SCI TOKENS

SciTokens: Capability-Based Secure Access to Remote Scientific Data

Jim Basney <jbasney@ncsa.Illinois.edu>

https://www.scitokens.org/

This material is based upon work supported by the National Science Foundation under Grant No. 1738962. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

SciTokens Project

- The SciTokens project:
  - Introduces a ***capabilities-based* authorization infrastructure** for distributed scientific computing,
  - Provides a **reference platform**, combining CILogon, HTCondor, CVMFS, and XRootD, and
  - **Implements specific use cases** to help our science stakeholders (LIGO and LSST) better achieve their scientific aims.

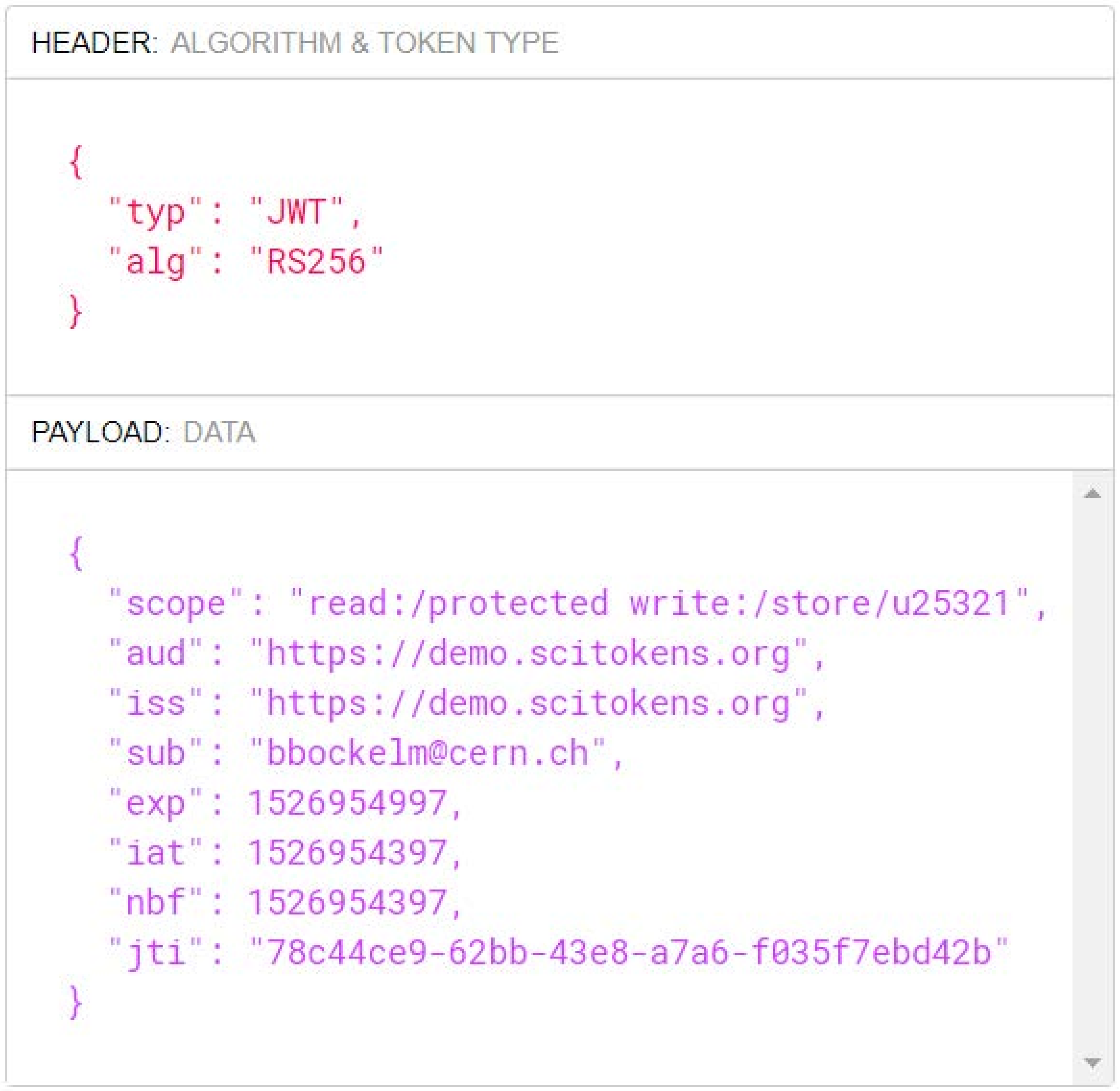
SciTokens uses standards

- RFC 6749: OAuth 2.0 Authorization Framework
  - token request, consent, refresh
- RFC 7519: JSON Web Token (JWT)
  - self-describing tokens, distributed validation
- RFC 8414: OAuth 2.0 Authorization Server Metadata
  - token signing keys, policies, endpoint URLs
- OAuth 2.0 Token Exchange (IETF OAuth WG I-D)
  - token delegation, drop privileges

Example Token, Decoded

SCI TOKENS

- The decoded token contains multiple scopes - basically filesystem authorizations.
- The audience narrows who the token is intended for.
- The issuer identifies who created the token; value used to locate the public keys needed to validate signature.
- The subject is an opaque identifier for the resource owner. In this case, it also happens to be the identity.
- The expiration is a Unix timestamp when the token expires. A typical lifetime is 10 minutes.



HEADER: ALGORITHM & TOKEN TYPE

```
{
  "typ": "JWT",
  "alg": "RS256"
}
```

PAYLOAD: DATA

```
{
  "scope": "read:/protected write:/store/u25321",
  "aud": "https://demo.scitokens.org",
  "iss": "https://demo.scitokens.org",
  "sub": "bbockelm@cern.ch",
  "exp": 1526954997,
  "iat": 1526954397,
  "nbf": 1526954397,
  "jti": "78c44ce9-62bb-43e8-a7a6-f035f7ebd42b"
}
```

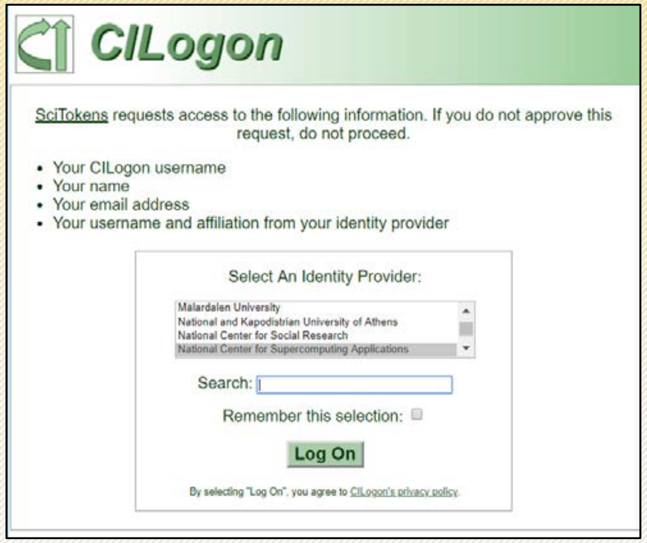
CILogon and SciTokens

<u>CILogon</u>

- Federated Identity Management
- OpenID Connect
- ID Tokens

<u>SciTokens</u>

- Federated Authorization
- OAuth 2.0
- Access Tokens



CILogon

SciTokens requests access to the following information. If you do not approve this request, do not proceed.

- Your CILogon username
- Your name
- Your email address
- Your username and affiliation from your identity provider

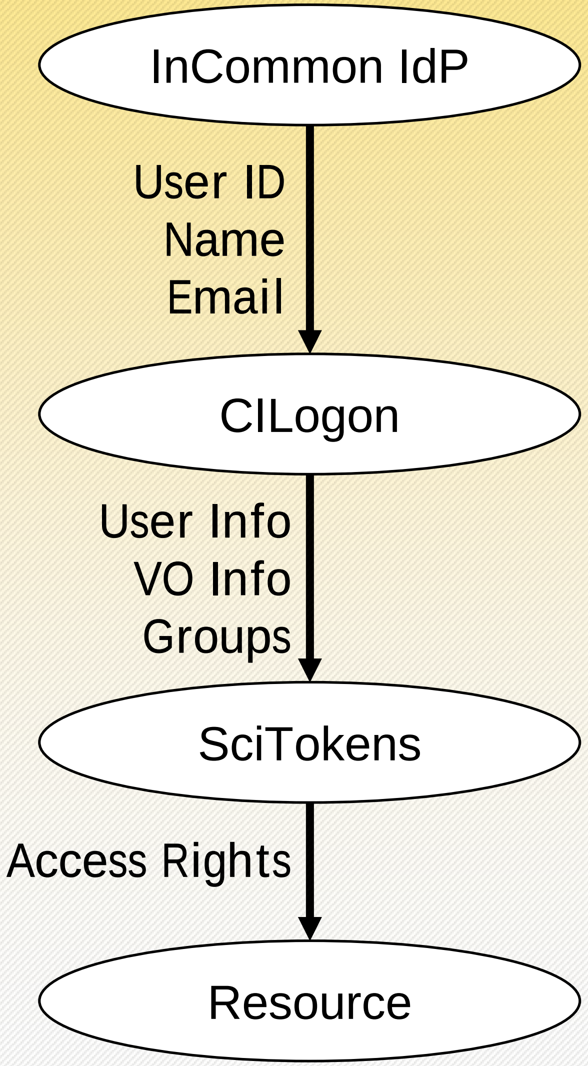
Select An Identity Provider:

Malardalen University
National and Kapodistrian University of Athens
National Center for Social Research
National Center for Supercomputing Applications

Search:

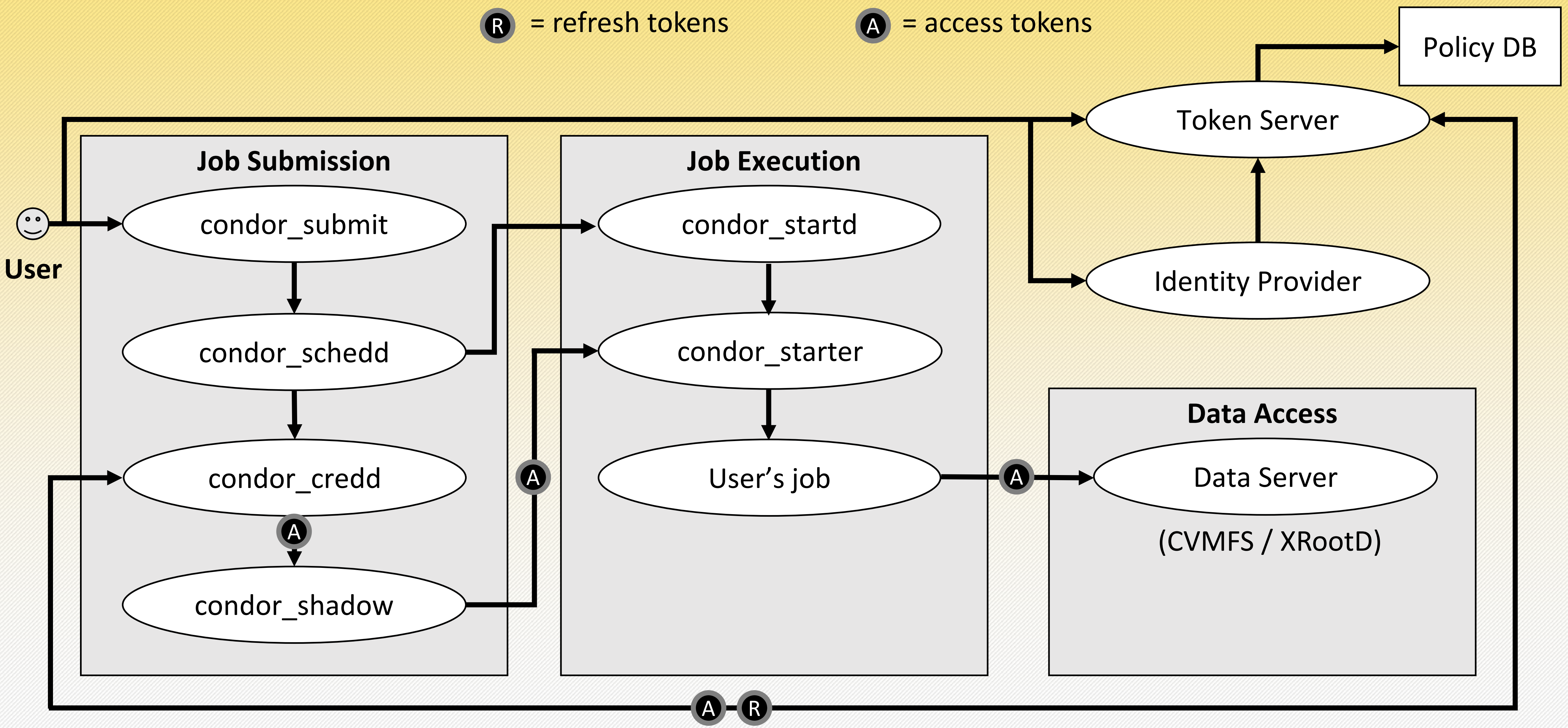
Remember this selection:

By selecting "Log On", you agree to CILogon's privacy policy.



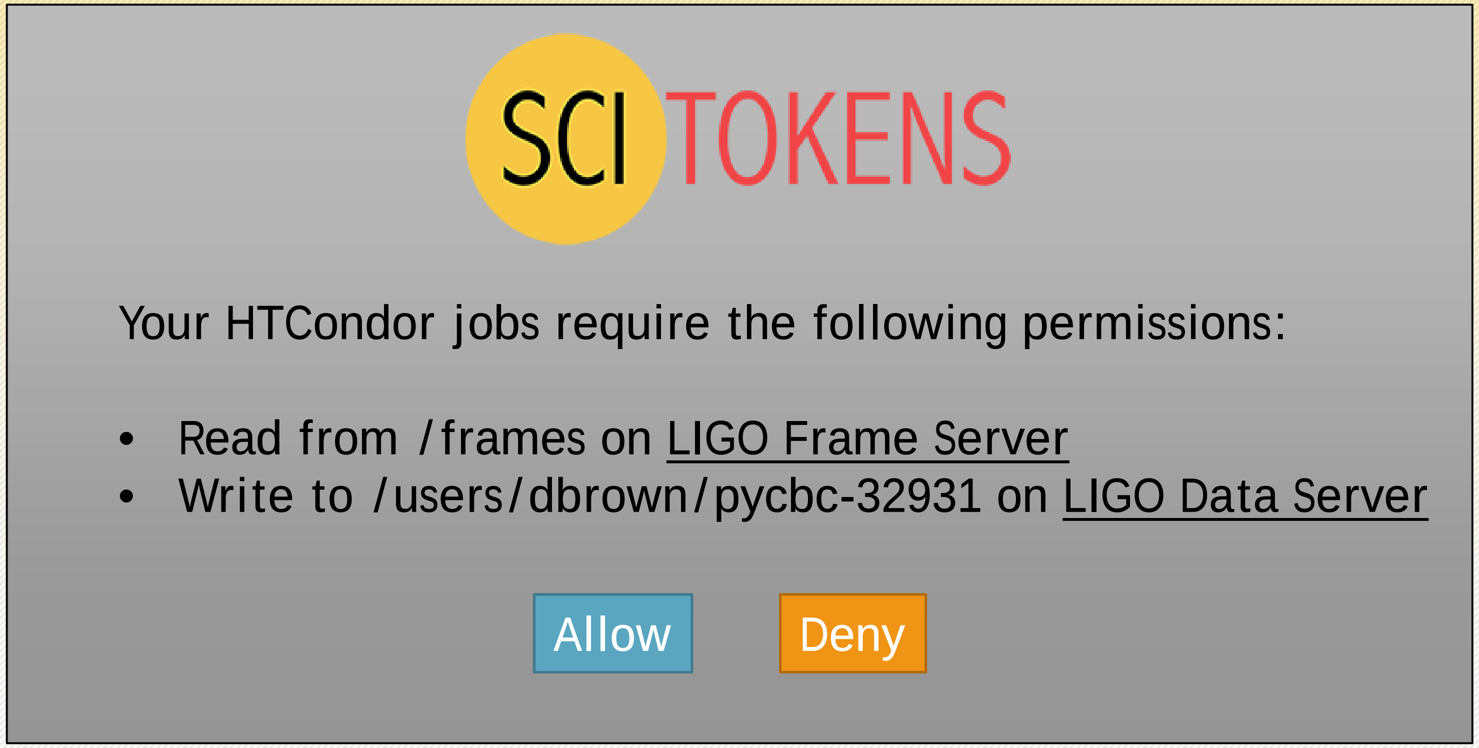
SciTokens System Architecture

SCI TOKENS



User Experience

```
user@chtc$ condor_submit workflow.jdl
Visit https://chtc.example.edu/authorize to authorize your jobs.
user@chtc$
```



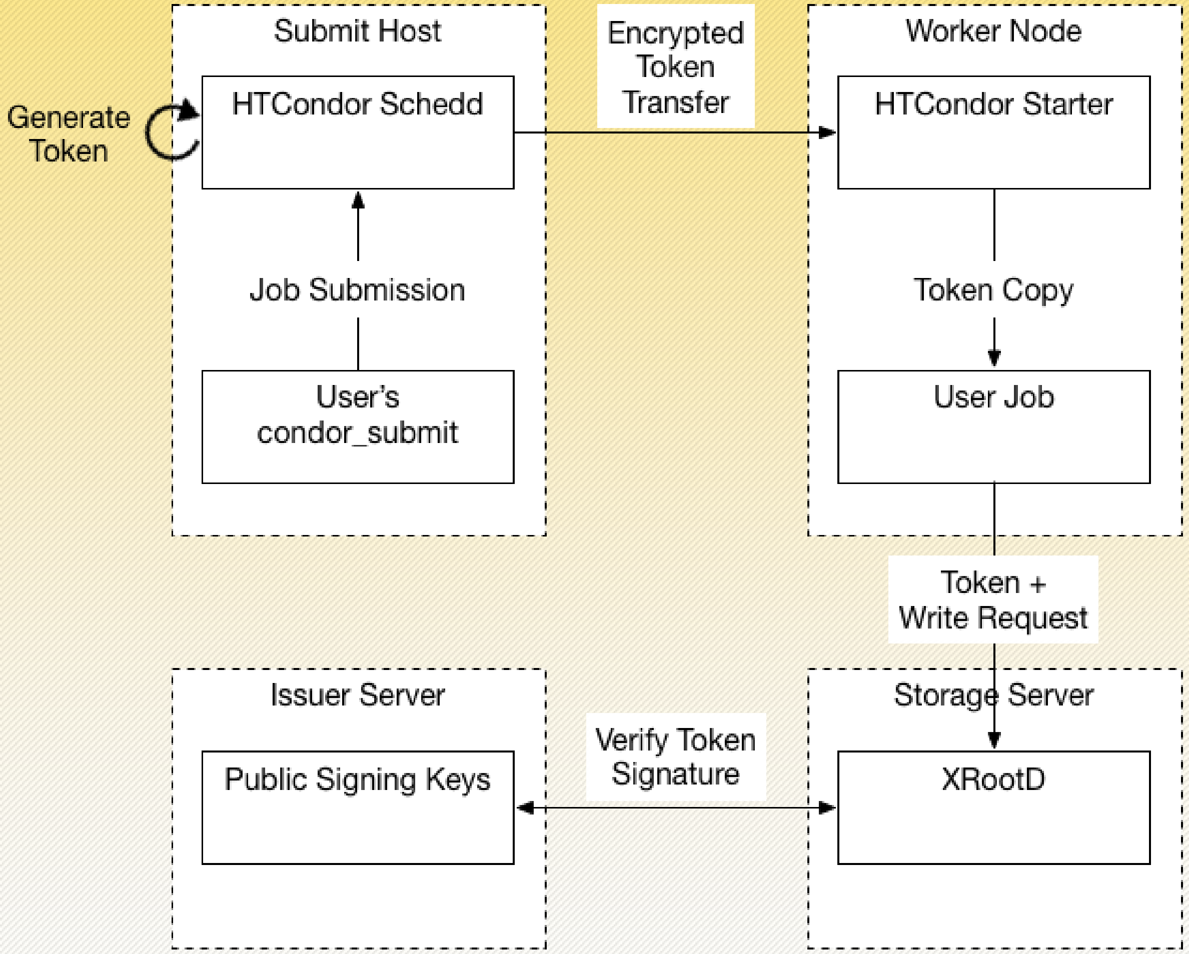
Your HTCondor jobs require the following permissions:

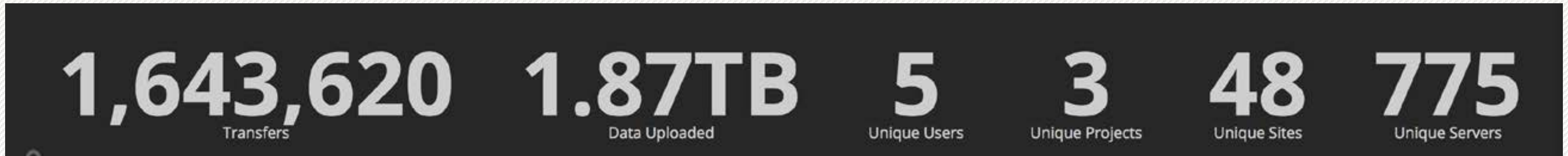
- Read from /frames on LIGO Frame Server
- Write to /users/dbrown/pycbc-32931 on LIGO Data Server

Early results on OSG

SCI TOKENS

- End-to-end token-based auth{z,n} workflow for the OSG VO submit service
- Includes patches to Xrootd to validate tokens presented via HTTPS and to write files out with the correct Unix user permissions
- **Details**:
  - instead of using OAuth2 to generate the token, we keep a signing key on the submit host.
  - only one token needed.
  - submit host and storage server owned by OSG.





1,643,620	1.87TB	5	3	48	775
Transfers	Data Uploaded	Unique Users	Unique Projects	Unique Sites	Unique Servers

Give SciTokens a try!

SCI TOKENS

- **https://demo.scitokens.org/** - token generator

- **https://github.com/scitokens/** - open source software
  - Java and Python implementations
  - SciTokens-aware token server
  - CVMFS, Nginx, and XRootD plugins
  - Docker image for XRootD setup

- **https://scitokens.org/** - docs, email lists