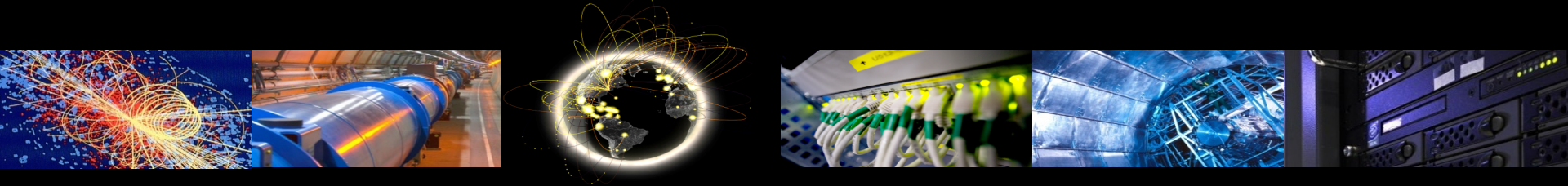


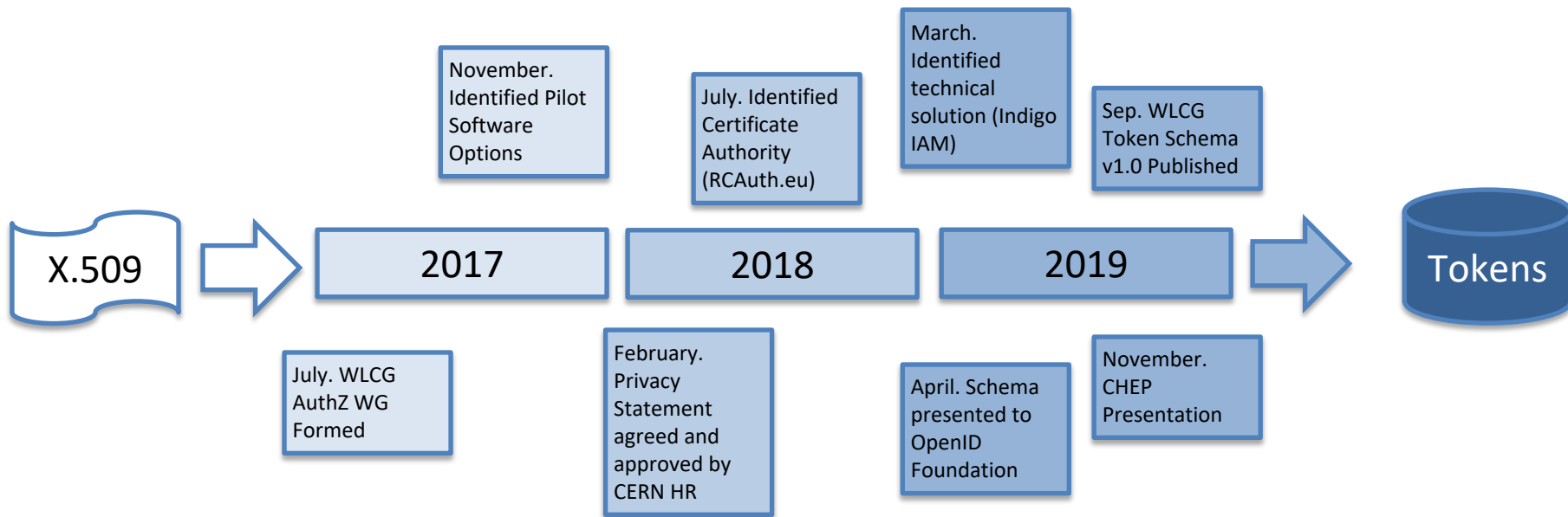
WLCG Authorisation; from X.509 to Tokens

Authored by the WLCG AuthZ Working Group

CHEP, Adelaide, November 5th 2019



Towards Tokens



Why? Motivation

- **Evolving Identity Landscape**
 - User-owned X.509 certificates -> federated identities (SAML & OpenID Connect)
- **Technology Readiness**
 - Increasing solutions for shielding users from the complexities of X.509 certificate management
 - Token-based authorisation widely adopted in commercial services and increasingly by R&E Infrastructures
- **Data Protection**
 - Tightening of data protection (GDPR) requires fine-grained user level access control, certain provisioning practices may need to be adjusted

However, current grid middleware does not support token (OAuth2) based authorisation.

Objective: Understand & meet the requirements of a future-looking AuthZ service for WLCG experiments

Who? WLCG AuthZ WG

Representation from wide range of institutes and experiments. Development work of pilot projects supported by:



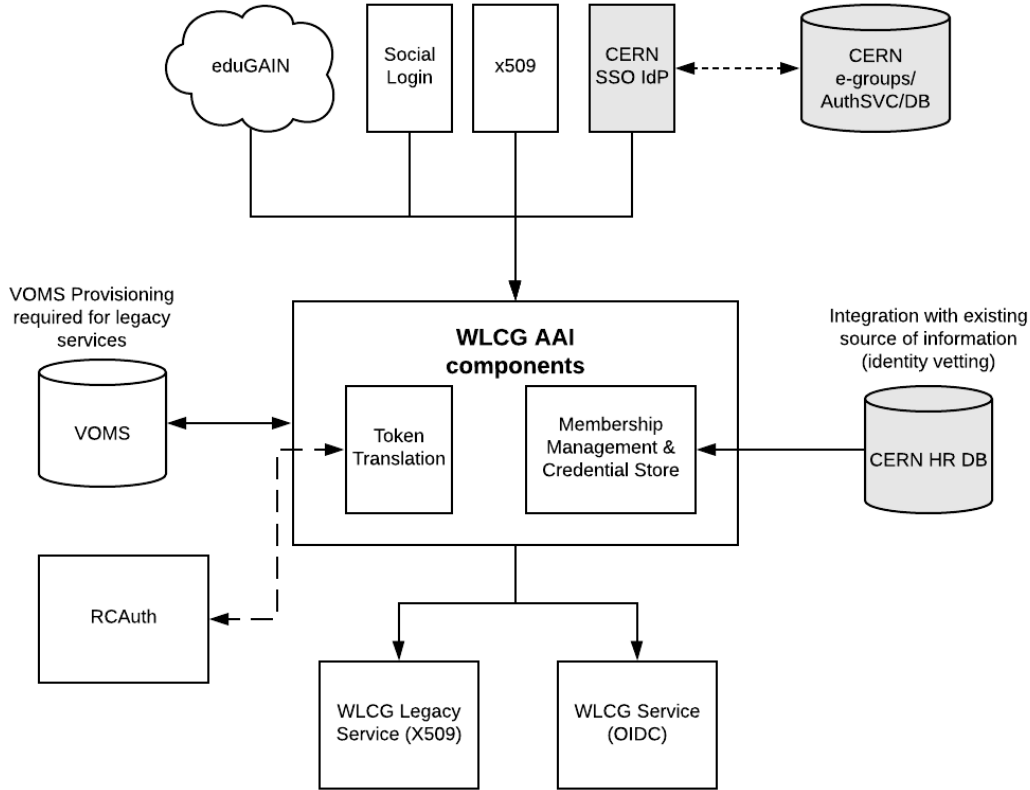
Work tracked in the [WLCG AuthZ WG wiki page](#)

What are we improving?

- Usability
 - Removing need for users to manage user-certificates
 - Ability to authenticate with home organisation credentials
- Membership Management
 - More flexibility on user authentication
- Simplified integration
 - Adopting widely accepted technologies (OAuth2 and OIDC)
 - Priority to stick to standards

What? Solution Design

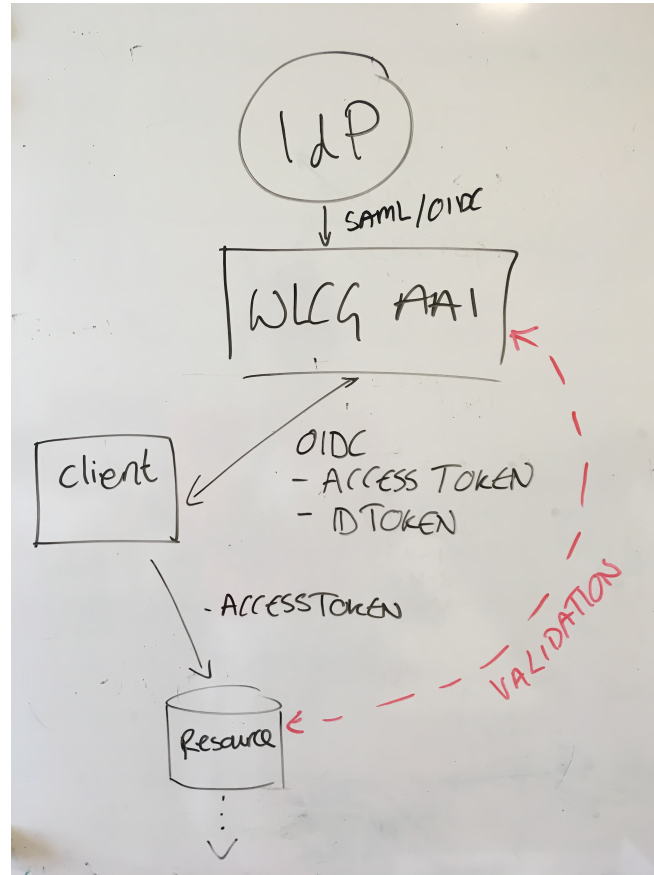
CERN components are optional configuration – technical solution is widely relevant!



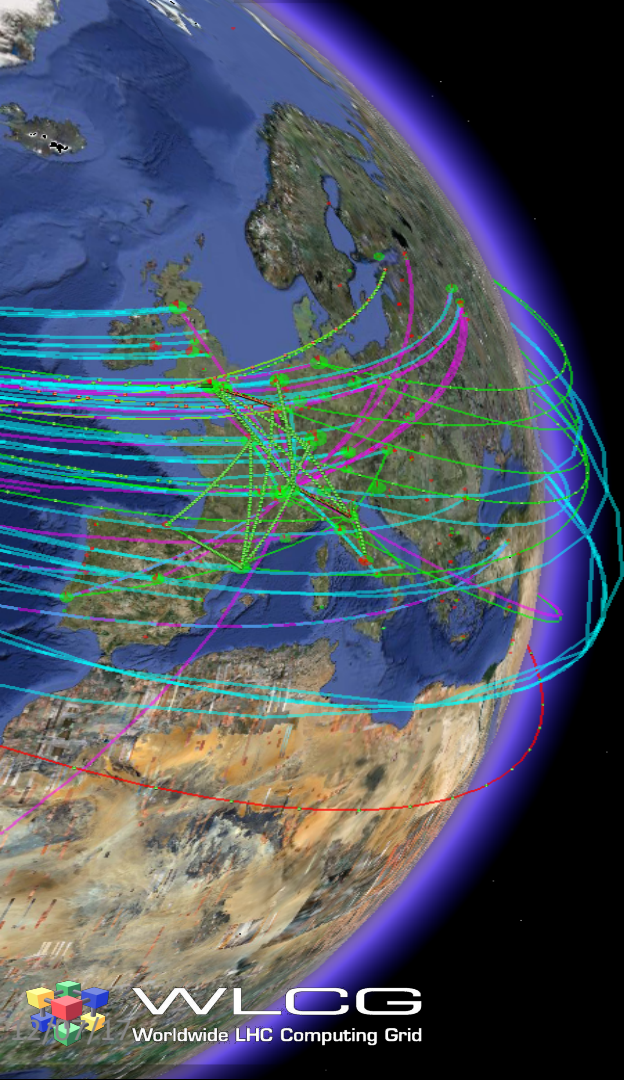
More details in Thursday's talk

WLCG AuthZ WG

Draft usage flow



Token Schema



V1.0

- Published on Zenodo, September 25th 2019
- Allows middleware developers to enable token based authorization to an agreed schema
- Tests to be run within WLCG DOMA WG
- Compliant Token issuer available for WLCG integration testing:

<https://wlcg.cloud.cnaf.infn.it>

September 25, 2019

Technical note Open Access

WLCG Common JWT Profiles

Altunay, Mine; Bockelman, Brian; Ceccanti, Andrea; Cornwall, Linda; Crawford, Matt; Crooks, David; Dack, Thomas; Dykstra, David; Groep, David; Igoumenos, Ioannis; Jouvin, Michel; Keeble, Oliver; Kelsy, David; Lassnig, Mario; Liampotis, Nicolas; Litmaath, Maarten; McNab, Andrew; Millar, Paul; Sallé, Mischa; Short, Hannah; Teheran, Jeny; Wartel, Romain

This document describes how WLCG users may use the available geographically distributed resources without X.509 credentials. In this model, clients are issued with bearer tokens; these tokens are subsequently used to interact with resources. The tokens may contain authorization groups and/or capabilities, according to the preference of the Virtual Organisation (VO), applications and relying parties.

Wherever possible, this document builds on existing standards when describing profiles to support current and anticipated WLCG usage. In particular, three major technologies are identified as providing the basis for this system: OAuth2 (RFC 6749 & RFC 6750), OpenID Connect and JSON Web Tokens (RFC 7519). Additionally, trust roots are established via OpenID Discovery or OAuth2 Authorization Server Metadata (RFC 8414). This document provides a profile for OAuth2 Access Tokens and OIDC ID Tokens.

Preview

Page: 1 of 35 Automatic Zoom?

WLCG Common JWT Profiles

Authored by the WLCG AuthZ Working Group

Version History:

Date	Version	Comment
------	---------	---------

<https://zenodo.org/record/3460258#.XacTNI2Q011>

Edit

New version

98

views

81

downloads

[See more details...](#)

Indexed in

OpenAIRE

Publication date:
September 25, 2019

DOI:
DOI: 10.5281/zenodo.3460258

Keyword(s):
jwt, oidc, OAuth2.0, wlcg

License (for files):
[Creative Commons Attribution 4.0 International](#)

V1.0

- Published on Zenodo, September 25th 2019
- Allows middleware developers to enable token based authorization to an agreed schema
- Tests to be run within WLCG DOMA WG
- Compliant Token issuer available for WLCG integration testing:

<https://wlcg.cloud.cnaf.infn.it>

The screenshot shows a web browser window with the URL `https://wlcg.cloud.cnaf.infn.it/iam-test-client/`. The page title is "INDIGO IAM Test Client Application". The user is logged in as "Andrea Ceccanti". The authorization request included the scopes: "openid profile wlcg.groups". The application has received the following information:

- access_token (JWT):
`eyJraWQiOiJyc2ExIiwiaWF0IjoiUjYyOTYifQ.eyJ3bGNnLnZlciI6IjEuMCIsInN1YiI6ImExYjYk4MzMLTK`
- access_token (decoded):

```
{
  "wlcg.ver": "1.0",
  "sub": "a1b98335-9649-4fb0-961d-5a49ce108d49",
  "scope": "openid wlcg.groups profile",
  "iss": "https://wlcg.cloud.cnaf.infn.it/",
  "exp": 1572829780,
  "iat": 1572826180,
  "jti": "eb7b51b9-c24b-41b6-86b6-28048b9fe334",
  "wlcg.groups": [
    "/wlcg"
  ]
}
```
- OAuth2 token introspection endpoint response (invoked on access_token, authorized by client credentials):

```
{
  "active": true,
  "scope": "openid profile wlcg.groups",
  "expires_at": "2019-11-04T02:09:41+0100",
  "exp": 1572829781,
  "sub": "a1b98335-9649-4fb0-961d-5a49ce108d49",
  "user_id": "andrea",
}
```

Token Claims

Common Claims

- sub
- exp
- iss
- acr
- aud
- iat
- nbf
- jti
- eduperson_assurance (REFEDS)
- wlcg.ver (WLCG)
- wlcg.groups (WLCG)

ID Token Claims

- auth_time
- general OIDC Claims

Access Token Claims

- scope (inspired by OAuth token exchange draft)

Note: Where unspecified, the origin is RFC7519 or OpenID Connect core

Two forms of Authorization

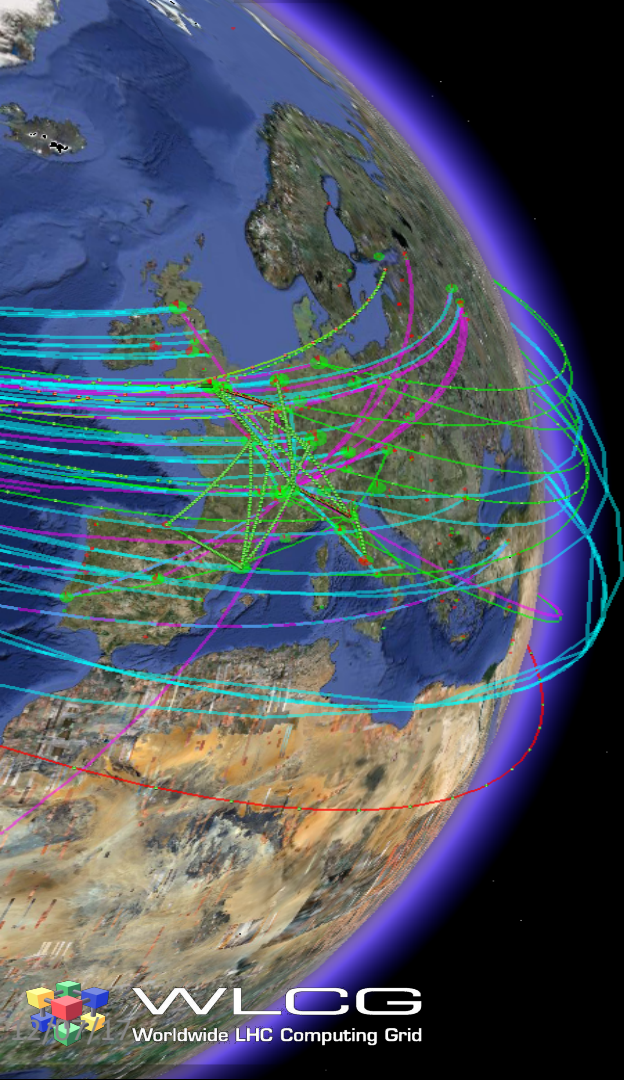
Tokens Assert **Group Membership**

- Similar to VOMS Groups
- VOMS Roles modeled as optional Groups

Token Asserts **Authorized Actions**

- Called “Capabilities/scopes”
- Specific ability to perform an action (optionally, at a specific path) e.g.
storage.create:/home/joe

*A single schema to rule them all:
Scitokens library will soon have
support for WLCG JWT profile*



Rucio AuthN/Z with Tokens

Rucio AuthN/Z with OAuth/OIDC and JWTs



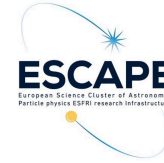
Jaroslav Guenther

For more details see this [talk](#)

- **Rucio user**
 - “account” (CERN LDAP username) + “identity” (auth_type + ID)
 - only carefully ‘pre-provisioned’ users
 - Rucio daemon syncs accounts & identities (using IAM* SCIM client)
- **Rucio development branch is currently supporting**
 - “Authorization Code Flow” for basic user authentication
 - “Token Exchange” to propagate permissions to downstream services (e.g. FTS)
 - “Refresh Token” to act on behalf of a user (for configurable time)
 - Supporting multiple Identity Providers (Escape IAM, XDC IAM, ...)
 - Rucio WebUI login
 - 3 CLI login strategies (automatic, browser redirect & browser redirect + auth server polling)

Practical Tests

- XDC IAM response time
 - Rucio user authentication currently @ 2Hz
 - XDC IAM test cluster average response time ~ 0.5 s
- Initiate data transfer:
 - authenticated Rucio user requests Rucio to make a transfer
 - Rucio exchanges the token for a new one with 'fts' in audience claim
 - FTS receives JWT in the header and initiates the transfer on dCache storage



✓ authorising via XDC IAM JWT
in FTS request header



✓ works only with:

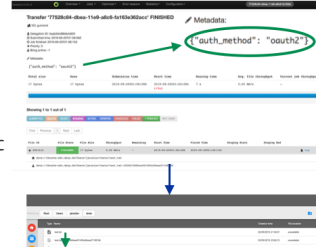
- https/davs on port 443

✓ only storages that support OIDC

- no token translation



```
FTS_TransferTool = rucio.transferrtool-fts3-FTS3TransferTool(C=https://fts3-rid.cern.ch:8442/?transfer_id=FTS3_TransferTool,submit(files,_job_params,_timeout=3600)
```



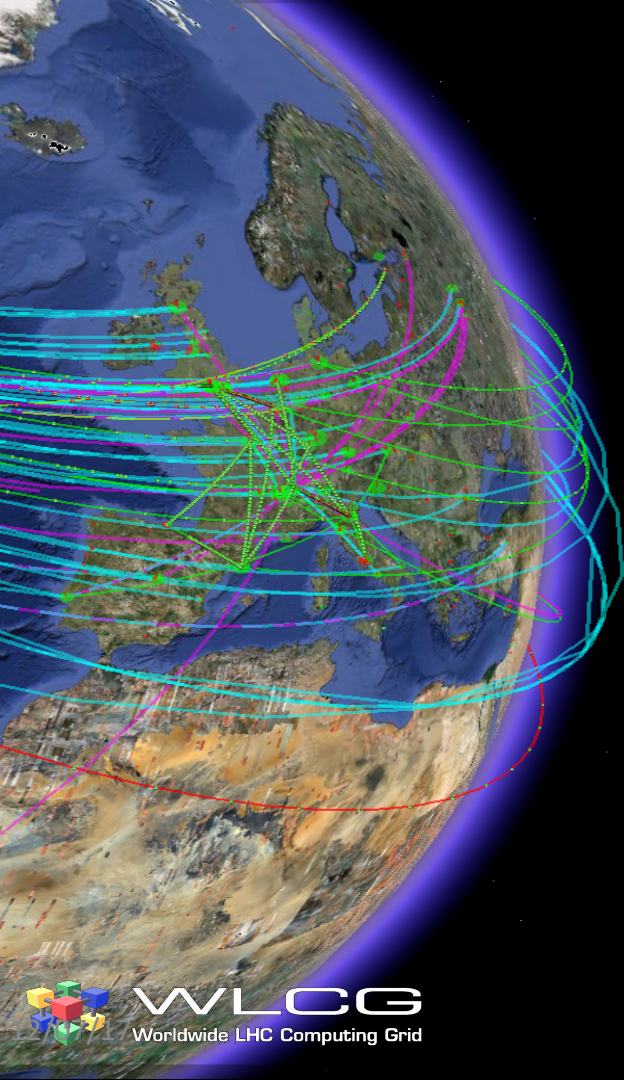
This works thanks to
DOMA-TPC work with
HTTP-TPC – also being
presented at CHEP*!

- Deploying testbed on Rucio Escape Instance:
 - development release is being deployed on Rucio Escape Instance

Jaroslav Guenther

Conclusions

- Common JWT Profile has reached v1.0
 - **stable reference** for developers and integrators
- WLCG IAM instance **available today** to start the integration work



Questions?