hello.test.fedcloud.eu

```
Hostname: 4aa7f0bc4746
IP: 127.0.0.1
IP: ::1
IP: 172.18.0.14
RemoteAddr: 172.18.0.9:60850
GET / HTTP/1.1
Host: hello.test.fedcloud.eu
User-Agent: Mozilla/5.0 ...
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.5
Cookie: offa-session=SLn7J9WOST...
...
Via: 2.0 Caddy
X-Forwarded-For: XX.XX.XXX.XXX
X-Forwarded-Host: hello.test.fedcloud.eu
X-Forwarded-Proto: https
X-Forwarded-Provider: https://idp.mivanci.incubator.hexaa.eu
X-Forwarded-Subject: testuseruid
X-Forwarded-User: testuseruid
```

Openid Federation Forward Auth

Future plans

- Continue our current work, finish the Testbed
- SimpleSamlPHP RP
- apache_mod_oidfed

Reach out to us!