Rucio token workflow evolution

Introduction

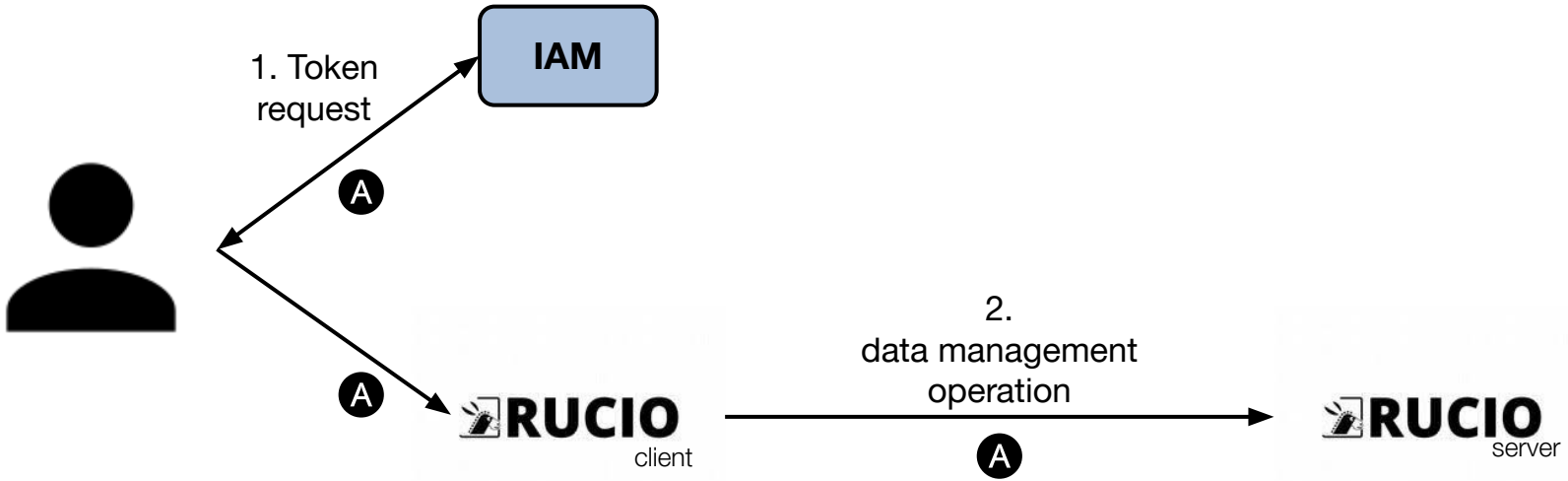
Martin Barisits, Dimitrios Christidis
on behalf of the Rucio team

RUCIO
SCIENTIFIC DATA MANAGEMENT

A bit of background

- OIDC/OAuth2 Tokens added to Rucio in 2019 via XDC project funding
- Functional, and continuously tested in DOMA testbed
- However, functionality is quite coarse
  - Largely replaces usage of X509 proxies with “fat-tokens”
- Ongoing discussion in WLCG AuthZ workgroup and DOMA TPC/DBT
  - Thanks a lot to all participants!
- Lots of interest from non-HEP Rucio communities
  - Especially Astronomy sector, since fine-grained tokens could solve their data-embargo issues
  - X509 is a big barrier-of-entry for new (non-HEP) communities

Authentication to Rucio 1/2

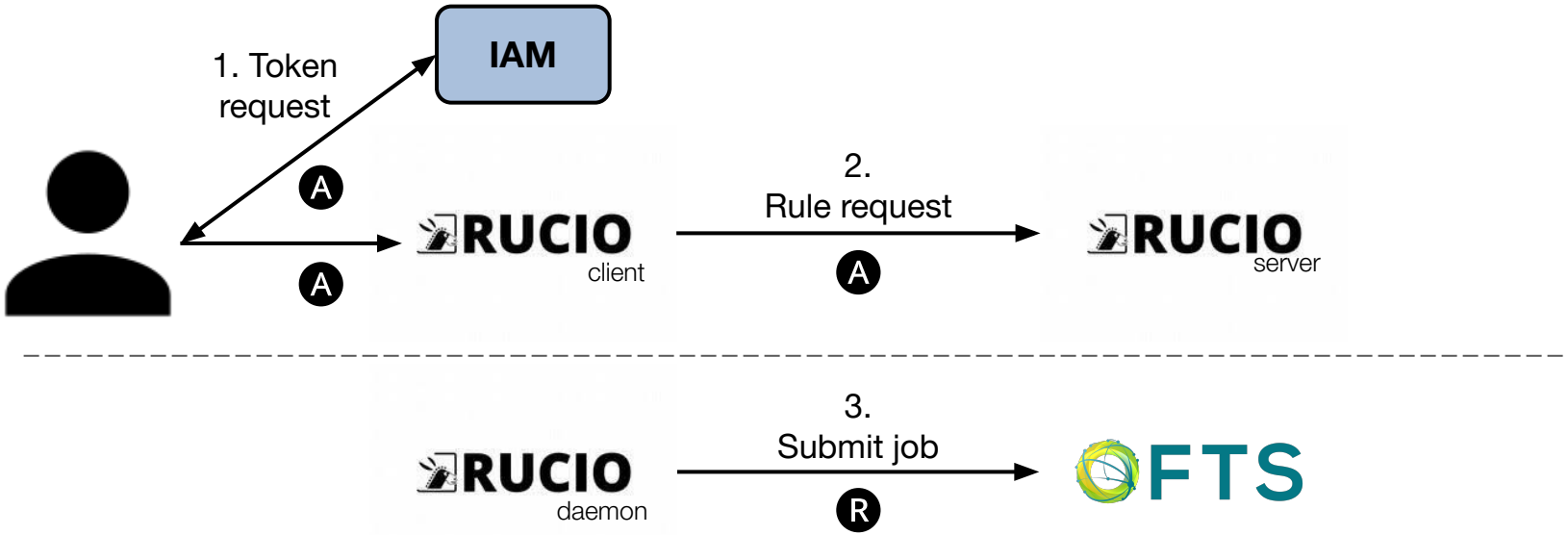


Authentication to Rucio 2/2

- The access token Ⓐ presented to Rucio needs to have an audience (**aud**) claim corresponding to the Rucio instance, otherwise authentication will be rejected

```
{
        "sub": "e1eb758b-b73c-4761-bfff-adc793da409c",
        "iss": "https://atlas-iam.cern.ch",
        "aud": "https://atlas-rucio.cern.ch",          ← Rucio audience
        "exp": 1555060391,
        "iat": 1555059791,
        "jti": "aef94c8c-0fea-490f-9027-ff444dd66d8c"
}
```

User initiated transfer (rule creation) 1/2



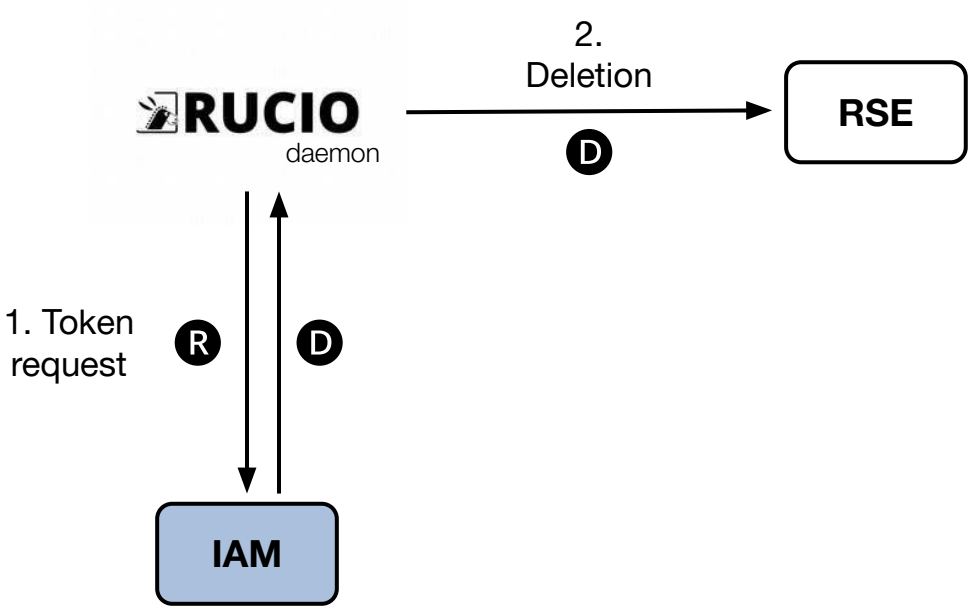
User initiated transfer (rule creation) 2/2

- Rucio only uses its own identity to submit to FTS
- Example

```
{
        "sub": "atlas-rucio",                                   ← RUCIO SERVICE IDENTITY
        "iss": "https://atlas-iam.cern.ch",
        "aud": "fts-atlas.cern.ch",                             ← FTS audience
        "exp": 1555060391,
        "iat": 1555059791,
        "jti": "aef94c8c-0fea-490f-9027-ff444dd66d8c"
}
```

- FTS will subsequently issue token exchanges for storage scoped tokens
  - See WLCG Token Usage and Discovery article

Rucio issued deletion requests to storage 1/2



- Rucio requests a new token for its own identity which is properly scoped and audience restricted
- These tokens are cached and re-used within the deletion daemon

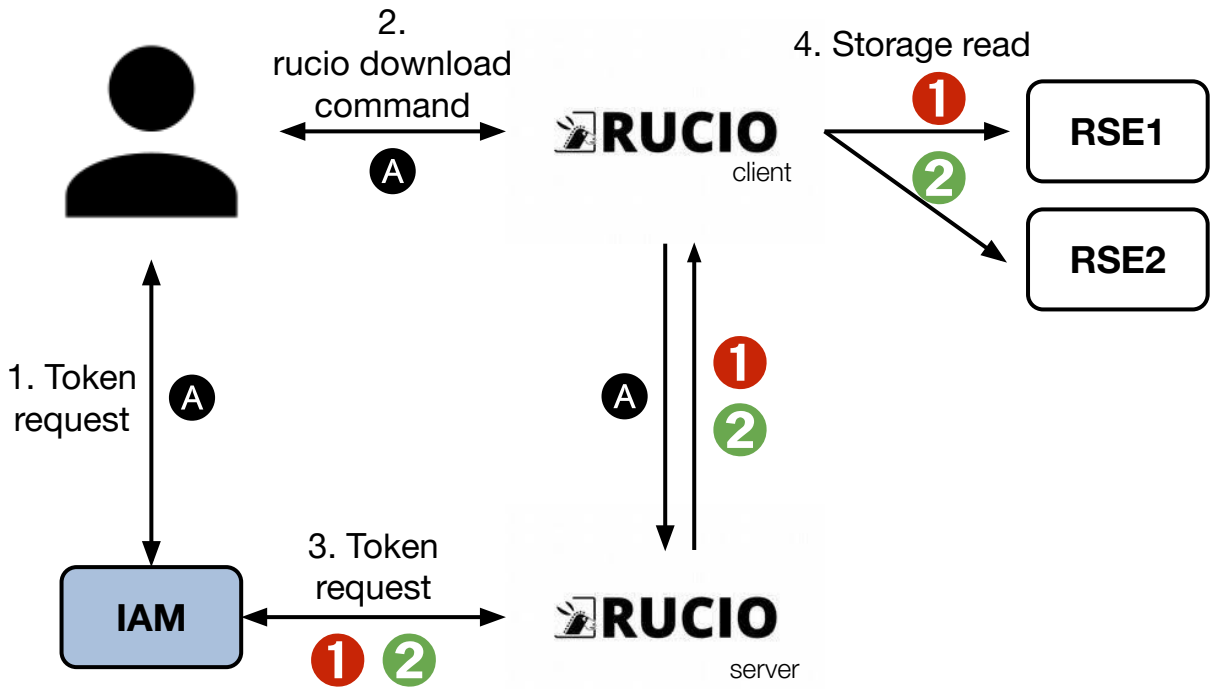
Rucio issued deletion requests to storage 2/2

- The 🅓 access token used to issue the storage deletion requests should be
  - Audience restricted to the specific storage system
  - Scope restricted to be able to delete
    - Rucio plugin approach ⟶ It is up to the VO how restrictive they want their tokens to be
      - For WLCG the token will likely be on the “root” of the RSE (e.g. `/atlas/atlasdatadisk/rucio/`)
- Example

```
{
        "sub":   "atlas-rucio",                                          ← RUCIO SERVICE IDENTITY
        "iss":   "https://atlas-iam.cern.ch",
        "aud":   "eos-atlas.cern.ch",                                    ← Storage audience
        "exp":   1555060391,
        "iat":   1555059791,
        "scope": "storage.modify:/atlas/atlasdatadisk/rucio/"            ← Storage scope*; WLCG common token schema
        "jti":   "aef94c8c-0fea-490f-9027-ff444dd66d8c"
}
```

*: See slide 12 for scope restrictions

User downloading data from storage 1/2

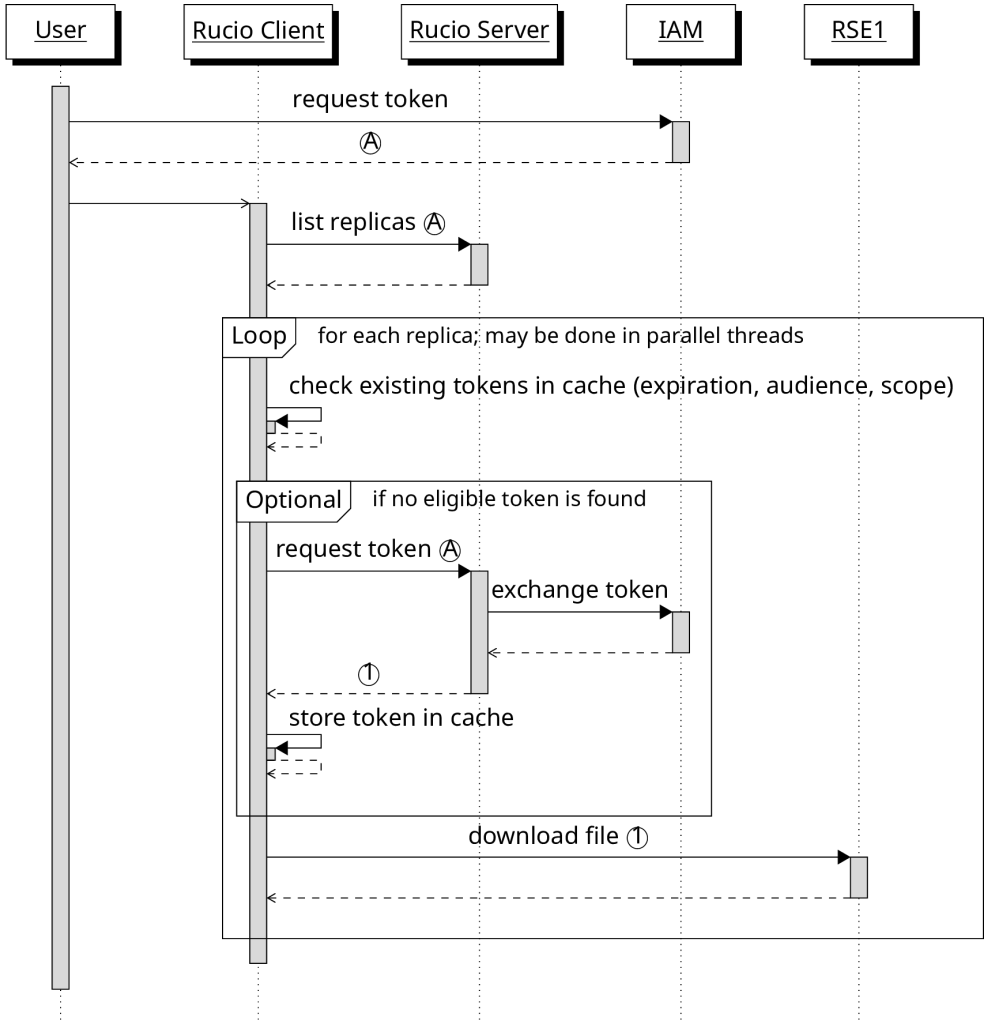


User downloading data from storage 2/2

- Example for ❶ and ❷

```
{
        "sub":   "atlas-rucio",                                          ← RUCIO SERVICE IDENTITY
        "iss":   "https://atlas-iam.cern.ch",
        "aud":   "eos-atlas.cern.ch",                                    ← Storage audience
        "exp":   1555060391,
        "iat":   1555059791,
        "scope": "storage.read:/atlas/atlasdatadisk/rucio/"              ← Storage scope*; WLCG common token schema
        "jti":   "aef94c8c-0fea-490f-9027-ff444dd66d8c"
        "act":   {
                    "sub":    "e1eb758b-b73c-4761-bfff-adc793da409c"     ← Optionally user identity for traceability
        }
}
```

*: See slide 12 for scope restrictions


User
Rucio Client
Rucio Server
IAM
RSE1
request token
Ⓐ
list replicas Ⓐ
Loop
for each replica; may be done in parallel threads
check existing tokens in cache (expiration, audience, scope)
Optional
if no eligible token is found
request token Ⓐ
exchange token
①
store token in cache
download file ①

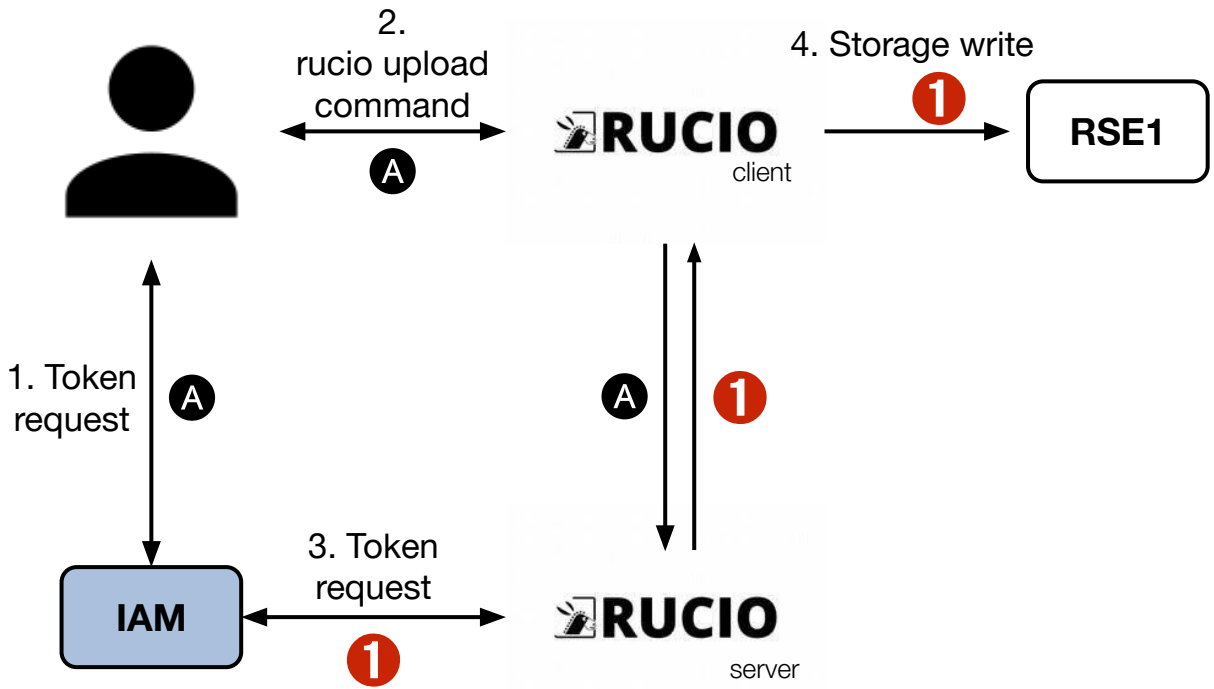
Limiting scope

- How limited should the scope be?
- Decision each VO needs to make; Rucio will offer a plugin mechanism
- Rucio will natively provide RSE / Scope / File
  - Additional plugins can be implemented by VOs

- 

- More restriction → more token exchanges → more load on IAM

User uploading data to storage 1/3



User uploading data to storage 2/3

- Workflow very similar to download
- Scope limitations for WLCG would need to be discussed
  - In general VO-configurable similar to download
- Certain upload workflows upload a temporary file and rename after success
  - Requires **storage.create** profile to allow renaming

User uploading data to storage 3/3

- Example for ❶

```
{
        "sub":   "atlas-rucio",                                                          ← RUCIO SERVICE IDENTITY
        "iss":   "https://atlas-iam.cern.ch",
        "aud":   "eos-atlas.cern.ch",                                                    ← Storage audience
        "exp":   1555060391,
        "iat":   1555059791,
        "scope": "storage.create:/atlas/atlasdatadisk/rucio/scope01/aa/bb/file.root"     ← Storage scope*; WLCG common token schema
        "jti":   "aef94c8c-0fea-490f-9027-ff444dd66d8c"
        "act":   {
                    "sub":    "e1eb758b-b73c-4761-bfff-adc793da409c"                     ← Optionally user identity for traceability   }
}
```

*: See slide 12 for scope restrictions

Implications

- Rucio becomes the central token issuing component of the infrastructure!
- Privileged workflows need to be foreseen
  - Pilots running on worker nodes might have their own tokens with the possibility to write to local storage
- IAM high-availability still crucial
  - Longer token lifetime will only safe us from some issues
  - Most workflows require token-exchanges, which require availability of IAM

Misc.

- Dual stack / hybrid functionality
  - Needs to be foreseen (WLCG, DUNE)
- Need to be able to talk to IAM + CILogon (Plugin approach)
  - Requires token exchange support (RFC 8693)
- Support possibility to add additional token profile plugins

Plans

Multiple steps inline with WLCG timeline

1. Review client authentication workflows [2023]
2. Evolve deletion workflows [2023]
3. Evolve transfer workflows [2023/4]
4. Implement upload and download workflows [~2024]

Functionality required for DC24 basically in place in Rucio

More information

Website	http://rucio.cern.ch
Documentation	https://rucio.cern.ch/documentation
Repository	https://github.com/rucio/
Images	https://hub.docker.com/r/rucio/
Online support	https://rucio.slack.com/messages/#support/
Developer contact	rucio-dev@cern.ch
Publications	https://rucio.cern.ch/publications.html
Twitter	https://twitter.com/RucioData