DOMA TPC Token-based AuthZ

Andrea Ceccanti
INFN CNAF

December, 18th 2019

Token-based AuthN/Z “Hackathon” @ CERN in January

What: sort out as many problems as possible while discussing things and coding together in a room with the objective of demonstrating **a full stack HTTP X509-free data transfer management chain**

- RUCIO->FTS->SEs
- SEs: EOS, dCache, DPM, StoRM, XRootD, Echo

Who: Developers of the above components

When/Where: **January, 16th 2020 @ CERN**

Please register:

https://indico.cern.ch/event/870616

What does it mean supporting the WLCG profile?

As an **OAuth resource server** (RS):

- Ability to extract an access token from an incoming HTTP request
- Ability to parse and validate the incoming access token
  - identify if it has been issue by a trusted and recognized authorization server
  - verify temporal validity
  - verify signature, following OAuth/OIDC conventions
- Ability to honour access token audience restrictions
  - the RS needs the ability to identity itself with (one or multiple) audience labels and honour audience restrictions in access tokens
- Ability to map defined scopes to local authZ
  - e.g., storage.read:/cms grants read access to the /cms namespace (and any subdirectory)
- Ability to map group-based to local authZ
  - e.g., /cms group membership as stated grants read access to the /cms namespace

What does it mean supporting the WLCG profile?

As an **OAuth resource server** (RS):

- Ability to extract an access token from an incoming HTTP request
- Ability to parse and validate the incoming access token
  - identify if it has been issue by a trusted and recognized authorization server
  - verify temporal validity
  - verify signature, following OAuth/OIDC conventions
- Ability to honour access token audience restrictions
  - the RS needs the ability to identity itself with (one or multiple) audience labels and honour audience restrictions in access tokens
- Ability to map defined scopes to local authZ
  - e.g., storage.read:/cms grants read access to the /cms namespace (and any subdirectory)
- Ability to map group-based to local authZ
  - e.g., /cms group membership as stated grants read access to the /cms namespace

This is typically sorted out by OAuth/OIDC libraries

Hackathon objectives: the testbed

We’ll use the wlcg IAM instance

We’ll need an endpoint configured to support the wlcg IAM VO for all the SEs

That endpoint should also be easily updatable during the hackathon, to get the latest code changes deployed and test integrations

Hackathon objectives: ideas

Ideally: demonstrate **a full stack HTTP X509-free data transfer management chain**

- using scope-based authz
- using group-based authz

showcasing that RUCIO->FTS->SEs chain works fine for all SEs using tokens issued by the wlcg IAM instance

Realistically:

- How do we structure this?
- We can have an introduction on the flows and the JWT profile as first thing
- The we can go round-table to understand the current level of support for token-based authn/z in each component and choose which problems to attack first

Thanks for your attention.
Questions?