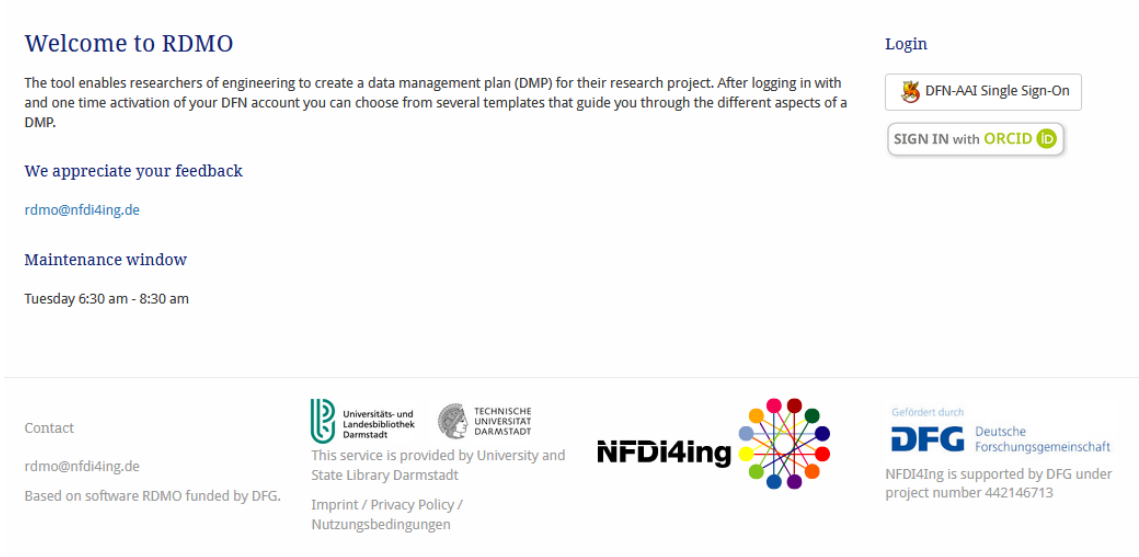
Local Authentication Infrastructure with RDMO as Use Case at ULB Darmstadt

RDMO-Community-Treffen March 29th, 2023

David Wallace
Team Forschungsdaten-Services
david.wallace@tu-darmstadt.de

Our NFDI4Ing Instance

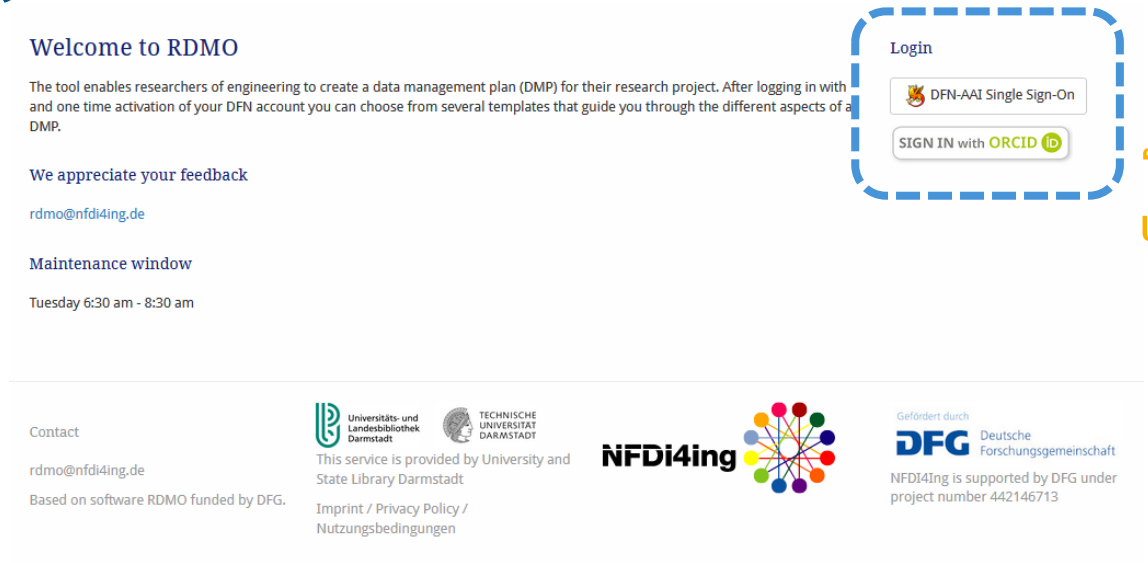
rdmo.nfdi4ing.de



Our NFDI4Ing Instance: feature request

rdmo.nfdi4ing.de

Authentication



DFN-AAI
&
“External users”

Requirements for the service

Challenge: to implement both SAML and OIDC based Authorization in parallel.

Authentication:

simple login for all potential users (Single Sign-On, no registration etc.)

- German Institutional login, DFN-AAI* Identity Providers **(SAML)**
- “external users” via ORCID or other social account providers **(OIDC)**

Authorization:

user permissions/roles are managed internally of the RDMO Instance

- Authenticated Users create their own Projects, can invite other Users to join those Projects.
- Certain authenticated users are given elevated permissions by RDMO admins.
  - Alternatively, manageable via a VO, shared with other services.

*AAI = Authentication and Authorization Infrastructure

Implementations of Authentication

Three main implementations for RDMO which are *mutually exclusive**

- Installing a Shibboleth service provider (SP) next to RDMO and connect to an identity provider or even a whole Shibboleth federation. (**SAML**)
- Regular user accounts with registration using the django-allauth library, can be extended with OAuth2 and OpenID Connect Providers. (**OIDC**)
- Using a (read-only) connection to a LDAP server
  - currently, not in our scope of interest

Considered as *mutually exclusive* for the moment. [1]

*however, a new Shibboleth Setup might allow for parallel implementations.

[1] rdmo.readthedocs.io/en/latest/configuration/authentication/index.html

Implementations of Authentication

Shibboleth Service Provider, institutional login

Technical stack for RDMO [2]:
shibboleth SP / apache2 vhost / django-shibboleth-remoteuser

Requirements:
- ☑ SAML: DFN-AAI Discovery Service
- ☒ OIDC: *currently* not possible because of vhost configuration*

Possible solutions:
- ☒ One Identify Provider that supports both protocols?
  - There are currently no IdPs in the DFN-AAI that support OIDC, because OIDC does not yet support federation
- ? Change the vhost config so that it allows for protocols in parallel
  - *a new Shibboleth Setup might enable this

[2] rdmo.readthedocs.io/en/latest/configuration/authentication/shibboleth.html#shibboleth

Implementations of Authentication

django-allauth, social login providers

Technical stack for RDMO [3]:

django-allauth / add supported providers in settings

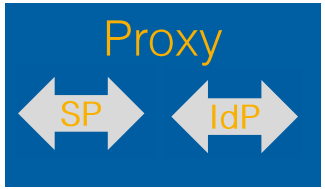
Requirements:

- ☒ SAML: protocol not supported
- ☑ OIDC: social IdPs can be added

Possible solutions:

- ☑ Add an OIDC Provider (proxy) that supports translation of protocols and mediates between the DFN-AAI Discovery Service and RDMO
  - This solution was implemented in the ULB

[3] https://rdmo.readthedocs.io/en/latest/configuration/authentication/allauth.html



Current solution: Keycloak and SaToSa

Provides IAM for different services of the ULB, supports both SAML & OIDC

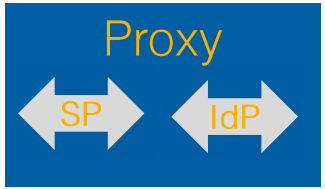
KEYCLOAK

Keycloak

- Open Source Identity and Access Management Software
- Lots of features and configuration:
  - federation, user management, fine-grained authorization
  - ☑ Supports: SAML & OIDC IdPs *
- Deployed with Docker, serves a web interface to URL
- Each service has its own „realm“, IdPs can be shared between realms.

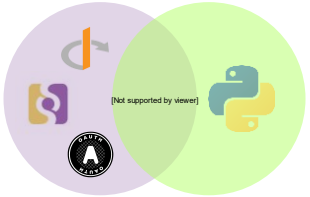
- ☒ However, the Discovery Service (wayf) could not be implemented

*alternative software with support for LDAP could be authentik



Current solution: Keycloak and SaToSa

Provides an IAM for different services of the ULB, supports both SAML & OIDC



[4] The Identity Python

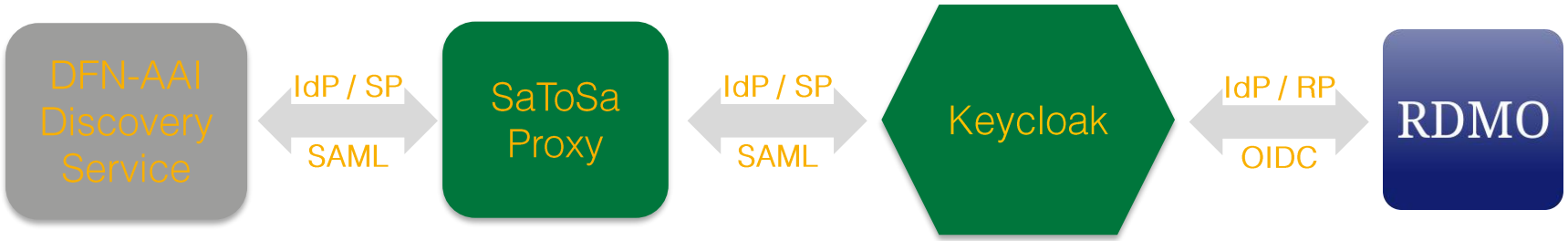
SaToSa[4]
- Proxy translating between different authentication protocols
- SAML < - > SAML
  - Single Service Provider < - > Multiple Identity providers
  - ☑ Supports the Discovery Service
- Attribute mapping in configuration
  - saml (pairwise-id or EduPPN), oidc (sub), orcid

- ☑ Setup and registered as a DFN-AAI shibboleth SP **(SAML)**
- ☑ In Keycloak installed as a “IdP” **(OIDC)**

- Embedded Discovery Service
  - a costum landing page for the Discovery Service was developed in go

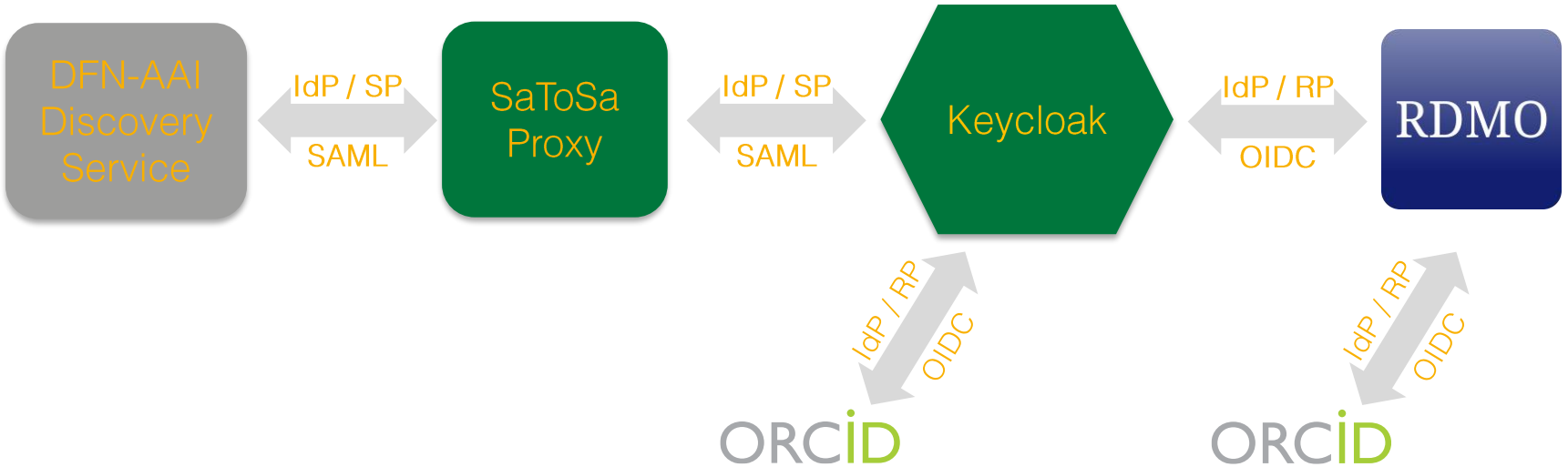
Current solution: Keycloak and SaToSa

Overview of local authentication infrastructure



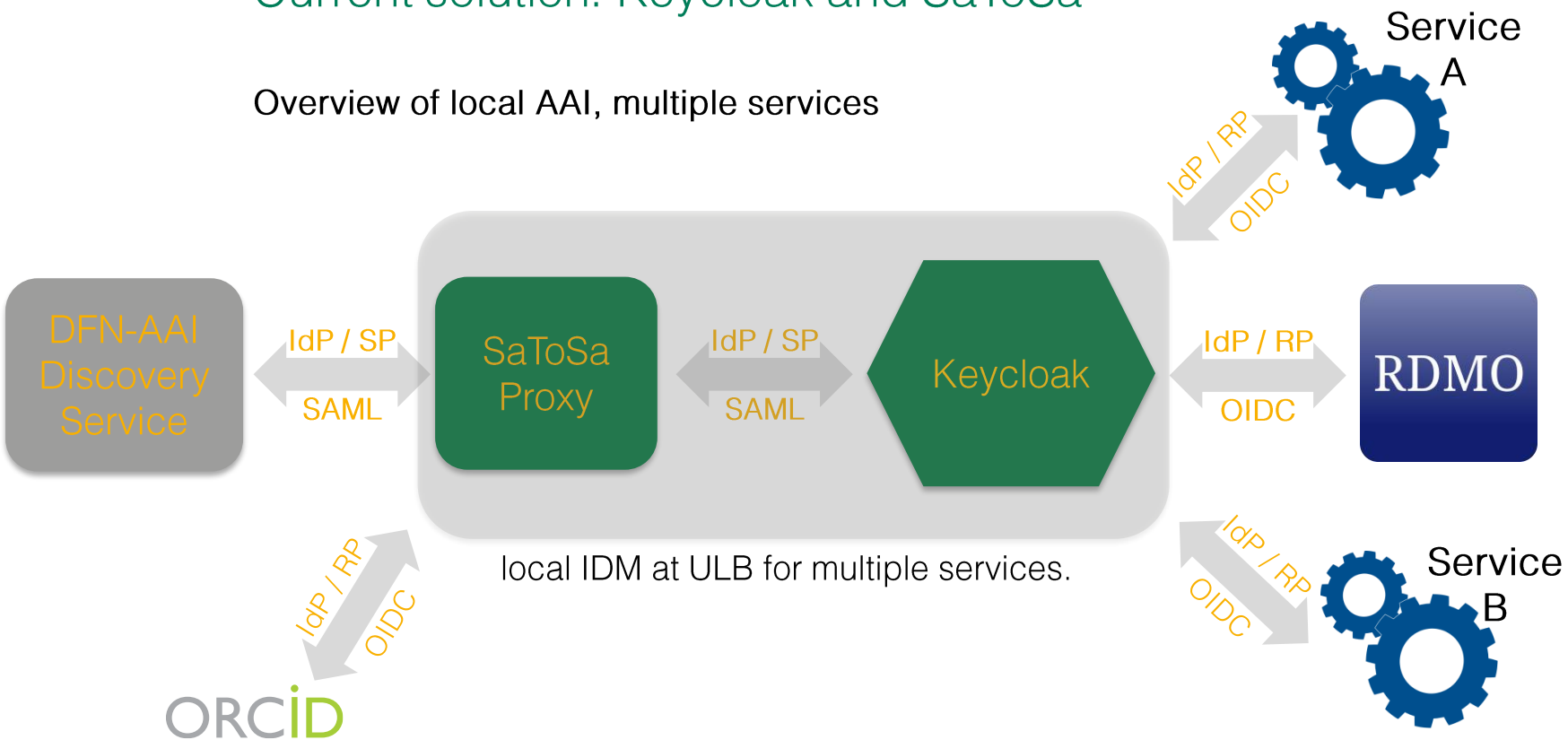
Current solution: Keycloak and SaToSa

Overview of local authentication infrastructure, ORCID addition



Current solution: Keycloak and SaToSa

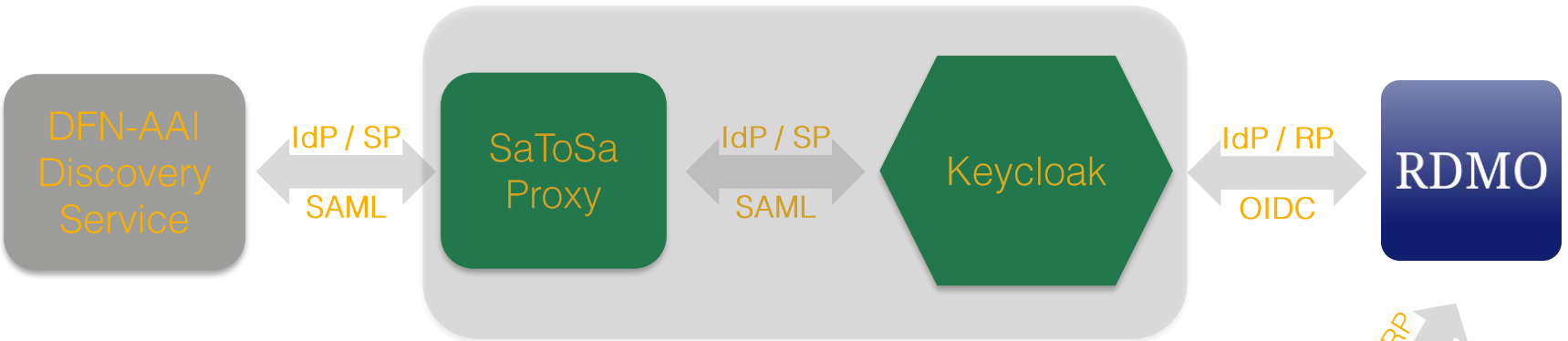
Overview of local AAI, multiple services



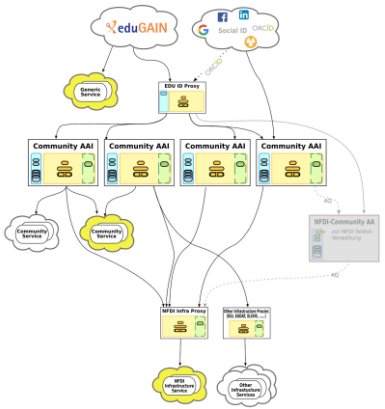
local IDM at ULB for multiple services.

Current solution: Keycloak and SaToSa

Overview of local authentication infrastructure, other AAI proxies.



For RDMO, this local solution could be replaced with other AAI proxies that connect to the DFN-AAI such as the NFDI AAI Community AAIs or Infrastructure Proxies.



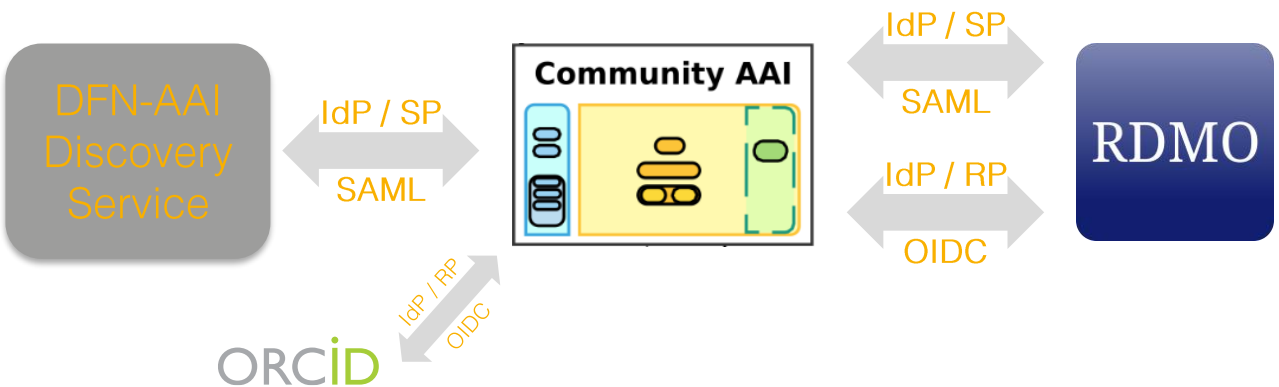
https://doc.nfdi-aai.de

Outlook NFDI AAI: „Community AAI“ or „Infrastructure Proxy“

The NFDI AAI implementation will support both SAML or OIDC

A NFDI Community AAI (for a NFDI Consortium) or Infrastructure Proxy would support both protocols. This would make implementation in RDMO straightforward.

- The authentication protocol for each service can be chosen based on certain requirements or technical considerations.



Thank you! Questions? Comments?

Special gratitude to:

Bund, Ländern und die Gemeinsame Wissenschaftskonferenz (GWK) für die Förderung und Unterstützung im Rahmen des Konsortiums NFDI4Ing. Gefördert durch die Deutsche Forschungsgemeinschaft (DFG) - Projektnummer 442146713.

David Wallace
Forschungsdaten-Services
ULB Darmstadt
david.wallace@tu-darmstadt.de