INDIGO IAM migration to Spring Authorization Server framework with a new customizable React user dashboard

Jacopo Gasparetto
INFN - CNAF

ICSC
Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

International Symposium on Grids & Clouds (ISGC)
24-29 March 2024

INFN
CNAF

INDIGO Identity and Access Management Service

First developed in the context of the **H2020 INDIGO DataCloud** project

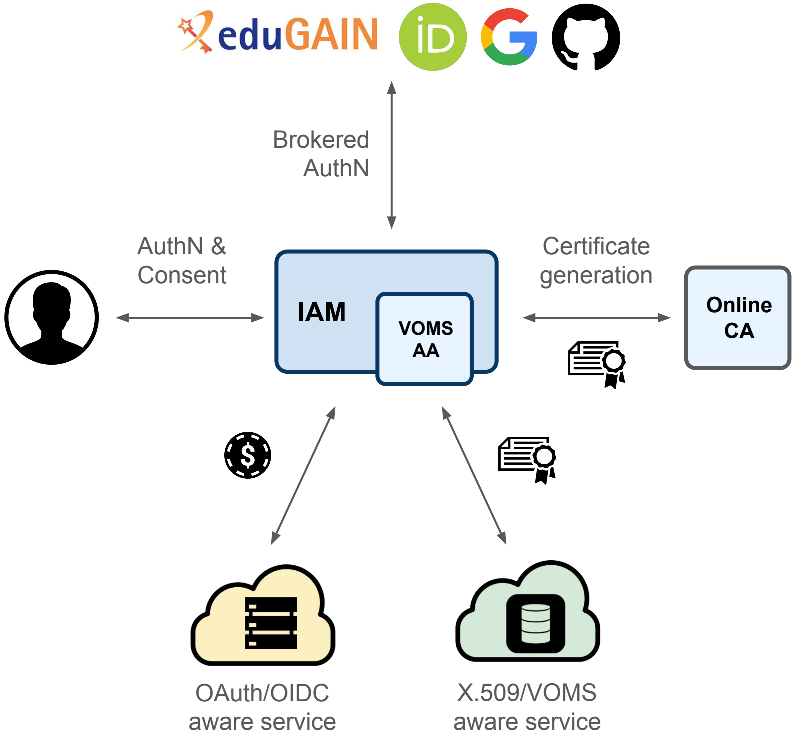
~8 years since 1st INDIGO IAM release v0.3.0

Selected by the WLCG management board to be the core of the future, token-based WLCG AAI

INFN commitment for the foreseeable future, with the current support of several Italian and European projects:

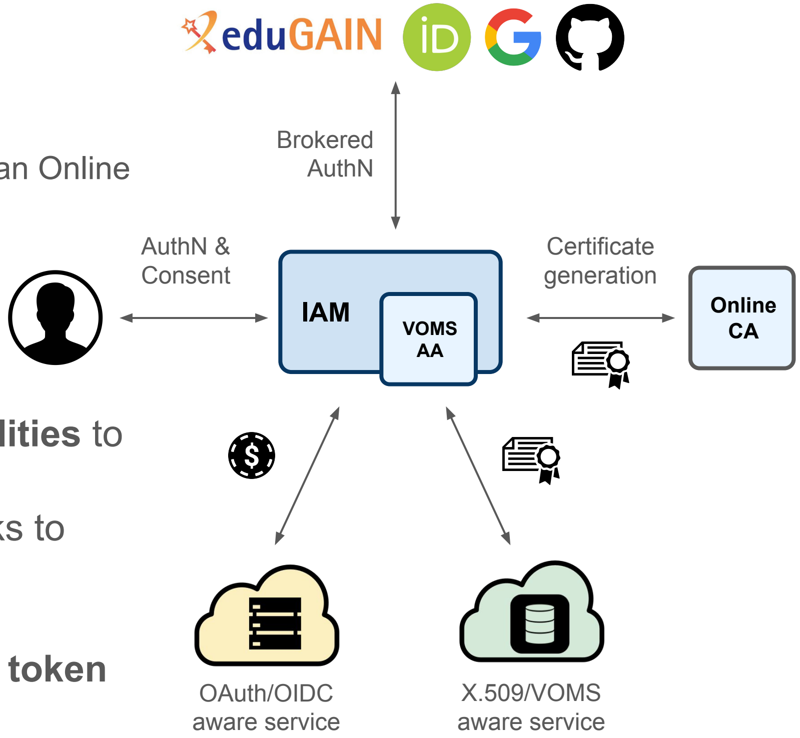
eosc | BEYOND

AARC   ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing   terabit



INDIGO Identity and Access Management Service

- Supports **multiple authentication mechanisms**
    - SAML, X.509, OpenID Connect, local users, etc.
- Supports **account linking**
    - also SSH RSA keys and a certificate generated through an Online CA can be linked
- Provides a **registration service** for moderated and automatic user enrollment
    - it can be disabled
- Enforcement of **AUP acceptance**
- Exposes **identity information**, **attributes** and **capabilities** to services via **JWT** tokens
- **Easy integration** with ready-to-use components thanks to OpenID Connect/OAuth
- Can integrate existing **VOMS**-aware services
- Supports **Web** and **non-Web access**, **delegation** and **token renewal**



IAM core technologies

IAM is a **Spring Boot** application

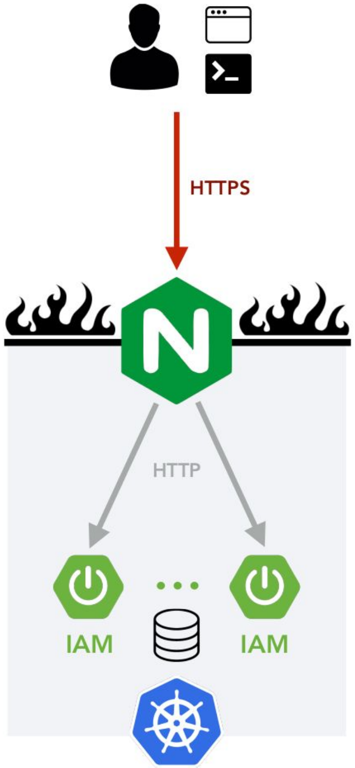
- OIDC/OAuth 2.0 implementation currently based on the MitreID Connect
- deployed behind an **NGINX**
- stores data in a **MariaDB/MySQL** database

Horizontally scalable

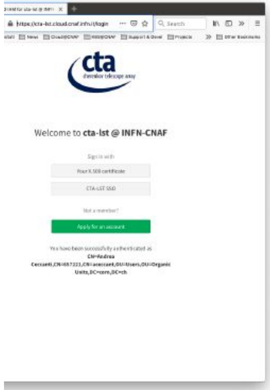
- sessions and external caching stored into Redis

We deploy IAM as a **containerized** service on top of **Kubernetes**

- autoscaling, zero downtime rolling updates



IAM deployments at CNAF


https://cta-lst.cloud.cnaf.infn.it/login

Welcome to **cta-lst @ INFN-CNAF**

Sign in with

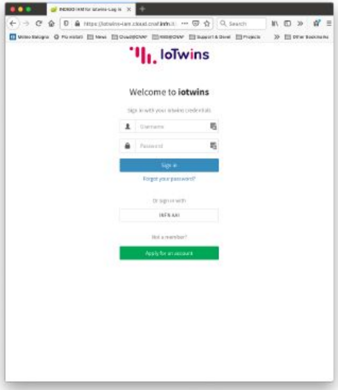
Your X.509 certificate

CTA-LST SSO

Not a member?

Apply for an account

You have been successfully authenticated as
CN=Andrea Ceccanti,CN=657321,CN=aceccant,OU=Users,OU=Organic Units,DC=cern,DC=ch


INDIGO IAM for iotwins - Log in

https://iotwins-iam.cloud.cnaf.infn.it

Welcome to **iotwins**

Sign in with your iotwins credentials

Username

Password

Sign in

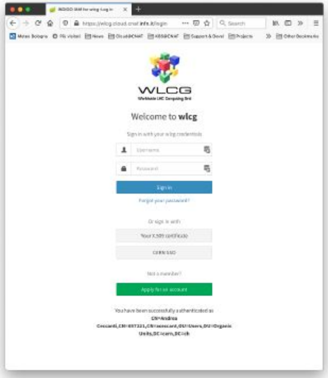
Forgot your password?

Or sign in with

INFN AAI

Not a member?

Apply for an account


INDIGO IAM for wlcg - Log in

https://wlcg.cloud.cnaf.infn.it/login

Welcome to **wlcg**

Sign in with your wlcg credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

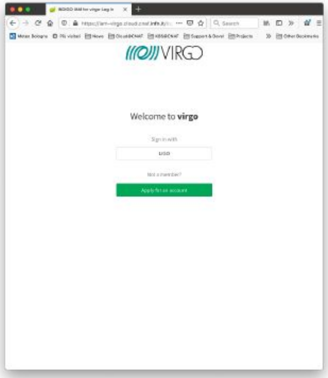
Your X.509 certificate

CERN SSO

Not a member?

Apply for an account

You have been successfully authenticated as
CN=Andrea Ceccanti,CN=657321,CN=aceccant,OU=Users,OU=Organic Units,DC=cern,DC=ch


INDIGO IAM for virgo - Log in

https://iam-virgo.cloud.cnaf.infn.it/

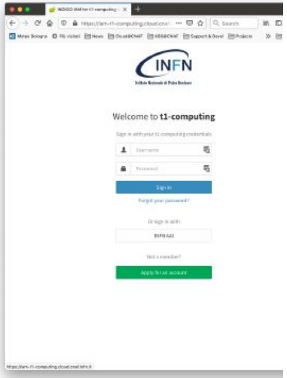
Welcome to **virgo**

Sign in with

LIGO

Not a member?

Apply for an account


INDIGO IAM for t1-computing

https://iam-t1-computing.cloud.cnaf.infn.it

Welcome to **t1-computing**

Sign in with your t1-computing credentials

Username

Password

Sign in

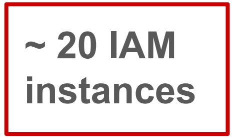
Forgot your password?

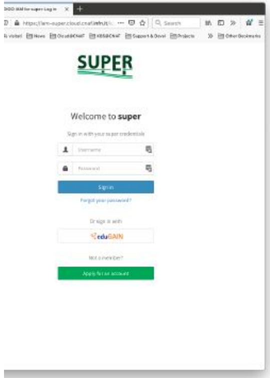
Or sign in with

INFN AAI

Not a member?

Apply for an account


~ 20 IAM instances


INDIGO IAM for super - Log in

https://iam-super.cloud.cnaf.infn.it/

Welcome to **super**

Sign in with your super credentials

Username

Password

Sign in

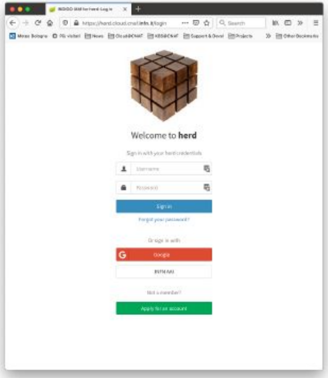
Forgot your password?

Or sign in with

eduGAIN

Not a member?

Apply for an account


INDIGO IAM for herd - Log in

https://herd.cloud.cnaf.infn.it/login

Welcome to **herd**

Sign in with your herd credentials

Username

Password

Sign in

Forgot your password?

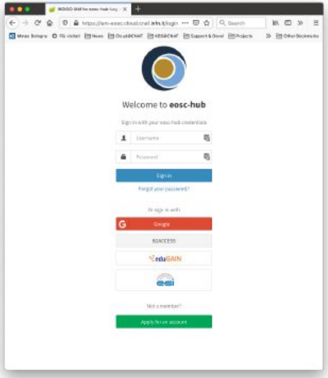
Or sign in with

Google

INFN AAI

Not a member?

Apply for an account


INDIGO IAM for eosc-hub - Log in

https://iam-eosc.cloud.cnaf.infn.it/login

Welcome to **eosc-hub**

Sign in with your eosc-hub credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

Google

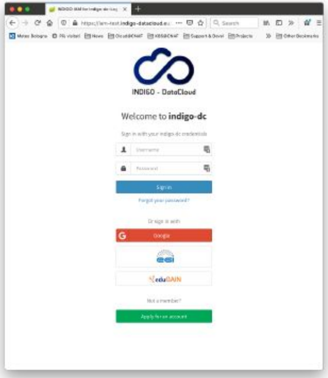
B2ACCESS

eduGAIN

egi

Not a member?

Apply for an account


INDIGO IAM for indigo-dc - Log in

https://iam-test.indigo-datacloud.eu

Welcome to **indigo-dc**

Sign in with your indigo-dc credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

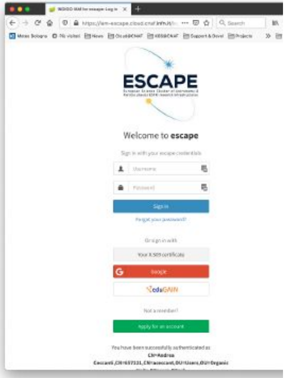
Google

egi

eduGAIN

Not a member?

Apply for an account


INDIGO IAM for escape - Log in

https://iam-escape.cloud.cnaf.infn.it/

Welcome to **escape**

Sign in with your escape credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

Your X.509 certificate

Google

eduGAIN

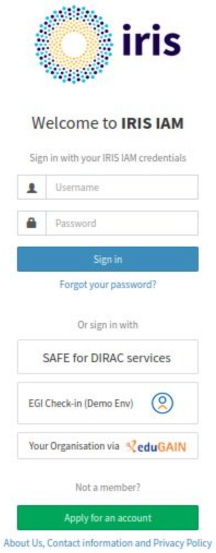
Not a member?

Apply for an account

You have been successfully authenticated as
CN=Andrea Ceccanti,CN=657321,CN=aceccant,OU=Users,OU=Organic

IAM deployments outside CNAF

~ 10 IAM instances



Welcome to **IRIS IAM**

Sign in with your IRIS IAM credentials

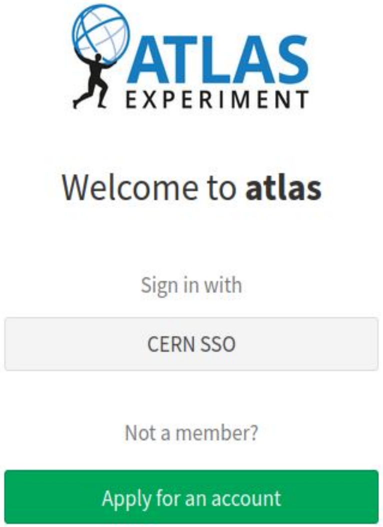
Username

Password

Or sign in with

Not a member?

iris-iam.stfc.ac.uk

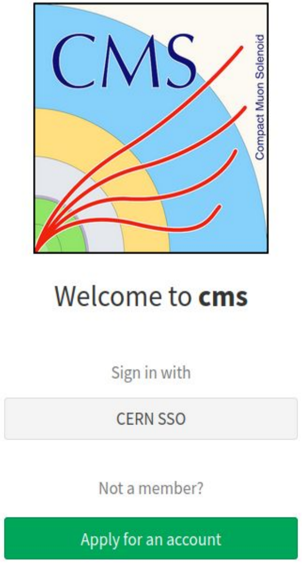


Welcome to **atlas**

Sign in with

Not a member?

atlas-auth.web.cern.ch

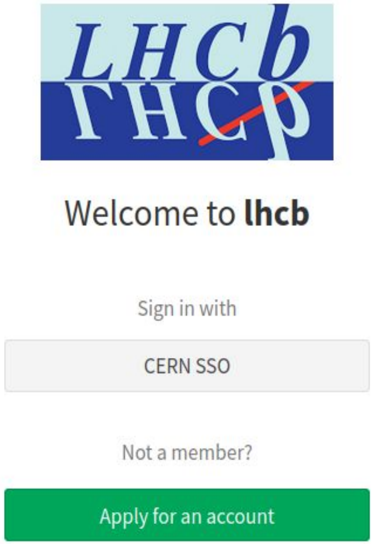


Welcome to **cms**

Sign in with

Not a member?

cms-auth.web.cern.ch

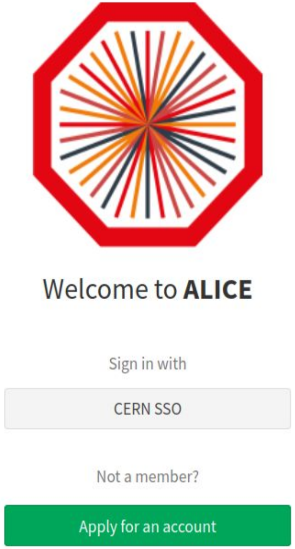


Welcome to **lhcb**

Sign in with

Not a member?

lhcb-auth.web.cern.ch



Welcome to **ALICE**

Sign in with

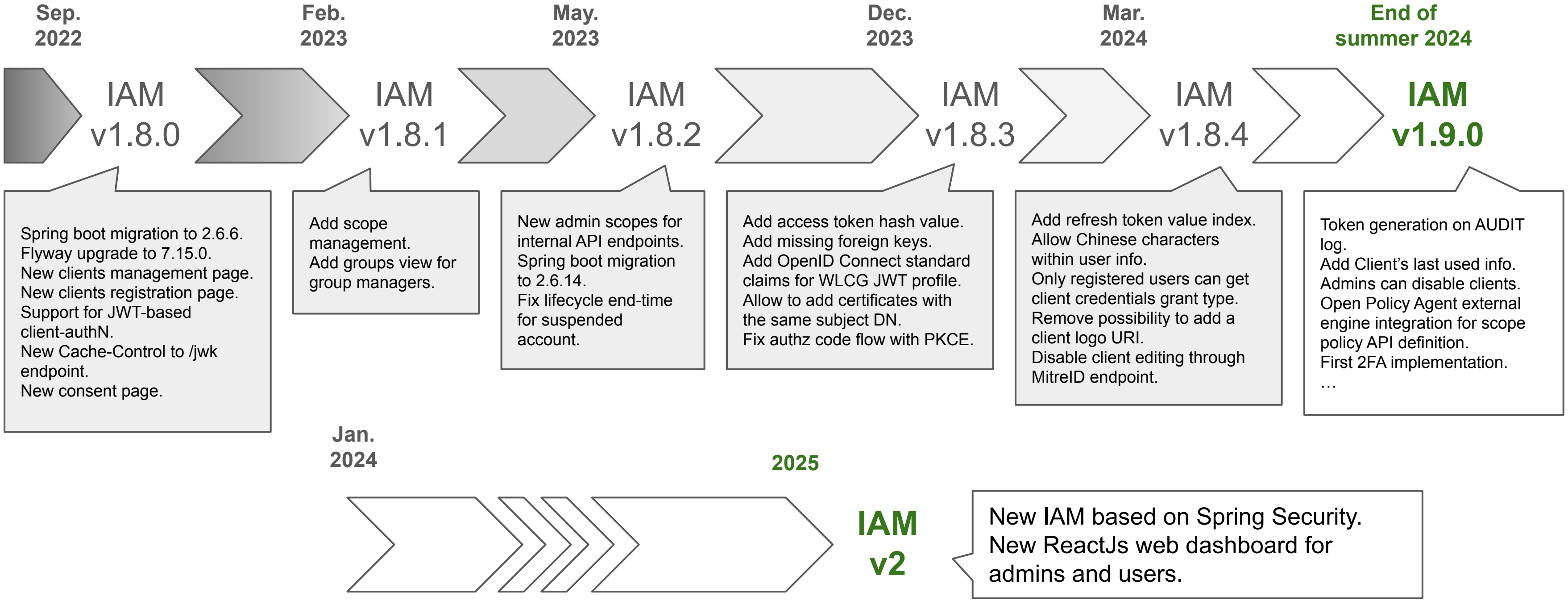
Not a member?

alice-auth.web.cern.ch

Releases roadmap

Latest release IAM v1.8.4 - released on **2024-03-25**

Changelog



Sep. 2022 — IAM v1.8.0

Spring boot migration to 2.6.6.
Flyway upgrade to 7.15.0.
New clients management page.
New clients registration page.
Support for JWT-based client-authN.
New Cache-Control to /jwk endpoint.
New consent page.

Feb. 2023 — IAM v1.8.1

Add scope management.
Add groups view for group managers.

May. 2023 — IAM v1.8.2

New admin scopes for internal API endpoints.
Spring boot migration to 2.6.14.
Fix lifecycle end-time for suspended account.

Dec. 2023 — IAM v1.8.3

Add access token hash value.
Add missing foreign keys.
Add OpenID Connect standard claims for WLCG JWT profile.
Allow to add certificates with the same subject DN.
Fix authz code flow with PKCE.

Mar. 2024 — IAM v1.8.4

Add refresh token value index.
Allow Chinese characters within user info.
Only registered users can get client credentials grant type.
Remove possibility to add a client logo URI.
Disable client editing through MitreID endpoint.

End of summer 2024 — **IAM v1.9.0**

Token generation on AUDIT log.
Add Client's last used info.
Admins can disable clients.
Open Policy Agent external engine integration for scope policy API definition.
First 2FA implementation.
…

Jan. 2024

2025 — **IAM v2**

New IAM based on Spring Security.
New ReactJs web dashboard for admins and users.

Current main development targets

- **Auditing** improvements
- **Superseded obsolete dependencies**
  - MitreID → Spring Authorization Server
  - AngularJS → React JS
- Improve **usability** for users & admins
- **Scalability and Performances** improvements
  - Access tokens not stored on database
  - Dedicated garbage collector service
  - Fine grained AuthZ with Open Policy Agent
- **Interoperability** focus
  - Support OIDC Federations
  - Improve conformance with AARC BluePrint Architecture and its guidelines
- **Security**
  - Add Multi-Factor Authentication (MFA)

Migration to
Spring Authorization Server

Spring Authorization Server

Spring Authorization Server is a framework, built on top of **Spring Security,** that provides a secure, lightweight and customizable foundation for building an **OAuth 2.1** and **OpenID Connect 1.0** Authorization Server implementation.

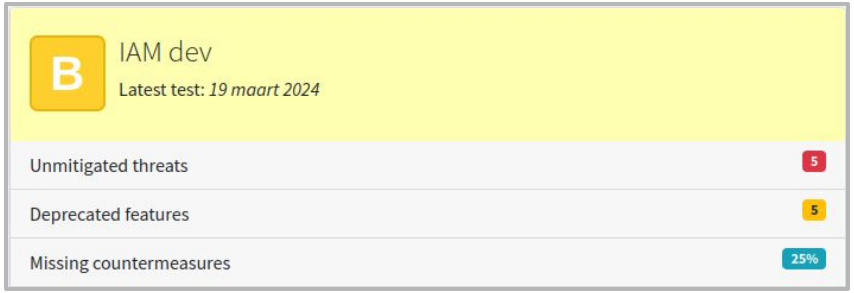
Why?

- We still rely on a forked and self-maintained version of MitreID Connect library which has no substantial support/evolution since few years
- It’s a natural evolution of the current architecture Java/Spring-based
- Long-term support and easier maintainability
- Better OIDC/OAuth standards compliance
  - Compliance with OAuth 2.1 standard

OIDC/OAuth standards compliance

Tested with OAuch.io

Where we are…



B

IAM dev

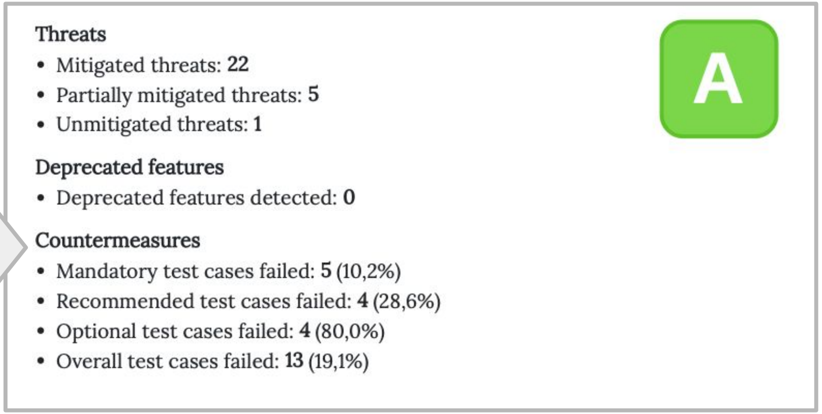
Latest test: *19 maart 2024*

Unmitigated threats	5
Deprecated features	5
Missing countermeasures	25%

Based on MitreID Connect library and OAuth 2.0 standard. OAuth 2.1 tests excluded because not supported.

First tests done with a rough application built on top of Spring Authorization Server

- already supports many OAuth standard grants
- many OIDC/OAuth endpoints are supported by default
- tests in progress



A

Threats

- Mitigated threats: **22**
- Partially mitigated threats: **5**
- Unmitigated threats: **1**

Deprecated features

- Deprecated features detected: **0**

Countermeasures

- Mandatory test cases failed: **5** (10,2%)
- Recommended test cases failed: **4** (28,6%)
- Optional test cases failed: **4** (80,0%)
- Overall test cases failed: **13** (19,1%)

New Dashboard
A React based web application

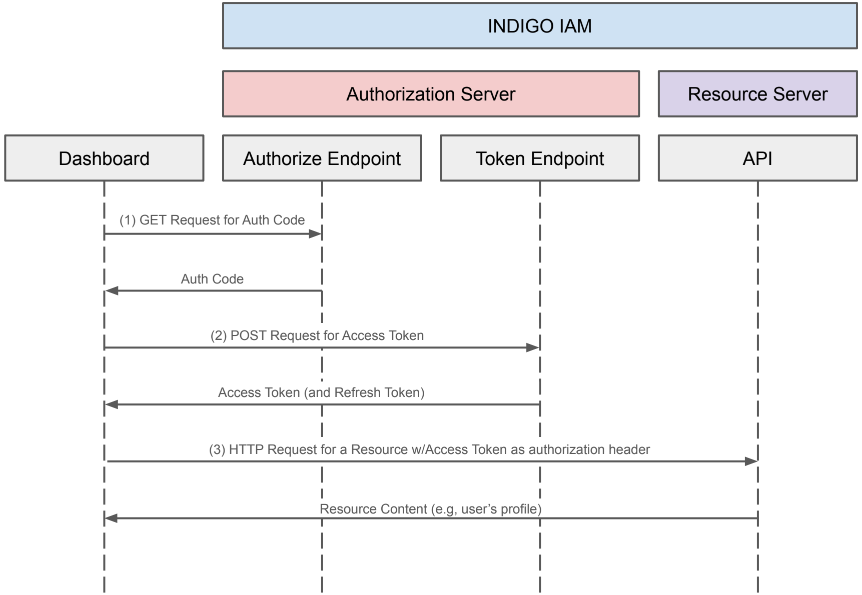
New Dashboard: a React based web-application

Motivation

- Remove AngularJS (EOF) and JavaServer Pages (JSPs)
- Full support of modern **HTML5** / **TypeScript** / **CSS** development stack based
- **Decouple** the frontend code from the INDIGO IAM codebase
- Handle AuthN/AuthZ via **OpenID Connect** and **OAuth2** frameworks
- Modern and lightweight rendering framework (**React**)
- **Customizable** by different organizations
- Reuse of standard and custom web components
- **Styles harmonization** for all future INFN web applications

Implementation details (Proof of Concept)

- Full browser-based application
- Public IAM client
- AuthN/AuthZ responsabilites managed by the web application
- OAuth2 Authorization Code flow (RFC6749) w/PKCE* extension (RFC7636)
- Requests to the INDIGO IAM endpoints authenticated via the obtained JWT access token
- Absence of any cookie-based session
- INDIGO IAM plays both the roles of Authorization Server and Resource Server

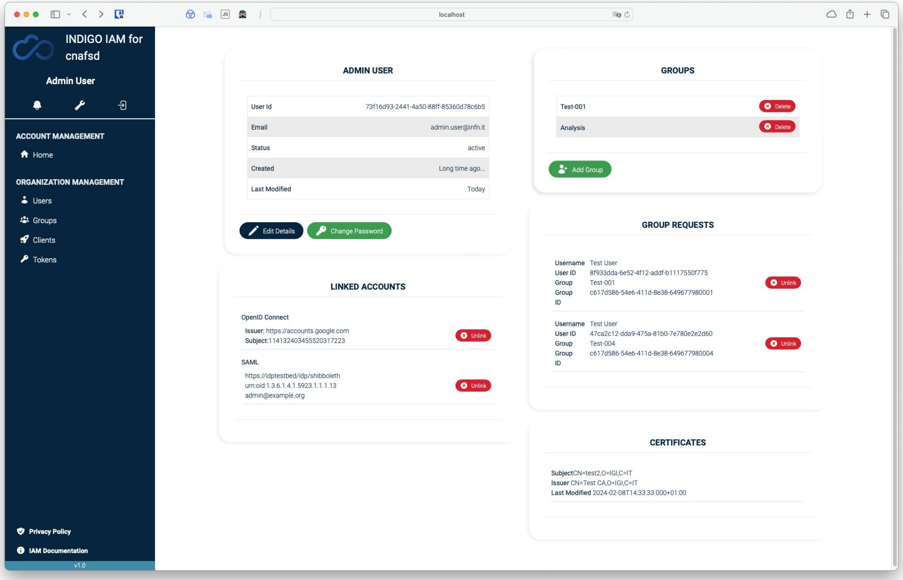

INDIGO IAM
Authorization Server
Resource Server
Dashboard
Authorize Endpoint
Token Endpoint
API
(1) GET Request for Auth Code
Auth Code
(2) POST Request for Access Token
Access Token (and Refresh Token)
(3) HTTP Request for a Resource w/Access Token as authorization header
Resource Content (e.g, user's profile)

OAuth2 Authorization Code flow (PKCE is not shown in figure)

* Proof for Key Code Exchange

Where we are: Proof of Concept

- Simple and lightweight
- Fully executed within the browser as Single Page App (SPA)
- Straightforward deployment as a Docker image derived from NGINX
- Highly scalable
- Currently a demo version is deployed on our development Kubernetes cluster using Argo CD
- GitHub Source



ADMIN USER

User Id	73f16d93-2441-4a50-88ff-85360d78c6b5
Email	admin.user@infn.it
Status	active
Created	Long time ago...
Last Modified	Today

LINKED ACCOUNTS

OpenID Connect

Issuer: https://accounts.google.com
Subject:114132403455520317223

SAML

https://idptestbed/idp/shibboleth
urn:oid:1.3.6.1.4.1.5923.1.1.1.13
admin@example.org

GROUPS

Test-001

Analysis

GROUP REQUESTS

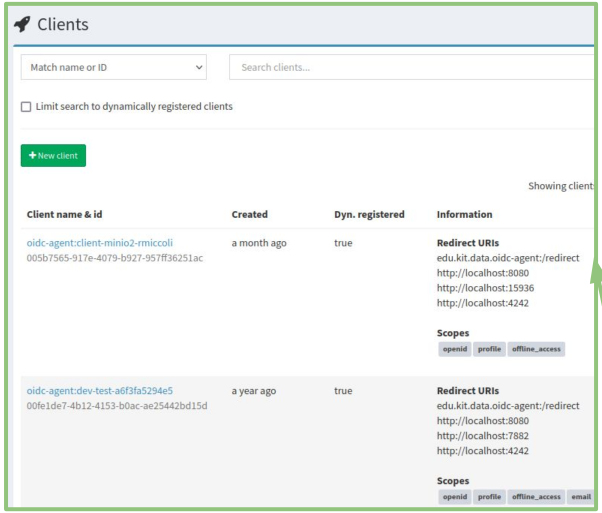
Username	Test User
User ID	8f933dda-6e52-4f12-addf-b1117550f775
Group	Test-001
Group ID	c617d586-54e6-411d-8e38-649677980001
Username	Test User
User ID	47ca2c12-dda9-475a-81b0-7e780e2e2d60
Group	Test-004
Group ID	c617d586-54e6-411d-8e38-649677980004

CERTIFICATES

SubjectCN=test2,O=IGI,C=IT
Issuer CN=Test CA,O=IGI,C=IT
Last Modified 2024-02-08T14:33:33.000+01:00

Homepage example

Current dashboard



Clients

Match name or ID

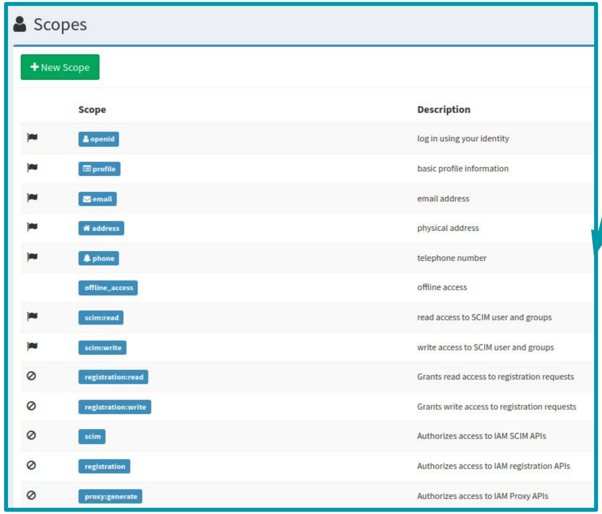
Search clients...

□ Limit search to dynamically registered clients

+ New client

Showing client

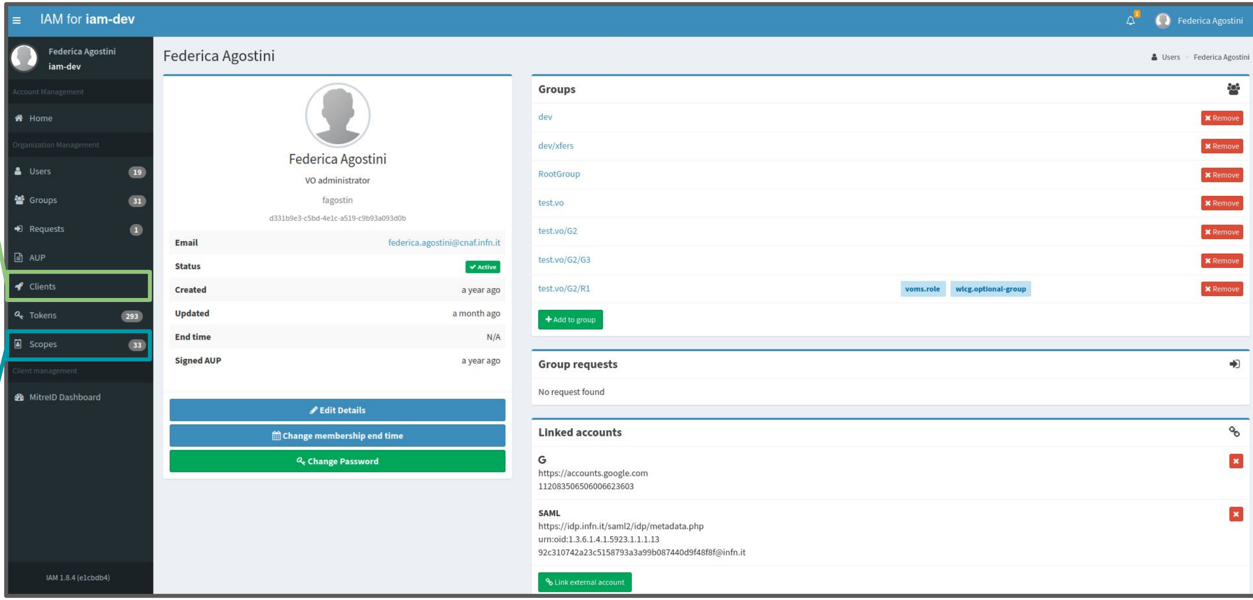
Client name & id	Created	Dyn. registered	Information
oidc-agent:client-minio2-rmiccoli 005b7565-917e-4079-b927-957ff36251ac	a month ago	true	**Redirect URIs** edu.kit.data.oidc-agent:/redirect http://localhost:8080 http://localhost:15936 http://localhost:4242 **Scopes** openid profile offline_access
oidc-agent:dev-test-a6f3fa5294e5 00fe1de7-4b12-4153-b0ac-ae25442bd15d	a year ago	true	**Redirect URIs** edu.kit.data.oidc-agent:/redirect http://localhost:8080 http://localhost:7882 http://localhost:4242 **Scopes** openid profile offline_access email



Scopes

+ New Scope

	Scope	Description
⚑	openid	log in using your identity
⚑	profile	basic profile information
⚑	email	email address
⚑	address	physical address
⚑	phone	telephone number
	offline_access	offline access
⚑	scim:read	read access to SCIM user and groups
⚑	scim:write	write access to SCIM user and groups
⊘	registration:read	Grants read access to registration requests
⊘	registration:write	Grants write access to registration requests
⊘	scim	Authorizes access to IAM SCIM APIs
⊘	registration	Authorizes access to IAM registration APIs
⊘	proxy:generate	Authorizes access to IAM Proxy APIs



Federica Agostini

Federica Agostini

VO administrator

fagostin

d331b9e3-c5bd-4e1c-a519-c9b93a093d0b

Email	federica.agostini@cnaf.infn.it
Status	✓ Active
Created	a year ago
Updated	a month ago
End time	N/A
Signed AUP	a year ago

Edit Details

Change membership end time

Change Password

Groups

dev		✖ Remove
dev/xfers		✖ Remove
RootGroup		✖ Remove
test.vo		✖ Remove
test.vo/G2		✖ Remove
test.vo/G2/G3		✖ Remove
test.vo/G2/R1	voms.role wlcg.optional-group	✖ Remove

+ Add to group

Group requests

No request found

Linked accounts

G
https://accounts.google.com
112083506506006623603

SAML
https://idp.infn.it/saml2/idp/metadata.php
urn:oid:1.3.6.1.4.1.5923.1.1.1.13
92c310742a23c5158793a3a99b087440d9f48f8f@infn.it

Link external account

Security Concerns

- The web application is **a public OAuth2 client** and thus cannot have secure secrets
- Access Token and Refresh Tokens available to the JavaScript code, but this is not considered a recommended practice
  - An attacker can more easily gain direct control over the access token and send legitimate requests to the Resource Server on behalf of the legitimate owner (i.e., gather users' information, edit users and groups etc)
  - Similarly, an attacker could exploit a silent Refresh Flow to obtain a new fresh set of tokens
- Risk of scope escalation if not handled properly with policies

Future outlooks

Possible scenarios

- Static website with Backend
  - **Backend For Frontend (BFF)**: the backend handles all OAuth2 responsibilities and proxies requests to the Resource Server without exposing any token to the browser
  - Mediating-Token Backend: the backend handles all OAuth2 responsibilities and return an access token to the browser, which will perform authorized requests to the Resource Server
- Server-side rendering
  - All OAuth2 responsibilities are handled by the backend
  - Rendering and computations completely run on the backend server exposing only the final HTML content
  - Requires the usage of a complex framework, such as Next.js
  - This is the current architecture

Source: https://datatracker.ietf.org/doc/draft-ietf-oauth-browser-based-apps/

Conclusions

INDIGO IAM is a critical service widely adopted by many scientific communities. Our evolution roadmap includes:

- Migration to Spring Authorization Server
    - Go beyond the unsupported MitreID Connect library
    - Better compliance with OIDC / OAuth 2.1 standards
    - Rely on a more maintained and supported framework
- Development of a new dashboard
    - Go beyond old AngularJS based web user interface
    - Decouple frontend codebase from INDIGO IAM
    - Explore modern solutions to handle securely both the critical operations, such as the OAuth flows, and the critical endpoints (API)
    - A successful attempt of a Single-Page App (SPA) built in React, proved to be a good candidate to replace the current INDIGO IAM dashboard

Source: https://datatracker.ietf.org/doc/draft-ietf-oauth-browser-based-apps/

Many thanks to all the contributors

Federica Agostini, Roberta Miccoli, Enrico Vianello,
Stefano Zotti, Francesco Giacomini

Bkp

Core technologies in AAI

- **OAuth 2**
  - A standard framework for **delegated authorization**
  - Widely adopted in industry
  - Main specification is RFC 6749
- **OpenID Connect (OIDC)**
  - An **authentication** layer built on top of OAuth 2
  - Core specification
- **JSON Web Tokens (JWTs)**
  - A **compact, URL-safe** means of representing attributes (**claims**) to be transferred between two or more parties
  - Main specification is RFC 7519

BUNDESREPUBLIK DEUTSCHLAND
FEDERAL REPUBLIC OF GERMANY / REPUBLIQUE FEDERALE D'ALLEMAGNE
PERSONALAUSWEIS
IDENTITY CARD / CARTE D'IDENTITE

T22000129

Name/Surname/Nom
MUSTERMANN
GEB. GABLER

Vornamen/Given names/Prénoms
ERIKA

Geburtstag/Date of birth/Date de naissance
12.08.1964

Staatsangehörigkeit/Nationality/Nationalité
DEUTSCH

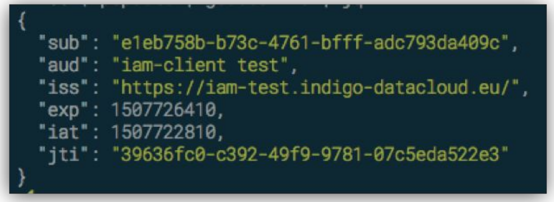
Geburtsort/Place of birth/Lieu de naissance
BERLIN

Gültig bis/Date of expiry/Date d'expiration
31.10.2020

938568

Unterschrift der Inhaberin/des Inhabers - Signature of bearer - Signature de la titulaire/du titulaire

MUSTER



```
{
  "sub": "e1eb758b-b73c-4761-bfff-adc793da409c",
  "aud": "iam-client test",
  "iss": "https://iam-test.indigo-datacloud.eu/",
  "exp": 1507726410,
  "iat": 1507722810,
  "jti": "39636fc0-c392-49f9-9781-07c5eda522e3"
}
```

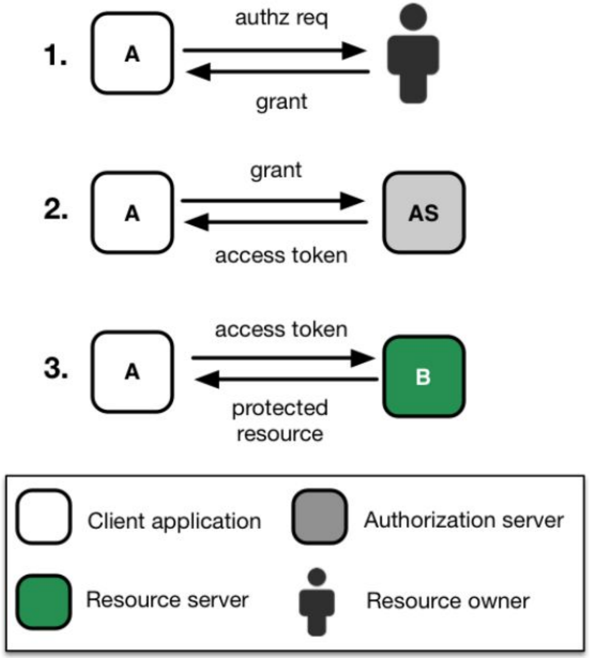
OAuth 2 roles

- **Resource owner**
    - A **user** that owns resources hosted at a service
- **Client**
    - An **application** that wants to have delegated access to user resources
    - It has to be registered on the Authorization Server
    - *Relying Party* (RP) in OIDC
- **Authorization Server (AS)**
    - A service that authenticates users and Clients
    - It **issues tokens** to Clients that can be used to access user resources
    - *OpenID Provider* (OP) in OIDC
- **Resource Server (RS)**
    - A service that **holds protected resources** (*e.g.*, user data)
    - It grants access based on tokens issued by the Authorization Server and presented by a Client
    - It has to validate the access token
    - Not mandatory to register a RS on the Authorization Server

The Authorization Server may be the same as the Resource Server

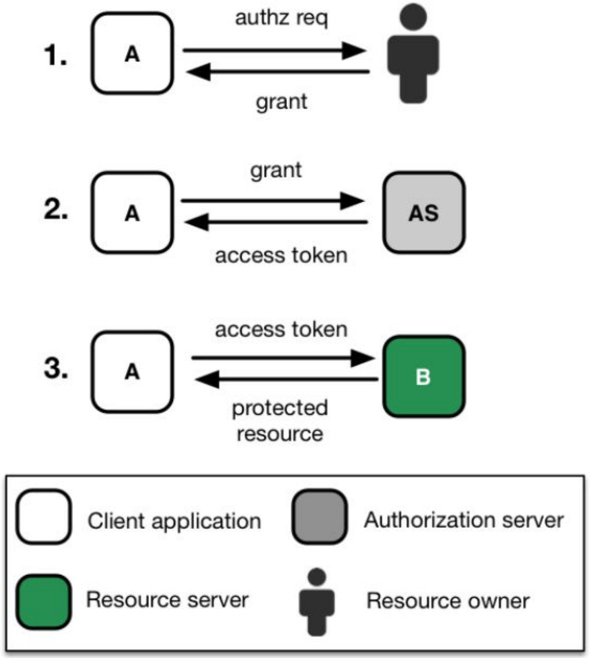
Authorization flow in theory

1. Authorization request to the resource owner
   - The Client (**A**) requests authorization from the resource owner to access a resource within a defined **scope**
     - the authorization request can be performed indirectly via the Authorization Server (**AS**)
   - The Client receives an **authorization grant**, which is a credential representing the resource owner's authorization
     - it depends on the authorization flow (aka *grant type*) used by the Client to perform the authorization requests
2. Authorization request to the AS token endpoint
   - The Client requests for an **access token** by authenticating with the AS and presenting the authorization grant
     - additional tokens can be requested at this stage



Authorization flow in theory

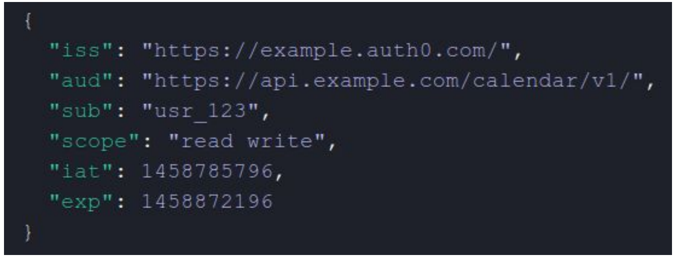
3. Access to the protected resource
   - The Client requests the protected resource from the Resource Server (**B**) and authenticates by presenting the access token
   - The RS validates the access token, and if valid, serves the request
   - Access is granted/denied according to the contents of the access token
     - local policies that map token claims into permissions may be applied by the RS



OAuth/OIDC token types

Access Token (AT)

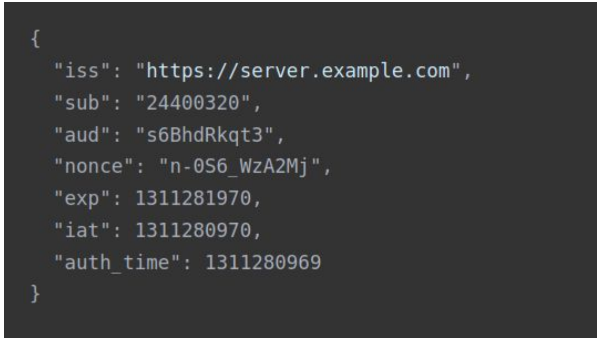
- Defined within OAuth 2
- Is a string that the Client uses to make requests to the Resource Server
  - do not have to be in any particular format
- AT may be *bearer tokens*, meaning that those who hold the token can use it



```
{
  "iss": "https://example.auth0.com/",
  "aud": "https://api.example.com/calendar/v1/",
  "sub": "usr_123",
  "scope": "read write",
  "iat": 1458785796,
  "exp": 1458872196
}
```

ID token

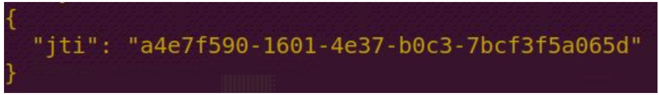
- Defined within OIDC
- Is a JWT intended to be read by the OAuth Client, which is the *audience* of the token
- May also contain information about the user such as their name or email address
  - client applications can use it to build a user profile to personalize the user experience



```
{
  "iss": "https://server.example.com",
  "sub": "24400320",
  "aud": "s6BhdRkqt3",
  "nonce": "n-0S6_WzA2Mj",
  "exp": 1311281970,
  "iat": 1311280970,
  "auth_time": 1311280969
}
```

Refresh token (RT)

- Defined within OAuth 2
- Is a string that the OAuth Client can use to get a new AT without the user's interaction
- Must not allow the Client to gain any access beyond the scope of the original grant



```
{
  "jti": "a4e7f590-1601-4e37-b0c3-7bcf3f5a065d"
}
```

OAuth/OIDC grant types

Authorization grant types

=

Authorization Flows

=

Ways for an application to get tokens

IAM supported OAuth grant types

Authorization grant types, or authorization flows, are ways for an application to get tokens

- **authorization code** → mainly used by server-side web applications which can maintain the confidentiality of client credentials
- **device code** → used by clients that can not easily trigger a browser-based authorization and could run on a separate device
- **refresh token** → it allows an application to act on behalf of a user and get tokens without user's interaction
- **client credentials** → used to obtain tokens not linked to user identities, since the client can make token requests by itself
- **token exchange** → satisfies the needs to access resources hosted by other downstream services on behalf of the user
- **implicit** (deprecated in OAuth 2.1) → it simplifies the authorization code flow, mainly used by client-side web applications
- **password** (deprecated in OAuth 2.1) → linked to user's credentials, does not support delegation

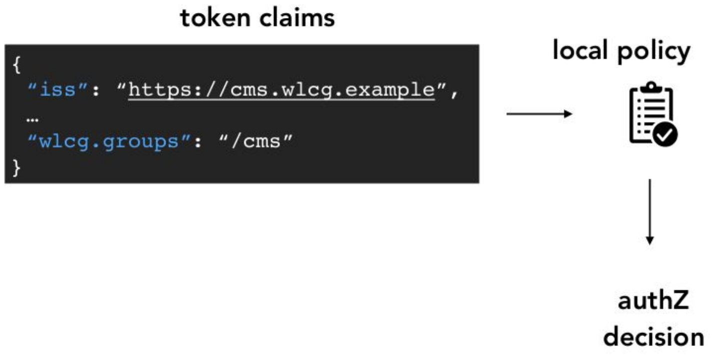
Identity-based vs Scope-based Authorization

Identity-based authorization

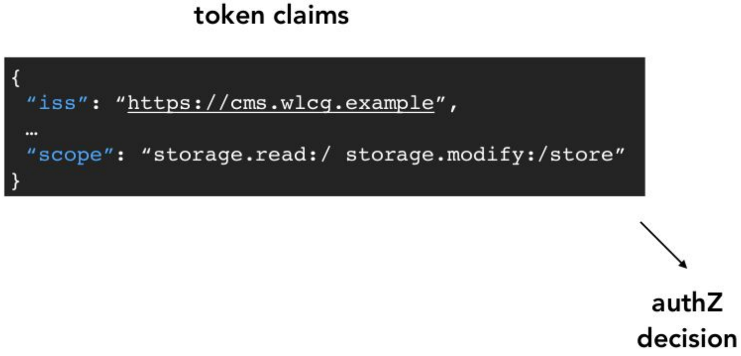
- the token brings information about attribute entitlement (*e.g.*, group/role membership)
- the service maps these attributes to a local authorization policy

Scope-based authorization

- the token brings information about which actions should be authorized at a service
- the service needs to understand these capabilities and honor them
- the authorization policy is managed at the VO level (*i.e.*, IAM)



```
{
  “iss”: “https://cms.wlcg.example”,
  …
  “wlcg.groups”: “/cms”
}
```



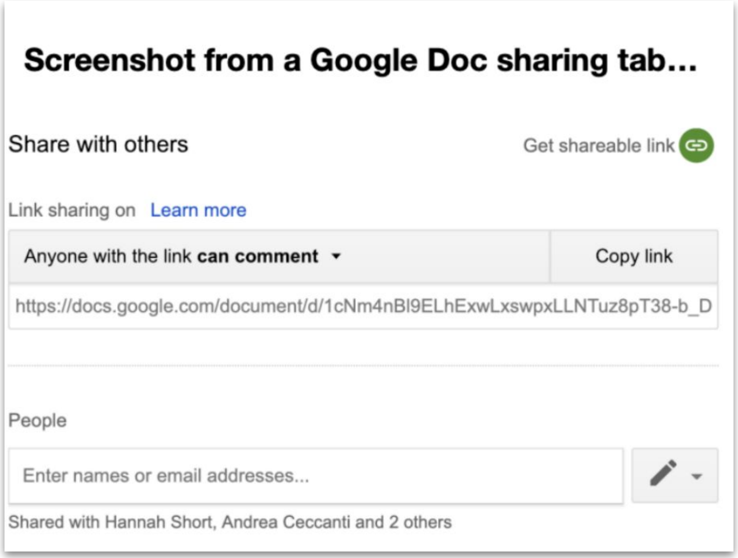
```
{
  “iss”: “https://cms.wlcg.example”,
  …
  “scope”: “storage.read:/ storage.modify:/store”
}
```

Identity-based vs Scope-based Authorization

The two models can coexist, even in the context of the same application!

Scope-based authZ →

Identity-based authZ →



Screenshot from a Google Doc sharing tab...

Share with others

Link sharing on Learn more

Anyone with the link **can comment**

https://docs.google.com/document/d/1cNm4nBl9ELhExwLxswpxLLNTuz8pT38-b_D

People

Enter names or email addresses...

Shared with Hannah Short, Andrea Ceccanti and 2 others