RIA-653549

INDIGO – DataCloud

How INDIGO brokers identities and does Authentication and Authorization

Andrea Ceccanti (INFN)
on behalf of the INDIGO AAI TF

indigo-aai-tf@lists.indigo-datacloud.org

1st ASTERICS-OBELICS Workshop
12/12/2016, Rome, Italy

H2020-Astronomy ESFRI and Research Infrastructure Cluster
(Grant Agreement number: 653477).

The INDIGO-DataCloud project

INDIGO-DataCloud

- An H2020 project approved in January 2015 in the EINFRA-1-2014 call
  - 11 M€
  - 30 Months (Apr. 2015 -> Sept. 2017)
- **Who:** 26 partners from 11 European countries
- **What:** develop an **open source** platform for computing and data targeted at **multi-disciplinary scientific communities**
- **Where:** provisioned over hybrid (public and private) e-infrastructures

INDIGO objectives

- Provide **seamless access** to data and computing provisioned over private, public or hybrid e-infrastuctures
- Leverage and extend current Cloud technologies, **fill the gaps,** provide tools and services to support scientists, software developers, resource providers for the **efficient exploitation of computing, data and network technologies**:

Better software for better science

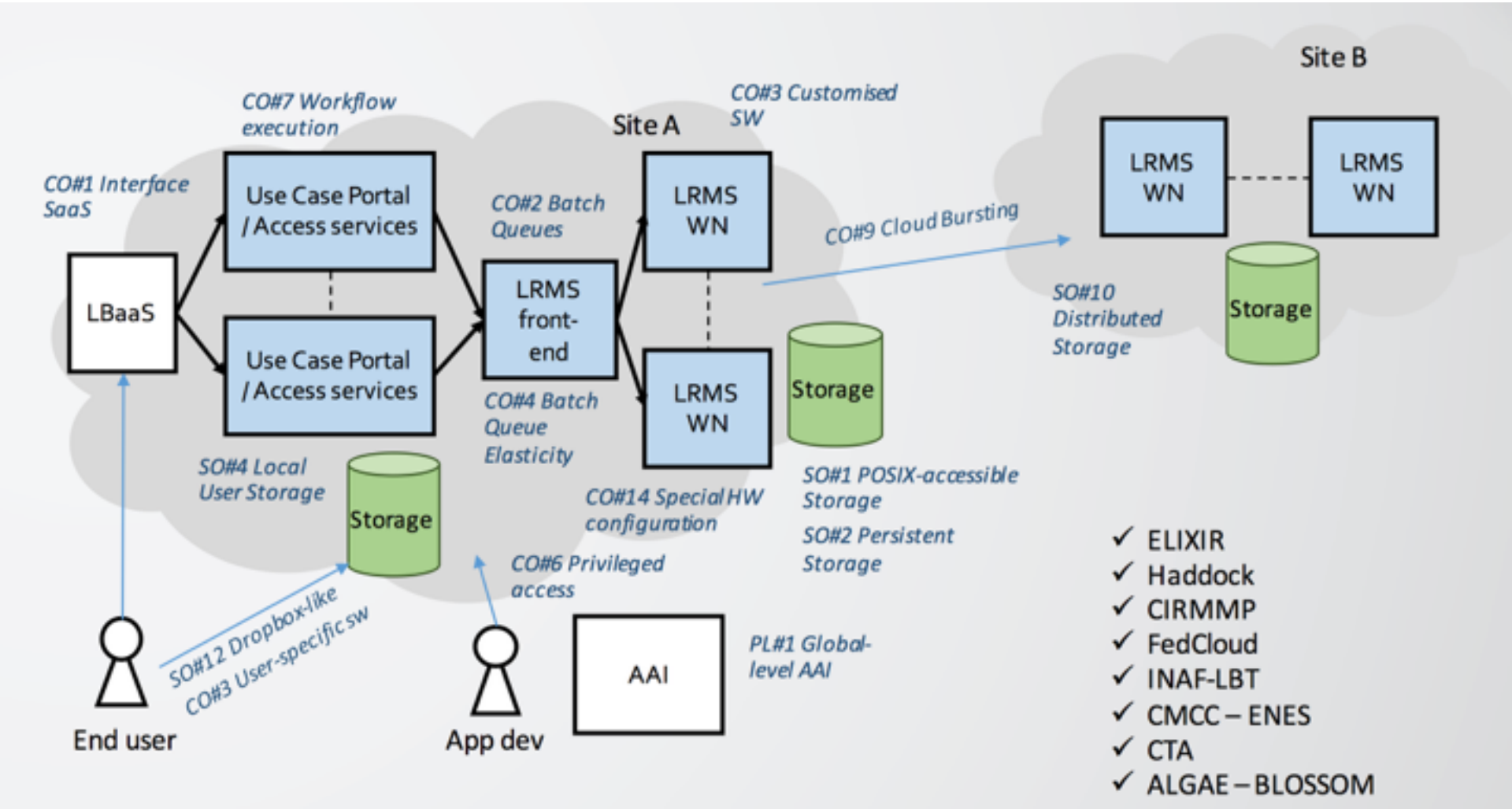
INDIGO - DataCloud

The INDIGO approach

- Based on Open Source solutions
  - widely supported by big communities
- whenever possible exploit general solutions instead of specific tools/services
  - increased sustainability
- ensure that the framework offered to final users, as well as to developers, will have a **low learning curve**
  - ease of adoption and integration

Example use case scenario

Computational Portal "as a service"



The INDIGO-Datacloud AAI

INDIGO AAI: main challenges

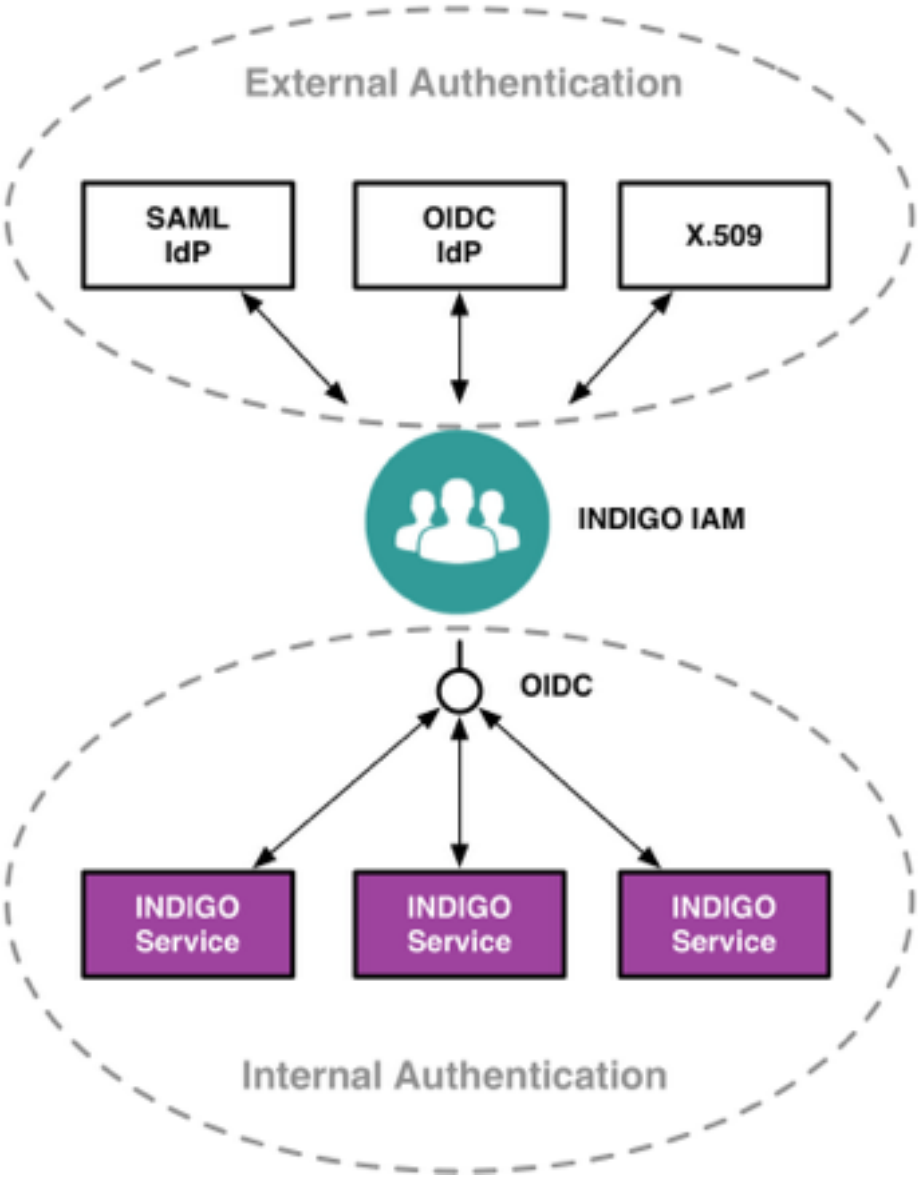
- Authentication
  - Support for **federated AuthN & social logins**
- Identity Harmonisation
  - Link multiple accounts to a single INDIGO identity, providing a **persistent identifier** orthogonal to AuthN mechanism
- Authorization
  - **Orthogonal to AuthN**, attribute-based, dynamic
  - Consistent across heterogeneous infrastructures
- Delegation
  - Provide the **ability for services to act on behalf of a user**
  - Support **offline access for long-running applications**
- Provisioning
  - provision/de-provision identities to services/relying resources
- Token translation
  - **enable integration with services relying on heterogeneous AuthN mechanisms**

Authentication/Identity

Slide courtesy of Paul Millar

Identity in INDIGO

- The INDIGO identity layer speaks **OpenID-connect**
- The INDIGO **Identity and Access Management Service** is an OIDC provider
  - Authenticates users with supported AuthN mechanism
    - SAML, X.509, OIDC
  - Provides persistent identifier and links other attributes (e.g., group membership) to the INDIGO identity
- Provides to RP access to identity information through standard OIDC interfaces



Why OpenID connect

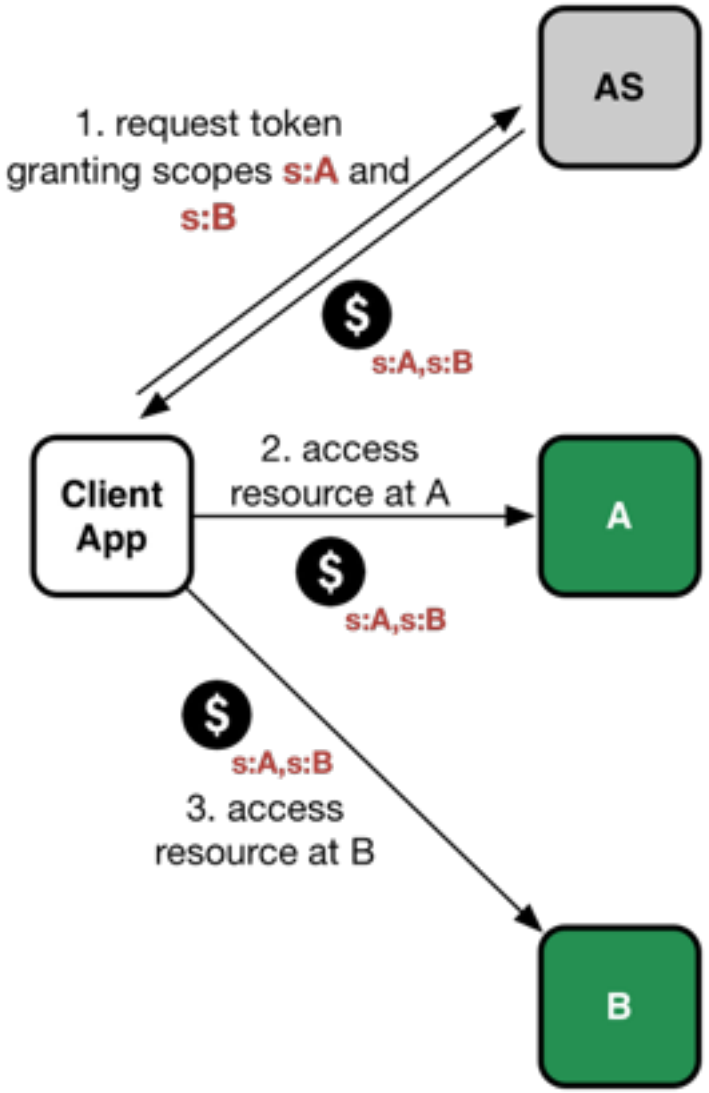
- Standard and widely adopted in industry
  - Don't reinvent the wheel
- Reduced integration complexity in relying services
  - A lot easier than SAML
- Lots of things we need are covered and standardized
  - Dynamic Registration of clients/relying parties
  - Token revocation
  - Discovery
  - Session management
  - Distributed/Aggregated claims
- Mobile-friendly

Authorization

Slide courtesy of Paul Millar

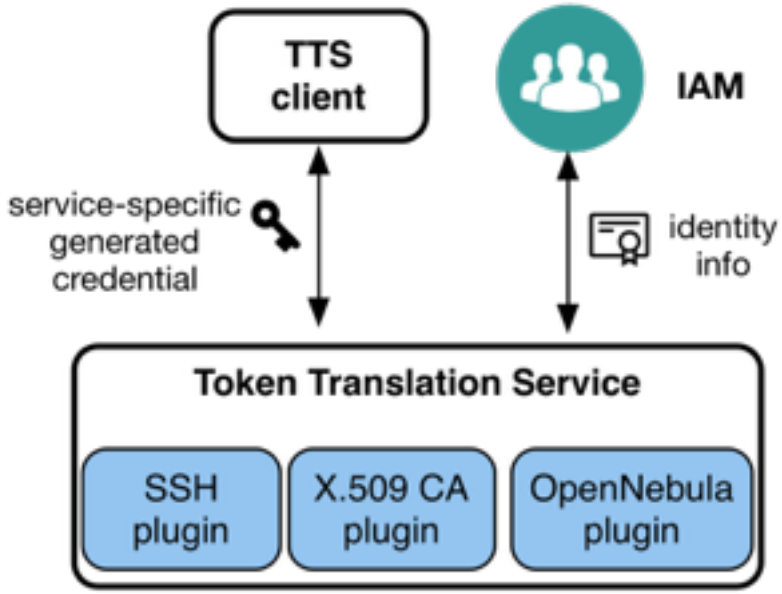
OAuth2 AuthZ in INDIGO

- INDIGO services are HTTP APIs protected by an **OAuth** Authorization Service (AS)
- In order to access resources, a client needs an **access token**
- **OAuth scopes** used to
  - target the token to specific APIs/services
  - provide hints for finer grained authZ
- **Identity layer provides other attributes** as base for AuthZ decisions
  - e.g., group membership attributes

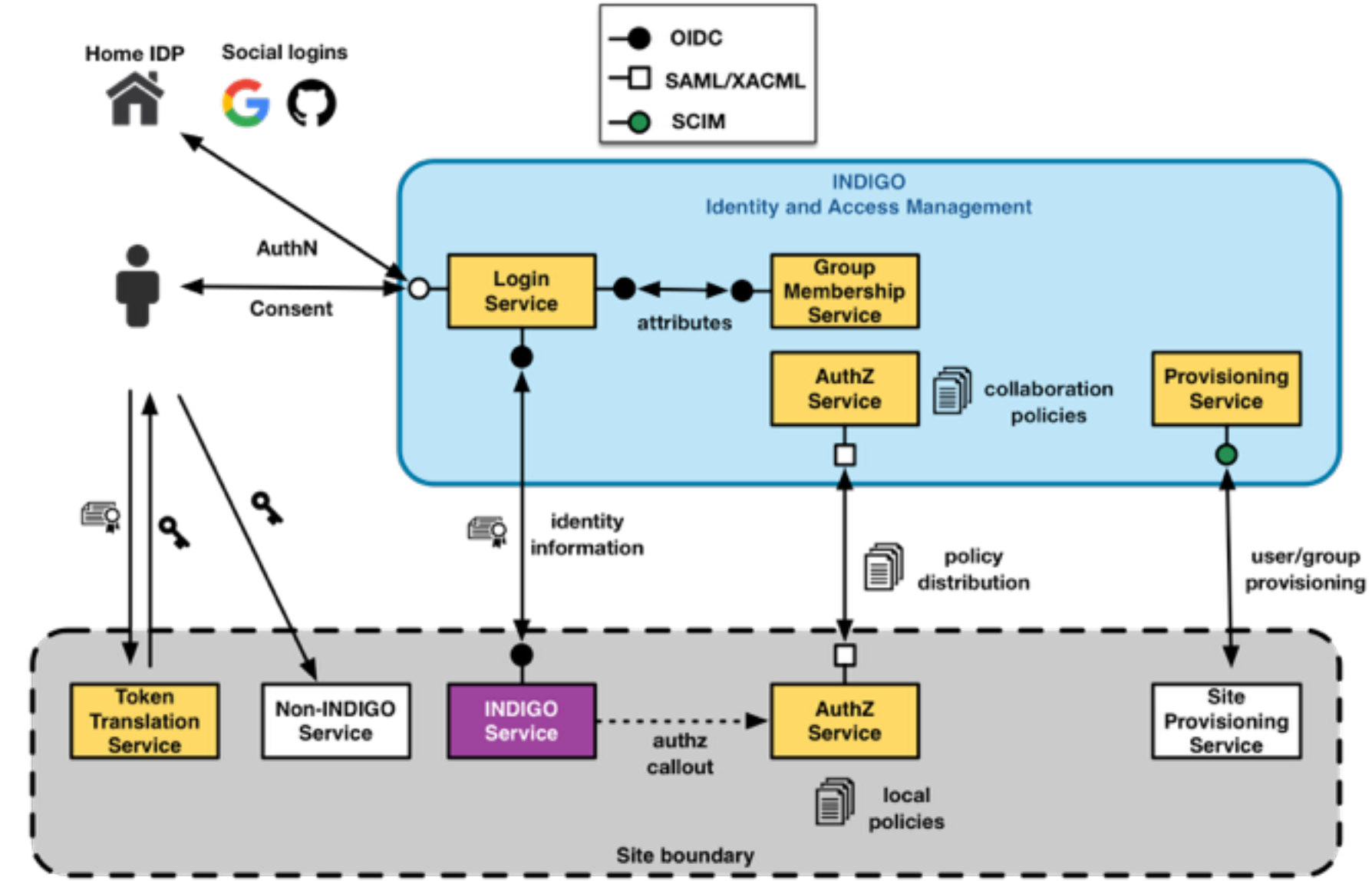


Token translation

- What about integration with services that do not speak OpenID-connect?
- The INDIGO **Token Translation Service** (TTS)
  - maps INDIGO identity & attributes to external service credentials
  - provides an extensible plugin-based architecture, and currently support the generation of
    - ssh keypairs
    - X.509 certificates
    - Opennebula username/password credentials



INDIGO AAI architecture



INDIGO - DataCloud

Which in practice looks like this…

(DEMOS)

INDIGO - DataCloud

Demo #1

Users can manage IAAS resources authenticating with their home IdP

Authorization takes into account groups linked to their INDIGO identity

Demo #1 setup

- **IFCA Openstack** has two configured projects available to users registered in the **INDIGO IAM** that belongs to specific groups:
- **vo:indigo**: access to this project (and its subprojects) is granted to any member of the IAM **Developers** group
- **vo:indigo:users**: access to this project is granted to any member of the IAM **ifca-users** group
- Authorization is **enforced consistently** for web and command-line clients

Demo #1 setup: users

- andrea is the **privileged** user
  - member of the **Developers** group
- silvio is the common user
  - member of the **ifca-users** group

Demo # 1

INDIGO - DataCloud

Demo #1: recap

- IFCA Openstack delegates user authentication to INDIGO IAM via **OpenID-connect**
- INDIGO IAM authenticates user via **SAML** and provides identity information (including **a persistent user identifier and group membership attributes**) to IFCA Openstack
- Based on the group information returned by the INDIGO IAM, IFCA Openstack **decides which projects** a user can access
- Authorization is **enforced consistently** for web and command-line clients

INDIGO - DataCloud

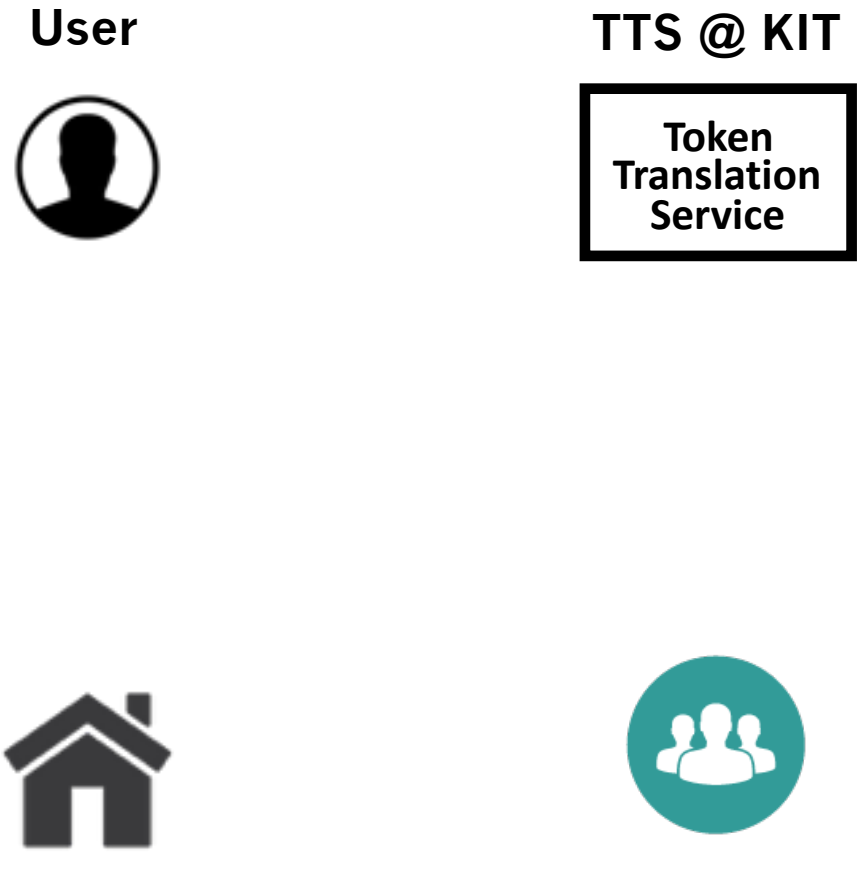
Demo #2

Users can obtain ssh credentials on demand based on their INDIGO account attributes

Authorization takes into account groups linked to their INDIGO identity

Demo #2 setup

- The **INDIGO Token Translation** Service (TTS) @ KIT is linked to the INDIGO IAM for user authentication/authorization
- Users in the **kit-ssh** IAM group can request ssh keypair generation based on their IAM credentials. The generated credentials can be used to access a VM running @ KIT
- Users in the **kit-x509** IAM group can request the generation of an X.509 certificate based on their IAM credentials.



INDIGO - DataCloud

Demo #2 setup: users

- **andrea** is the **privileged** user
  - member of the **kit-ssh** and **kit-x509** groups
- **silvio** is the unprivileged user
  - he's not a member of any group authorized by the TTS

Demo # 2

Demo #2: recap

- KIT Token Translation Service (**TTS**) delegates user authentication to INDIGO IAM via **OpenID-connect**

- INDIGO IAM authenticates user via **SAML** or **username/password** and provides identity information (including a persistent user identifier and group membership attributes) to the TTS

- Based on the group membership information returned by the INDIGO IAM, the **TTS decides whether credentials can be generated for a user**

INDIGO - DataCloud

INDIGO AAI added value

- **OpenID-connect** as the identity layer simplifies integration in relying services and works well with dynamic infrastructures
- **OAuth** as the authorization layer provides native support for delegation & offline access and is the standard for authz on HTTP APIs
- The **Identity and Access Management** (IAM) service
  - supports several authn mechanism (SAML, X.509, OpenID-Connect)
  - provides **persistent identifier** and **group membership attributes** to relying services via standard OpenID-connect interfaces
- The **Token Translation Service** (TTS) enables integration and Authn/AuthZ enforcement in services that do not speak OpenID-connect

INDIGO - DataCloud

Thanks!

indigo-aai-tf@lists.indigo-datacloud.eu

INDIGO - DataCloud

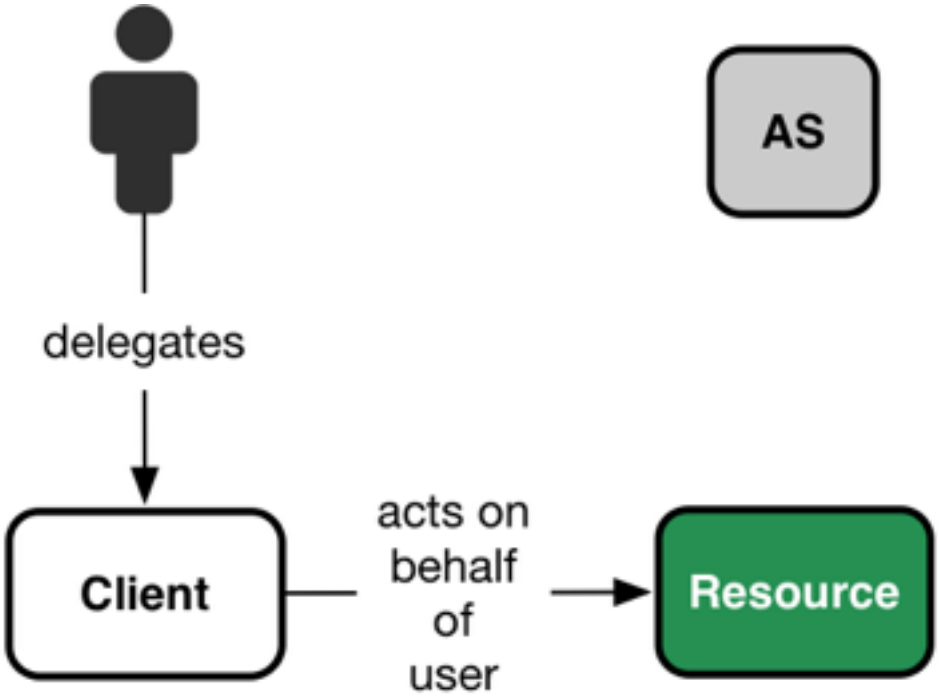
Backup slides

INDIGO - DataCloud

Delegation & offline access

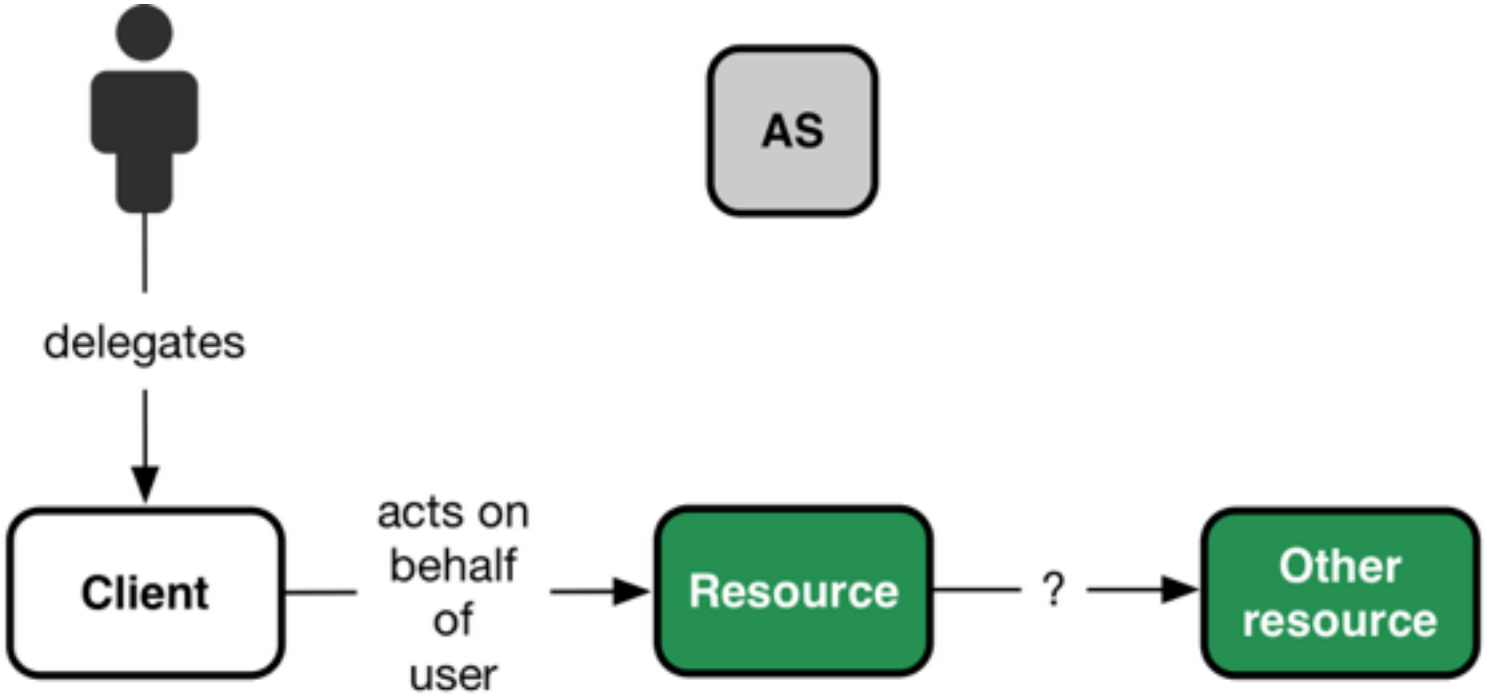
INDIGO - DataCloud

OAuth delegation



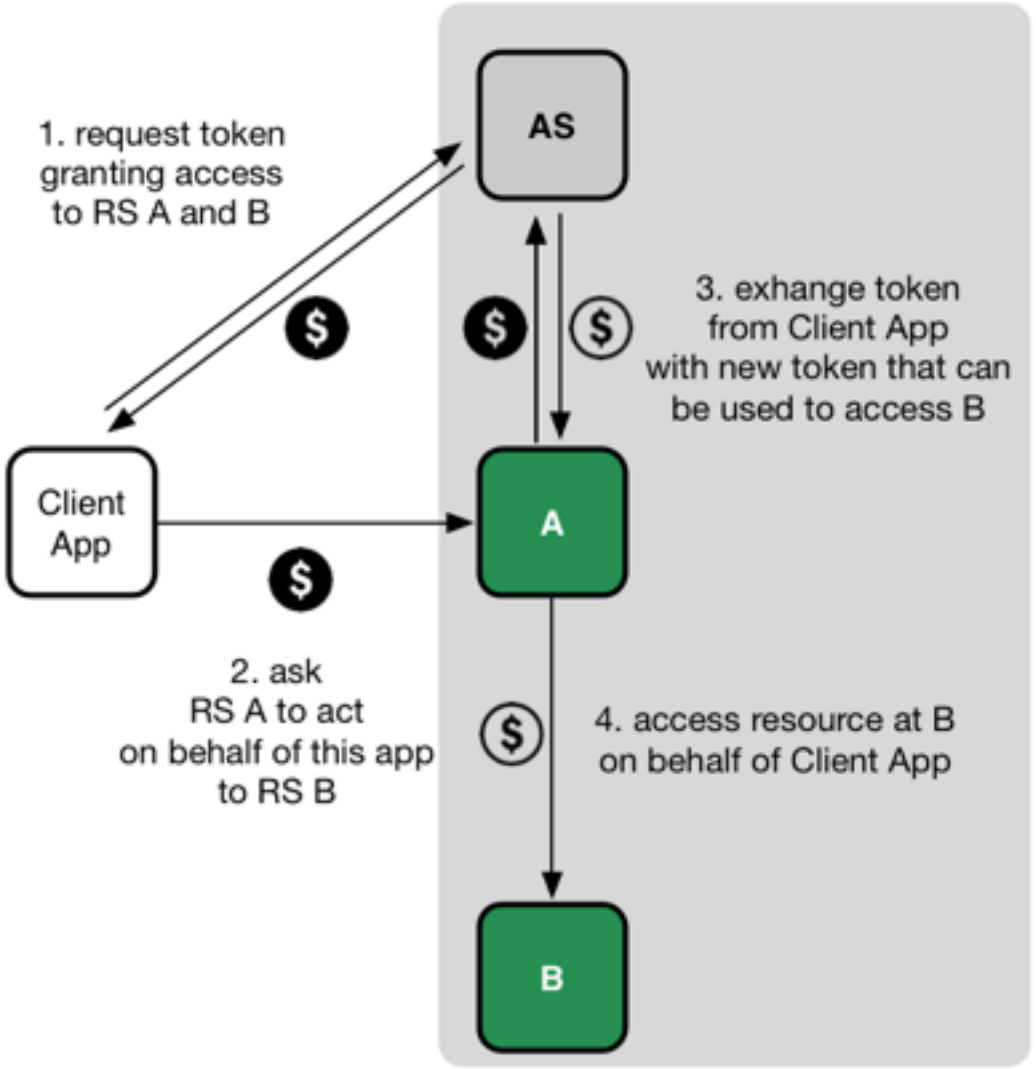
INDIGO - DataCloud

OAuth chained delegation?



OAuth token exchange

- OAuth flow to implement chained delegation among services
- Under standardization
- Supports impersonation and delegation



INDIGO - DataCloud

Provisioning

SCIM

- A distributed infrastructure demands and interoperable way of propagating identity and group information to all involved resources
- INDIGO AAI relies on the standard **S**ystem for **C**ross-domain **I**dentity **M**anagement (SCIM) v. 2.0
- Indigo IAM SCIM APIs provide means to propagate identity and group information to relying services, to implement, for instance, dynamic account creation and other resource lifecycle management at various levels of the INDIGO infrastructure depending on events related to user identity status.

Example Authentication Flow

INDIGO AuthN flow

INDIGO Service

Access service

Marcus wants to access some service at INDIGO service

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

Access service

The INDIGO Service (IS) sees that Marcus is not authenticated, and redirects him to INDIGO IAM for authentication

Home IdP

Indigo IAM

INDIGO AuthN flow

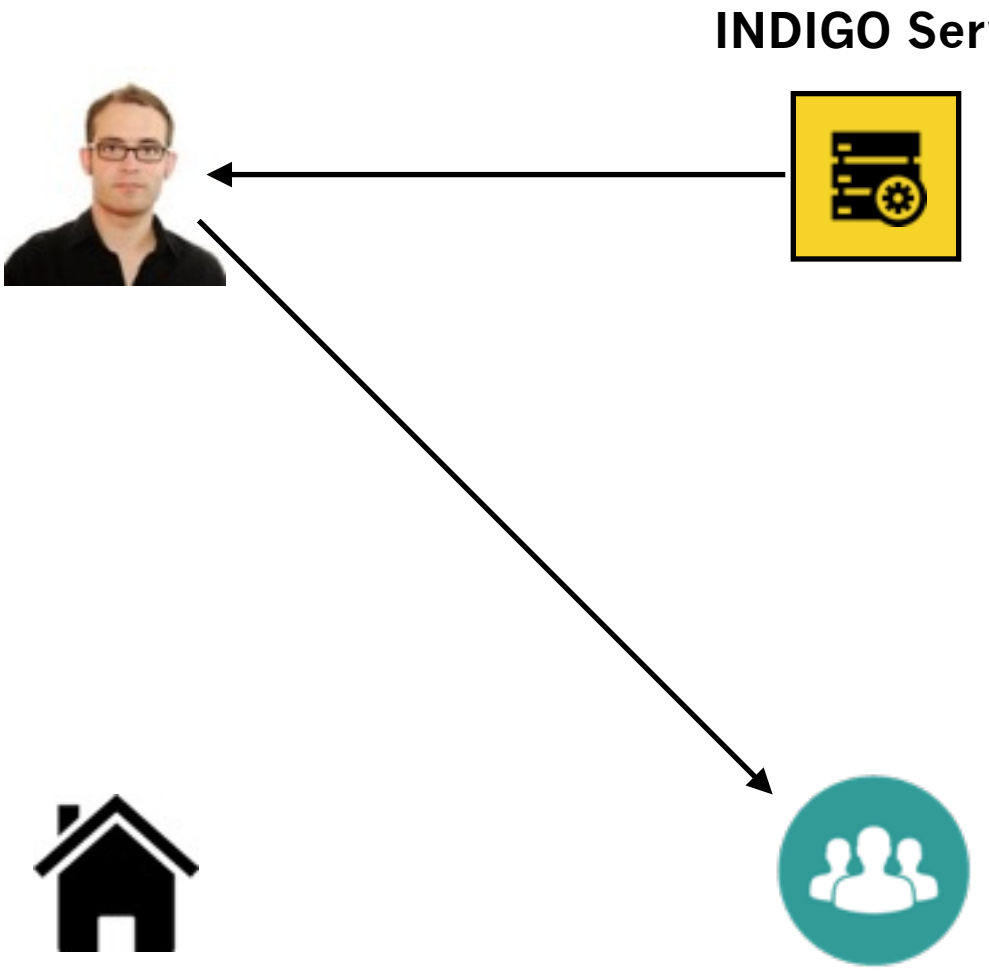
INDIGO Service

redirect Marcus to IAM
for AuthN

Home IdP

Indigo IAM

INDIGO AuthN flow

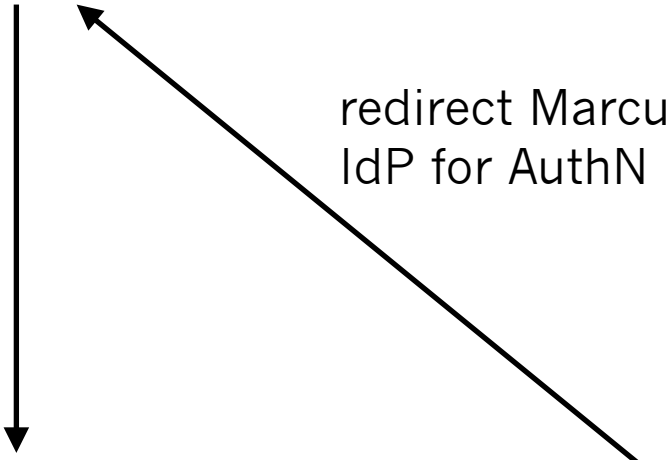


IAM lets Marcus choose how he wants to authenticate

Marcus chooses his Home IdP

INDIGO AuthN flow

INDIGO Service



Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

Home IdP authenticates Marcus and sends back a signed Authentication assertion.

This can be a SAML assertion or an OpenID connect JSON Web Token, depending on the type of the home IdP

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

IAM validates assertion.
Marcus is now
authenticated at IAM.

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

IAM sends and OIDC authorization code back to the IS via an http redirect. This is a standard OIDC authorization code flow

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

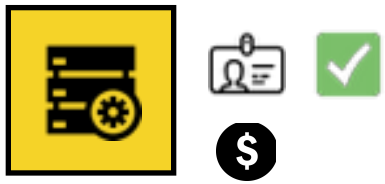
The IS exchanges the received authZ code for and OIDC ID-token and access token

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service



IS validates ID-Token.
Marcus is now authenticated at IS

Home IdP

Indigo IAM

INDIGO AuthN flow

INDIGO Service

IS requests additional profile information about Marcus from IAM user info endpoint.

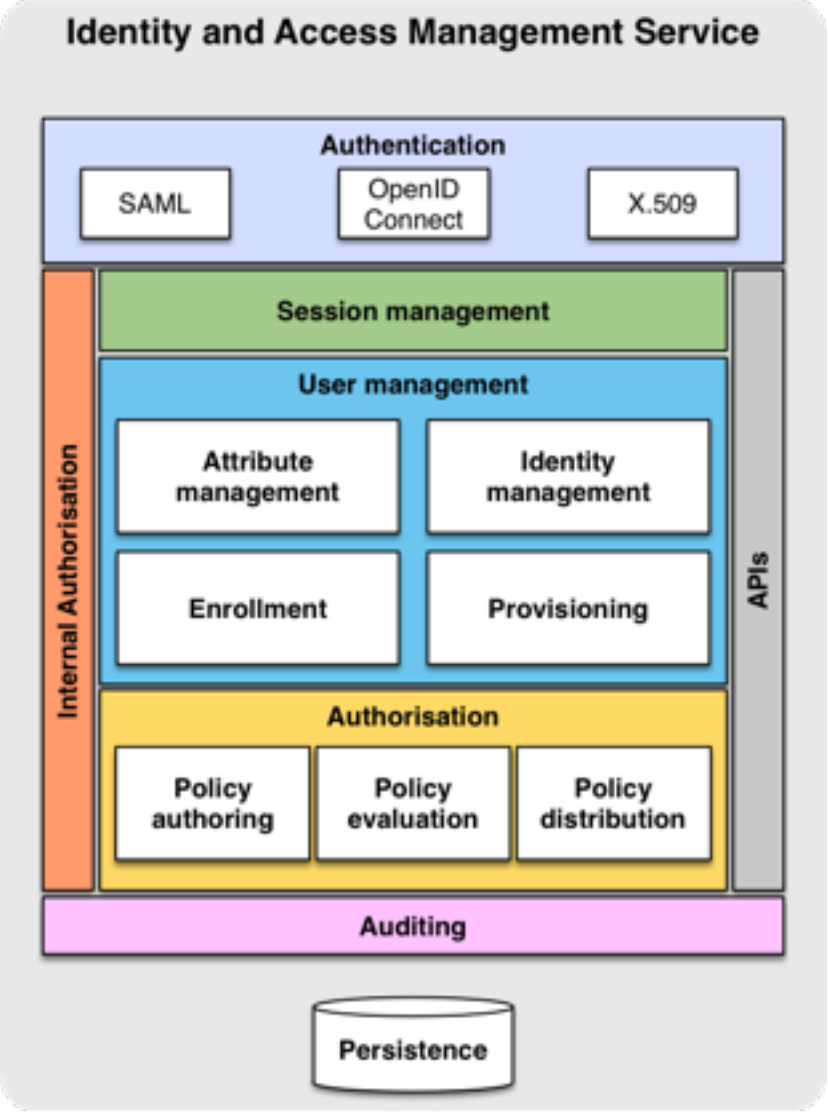
Home IdP

Indigo IAM

The INDIGO IAM service

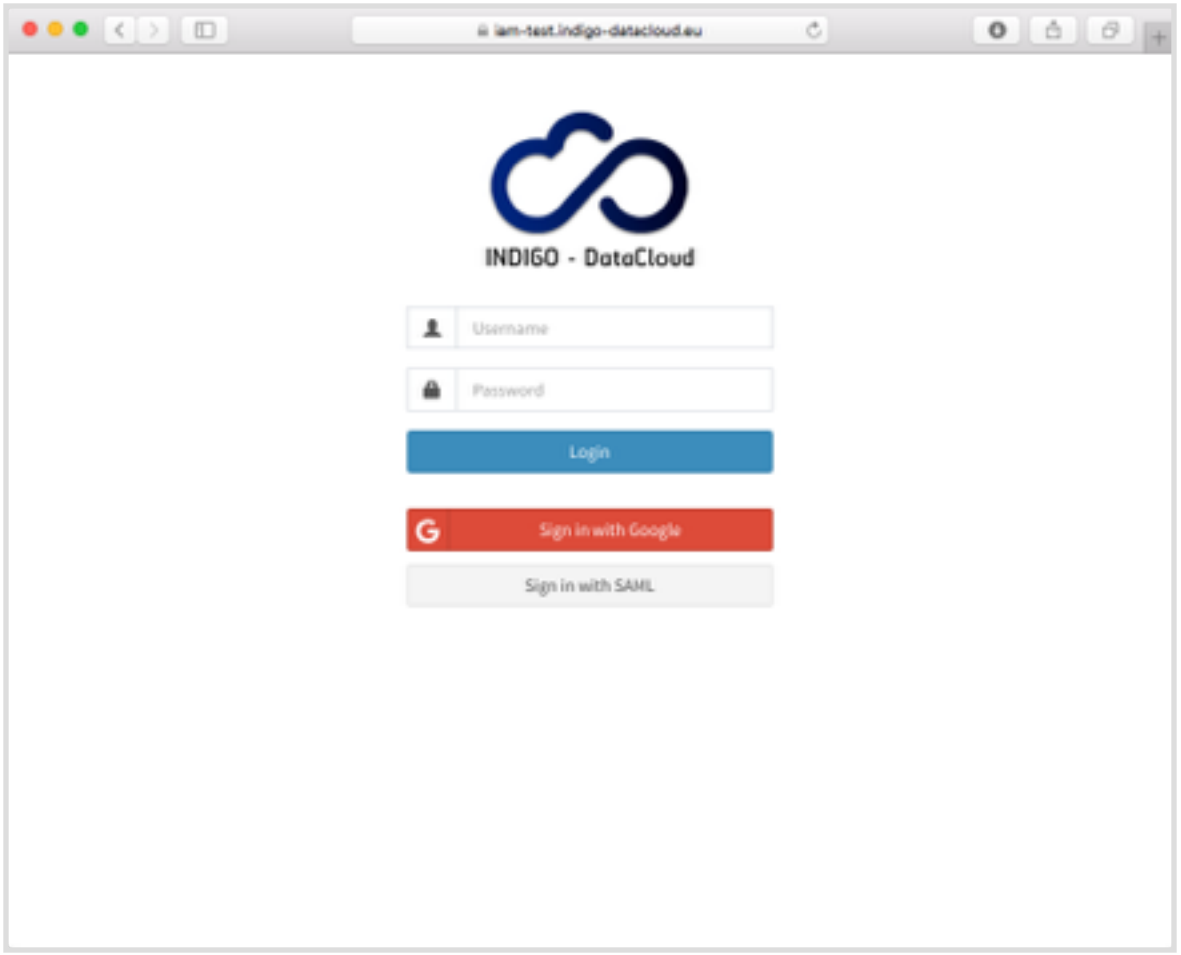
IAM: Goal of the service

- Provide a central service that deals with
  - User authentication (supporting SAML, OIDC, X.509)
  - Identity harmonization (link heterogeneous AuthN mechanisms to a single VO identity)
  - Management of VO membership (i.e., groups and other attributes)
  - Management of registration and enrollment flows
  - Provisioning of VO structure and membership information to services
  - Management, distribution and enforcement of authorization policies



iam-test.indigo-datacloud.eu

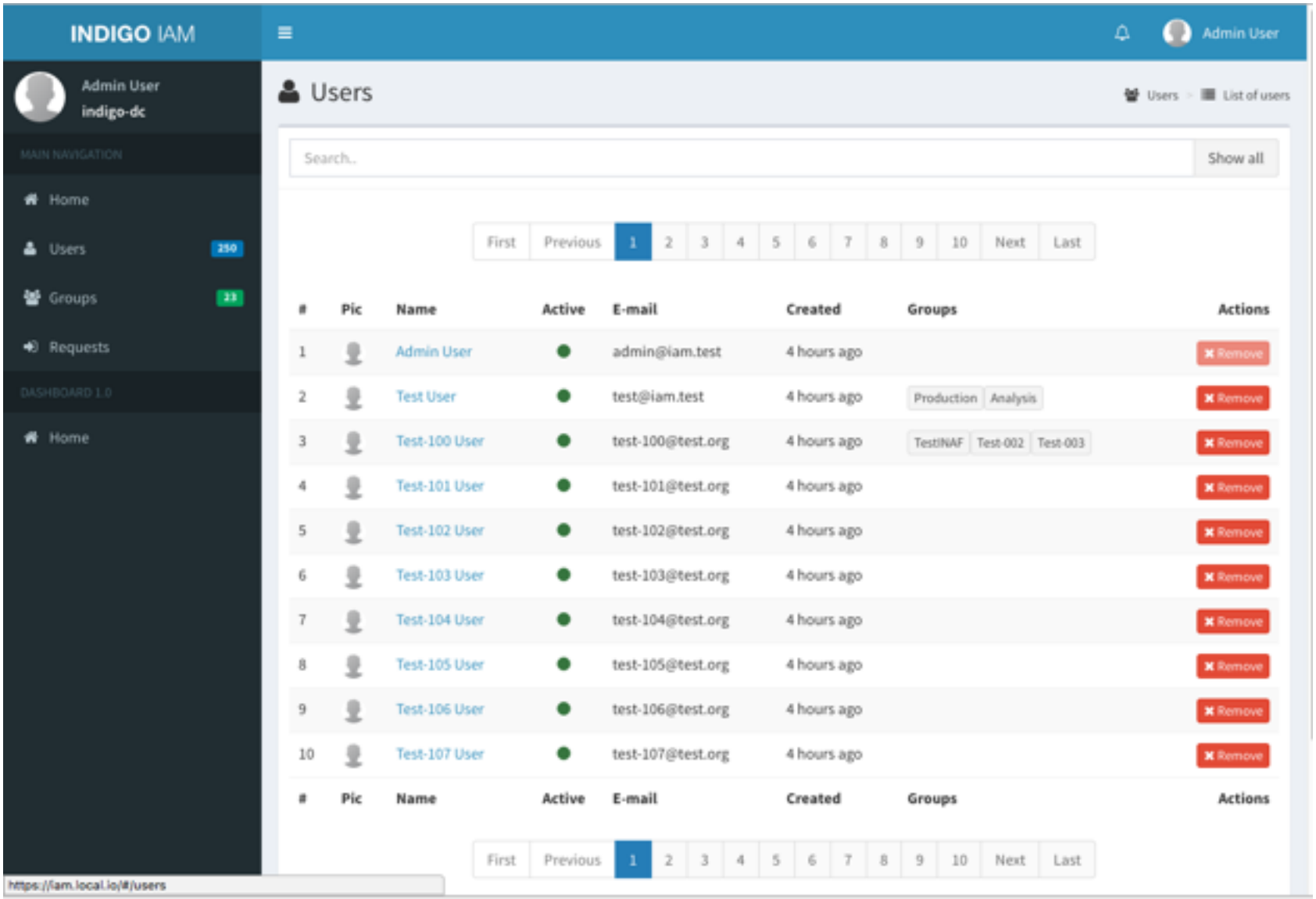
- IAM test instance deployed
  - https://iam-test.indigo-datacloud.eu
- Google AuthN supported
- EduGAIN AuthN supported
  - But your IdP must be configured to release the minimum required set of attributes



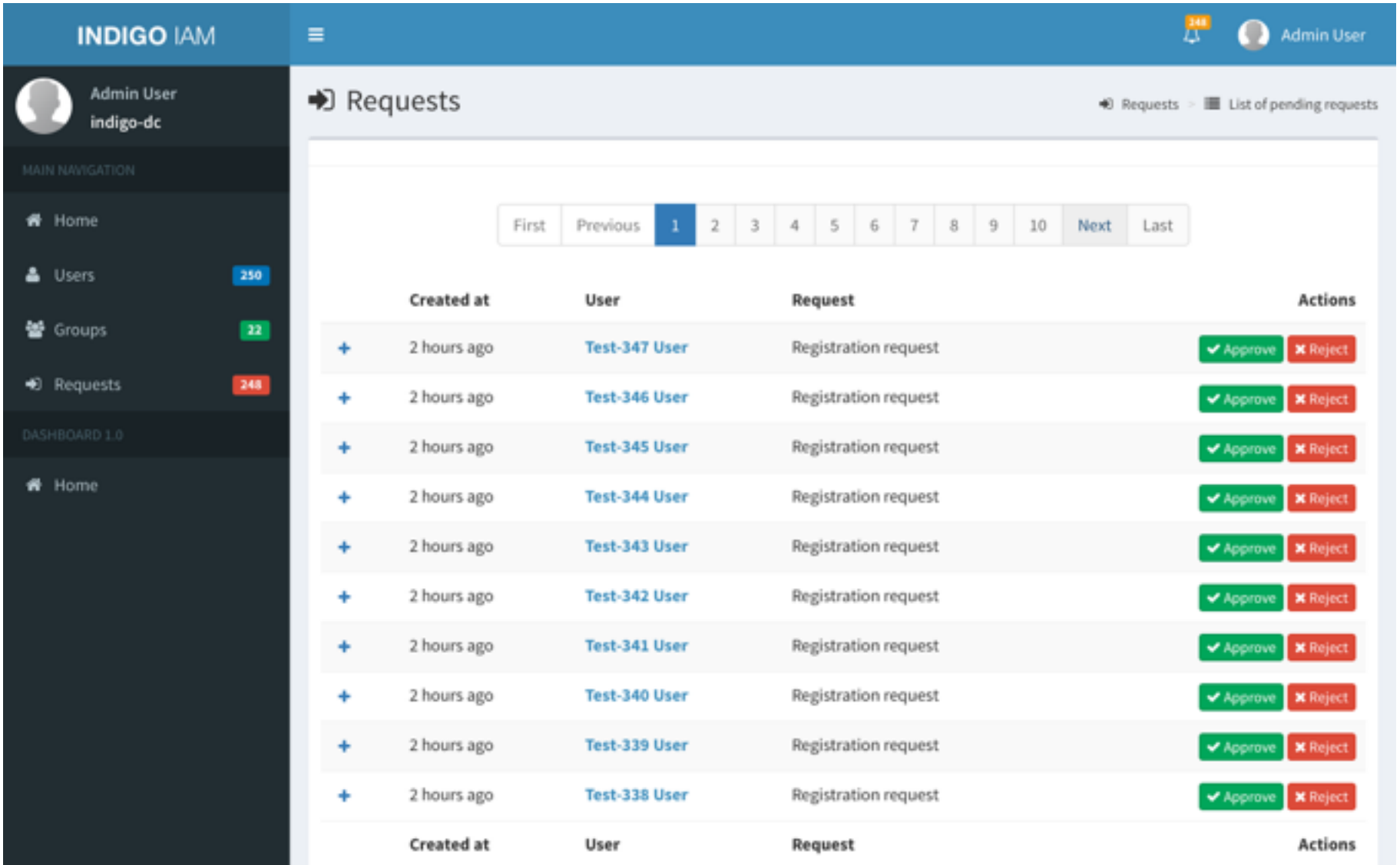
SCIM Provisioning

- Standard SCIM provisioning APIs
- User management
- Group management

IAM Dashboard

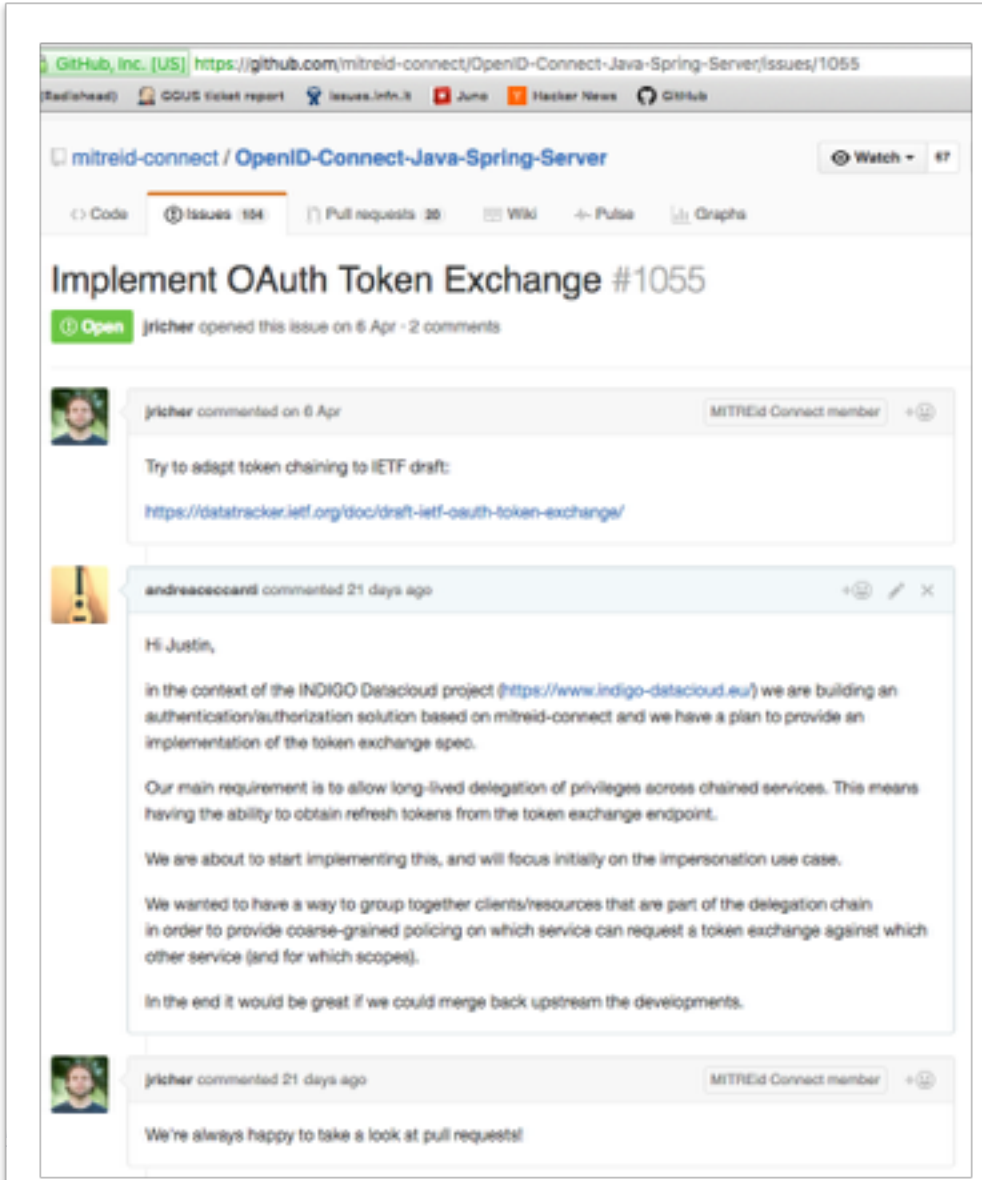


IAM Dashboard (...)



IAM OAuth token exchange

- We have implemented a first incarnation of the support for OAuth token exchange standard in the IAM by extending the MitreID connect library to support or main chained delegation use case
  - i.e. delegate offline access to identity information across services



INDIGO - DataCloud

Contacts and resources

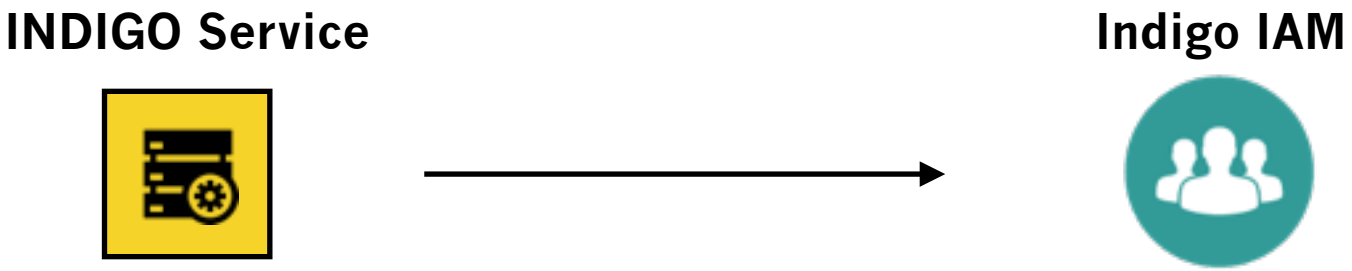
- The AAI-TF wiki:
  - https://project.indigo-datacloud.eu/projects/aai-taskforce/wiki/Wiki
- The AAI-TF mailing list:
  - https://lists.indigo-datacloud.eu/sympa/lists/info/indigo-aai-tf
- The AAI-TF slack room:
  - https://indigo-aai.slack.com/

INDIGO - DataCloud

AuthN flow for services

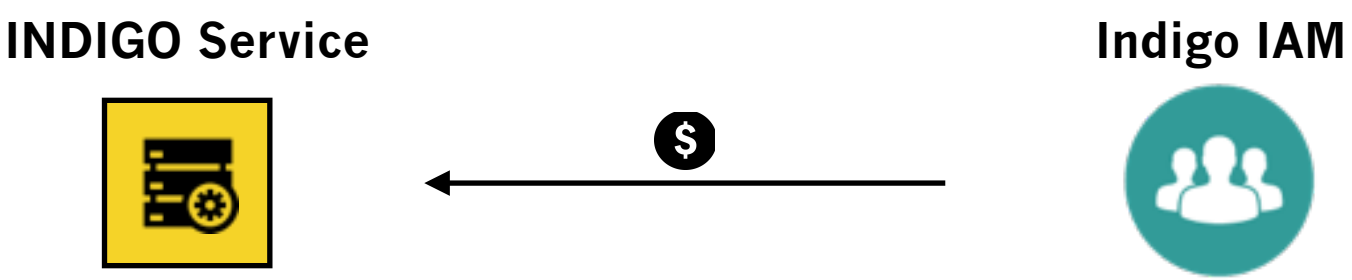
- Sometimes services need to act on behalf of themselves, to implement behavior not linked to a specific user
  - E.g. a garbage collector service that cleans up resources for a group of users
- IAM supports the OAuth client credentials flow that is designed for this type of authentication

Client credentials flow




```
curl -u ${CLIENT_ID}:${CLIENT_SECRET} \
  -d grant_type=client_credentials \
  http://localhost:8080/token
```

Client credentials flow




```
{
  "access_token": "eyJraWQiOiJyc2ExIiwiYWxnIjoiUlMyN
  "token_type": "Bearer",
  "expires_in": 3599,
  "scope": "read-tasks write-tasks openid profile"
}
```

INDIGO - DataCloud

AuthN flow for CLIs

- IAM supports the OAuth Resource owner password credentials flow
- A registered client that has such flow enabled can request a token directly form the IAM and then use the token to act on behalf of the user

INDIGO - DataCloud

Scope-based Authorization flow

INDIGO - DataCloud

Scope-based authorization

- Each service registers the supported scopes when it registers at the authorization server (AS)
- The AS maintains policies that determine which client is authorized to request a given scope
- The request for a given scope is authorized by the user through the OAuth consent mechanism
  - but is possible to define trusted, whitelisted client services for which user consent is not requested
- Authorization is enforced at the target service considering scopes and other relevant information

Scope-based authorization

Access SG

SG

Science Gateway

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Job Scheduler

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Redirect
for AuthN
& consent

Job Scheduler

SG

Science Gateway

Storage Service

Scope-based authorization

Redirect
for AuthN
& consen

SG

Science Gateway

Indigo IAM

Registered Scopes

```
js:submit_job
js:cancel_job

ss:read
ss:write
```

Authorization requested

Scientific Gateway would like to:

know your identity
submit jobs on JS on your behalf
read files from SS on your behalf
write files on SS on your behalf

Deny **Authorize**

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Returned
id token &
access token

Job Scheduler

SG

Science Gateway

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Submit job

$

Job Scheduler

SG

Science Gateway

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Submit job

$

Job Scheduler

Job scheduling is authorized by the js:submit_job scope

SG

Science Gateway

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Job Scheduler

Read job output data

$

SG

Science Gateway

Storage Service

Scope-based authorization

Indigo IAM

Registered Scopes

`js:submit_job`
`js:cancel_job`

`ss:read`
`ss:write`

Job Scheduler

Data access is authorized by the ss:read scope

Read job output data

$

SG

Science Gateway

Storage Service

INDIGO - DataCloud

Fine-grained authorization

- OAuth scope-based AuthZ provides a first coarse grained authorization step
- Finer-grained authorization can be implemented at services on top of this step taking into account
  - User identity attributes
  - Service authorization policies
  - Collaboration/VO policies
- Consistent authorization across services is enabled by callouts to the Argus authorization service

Fine-grained authorization