Introduction to SciTokens

European HTCondor Workshop 2018
RAL UK

Todd Tannenbaum tannenba@cs.wisc.edu
On behalf of the SciTokens Team

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: Federated Authorization Ecosystem for Distributed Scientific Computing

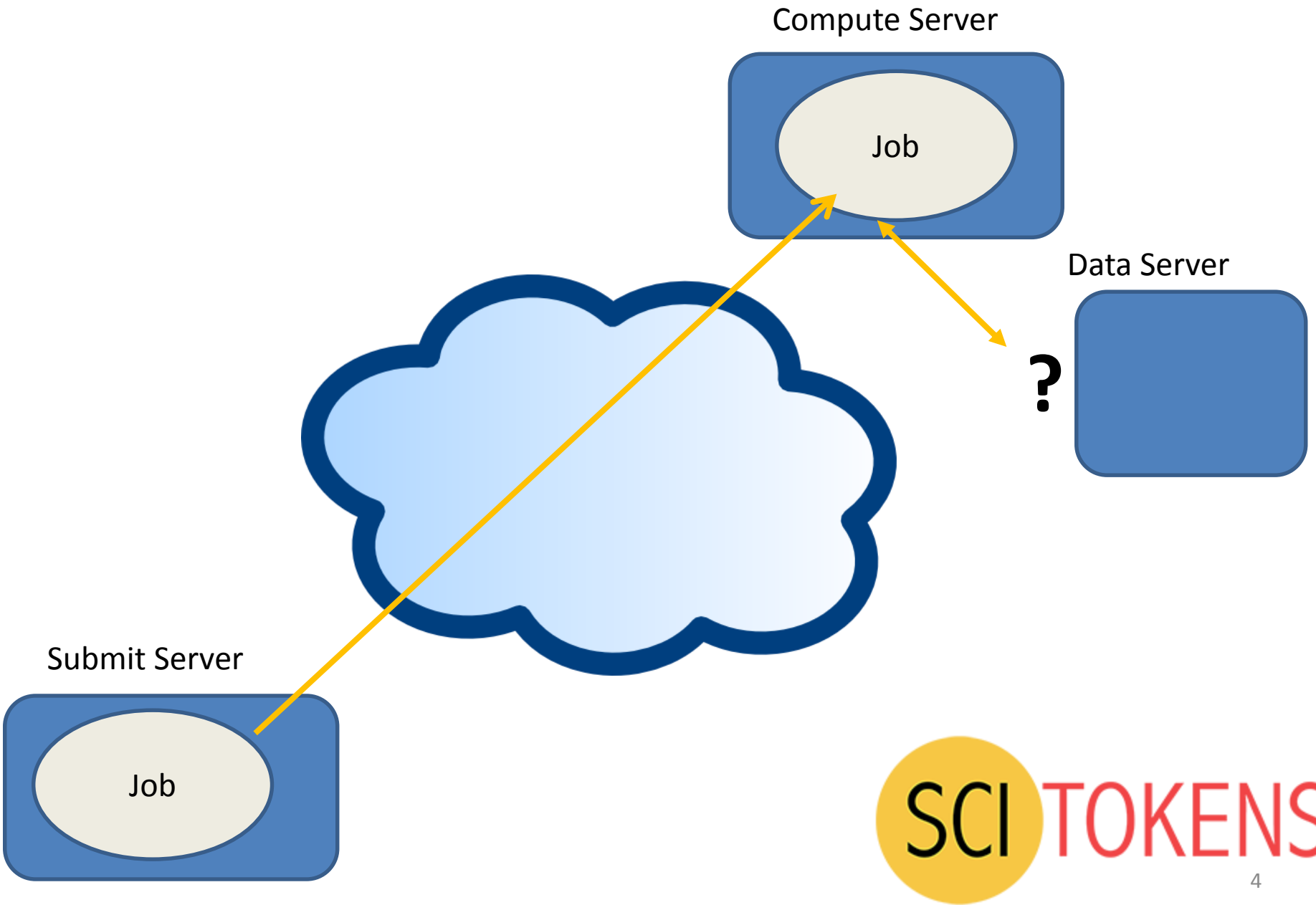
- The SciTokens project, started July 2017, aims to:
  - Introduce a ***capabilities-based*** **authorization infrastructure** for distributed scientific computing,
  - provide a **reference platform**, combining a token library with CILogon, HTCondor, CVMFS, and Xrootd, AND
  - **Deploy this service** to help our science stakeholders (LIGO and LSST) better achieve their scientific aims.
- In this presentation, I'd like to unpack what this means

SCI TOKENS

A common grid computing scenario

- Scientist submits a compute job
- This compute job is scheduled and ultimately starts running on some server out in the grid (or cloud, or HPC center)
- The job requests to read (and/or write) data from some remote data storage service

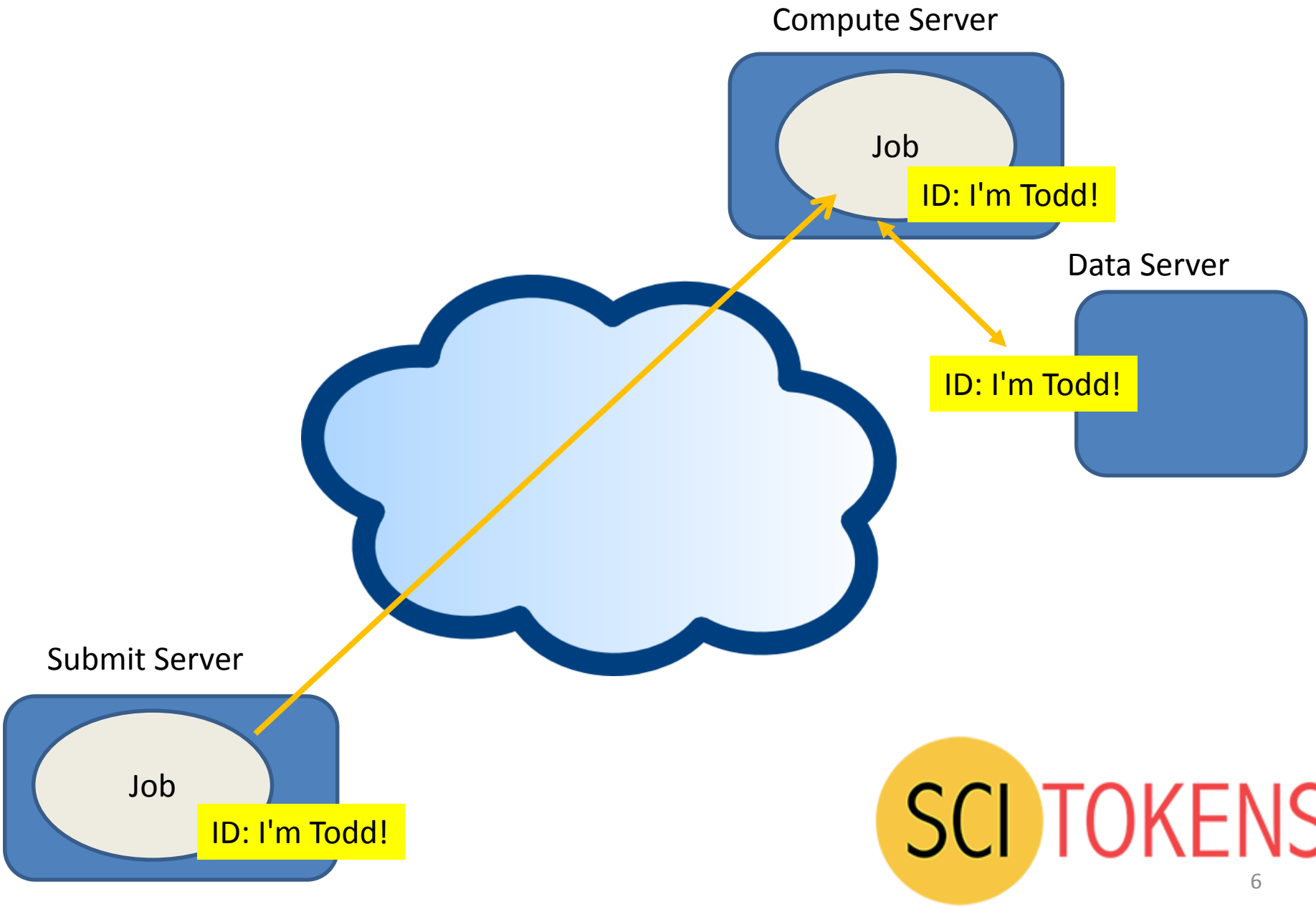
How should the storage service validate the job's request to access the data?


Compute Server
Job
Data Server
?
Submit Server
Job
SCI TOKENS

Identity & Impersonation-based Authorization Infrastructure w/ Certs

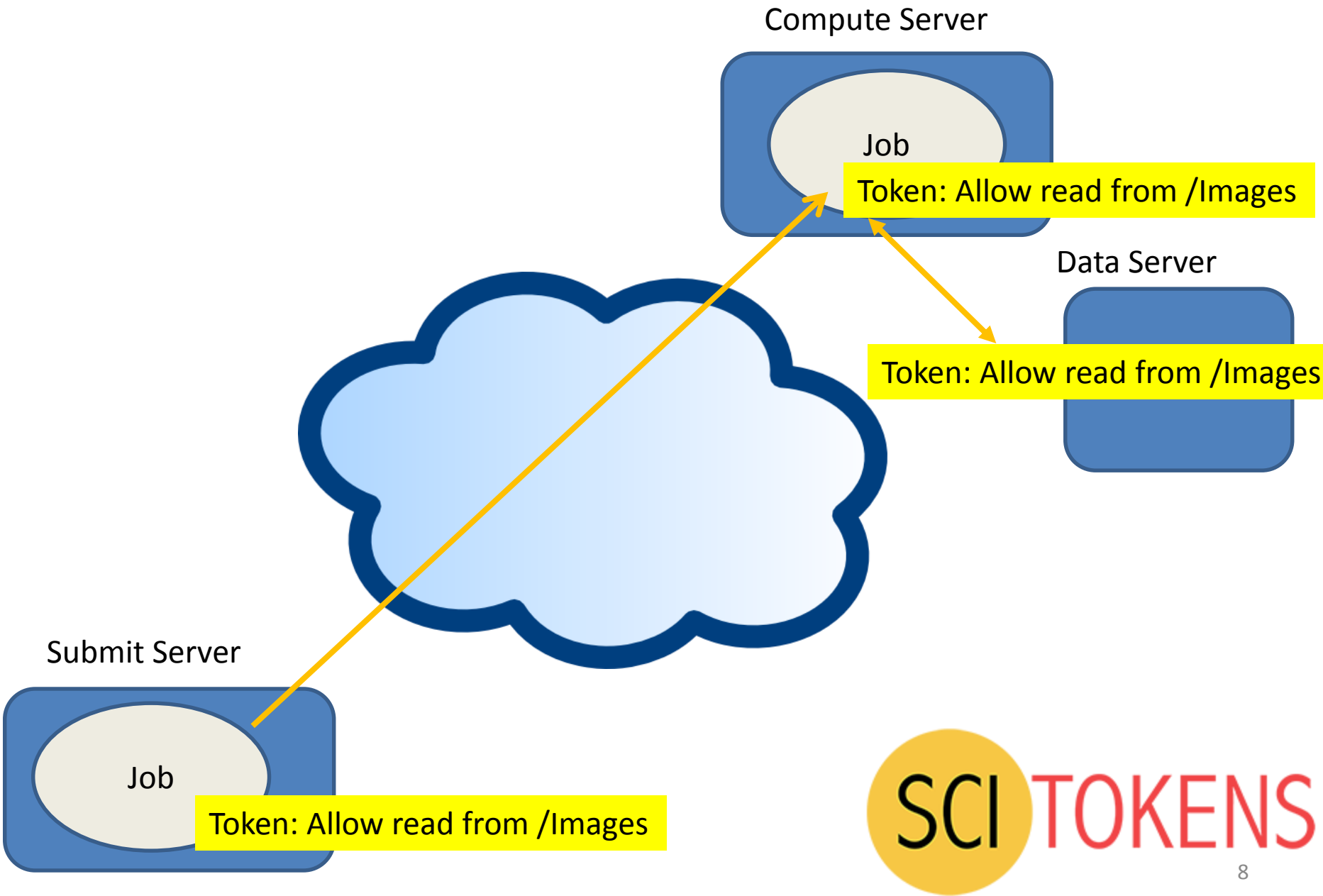
- Common grid solution used today: *identity* and *impersonation* via X.509 certificates.
  - Each user is assigned a grid certificate providing you with a globally-recognized identification.
  - The grid proxy, shipped with the job, allows a third party to impersonate you, (ideally) on your behalf.
  - The remote service maps your identity to some set of locally defined authorizations.
- Not ideal for a few reasons: Not least privilege (what if identity is stolen?), global identity complicates life…

SCI TOKENS


Compute Server
Job
ID: I'm Todd!
Data Server
ID: I'm Todd!
Submit Server
Job
ID: I'm Todd!
SCI TOKENS

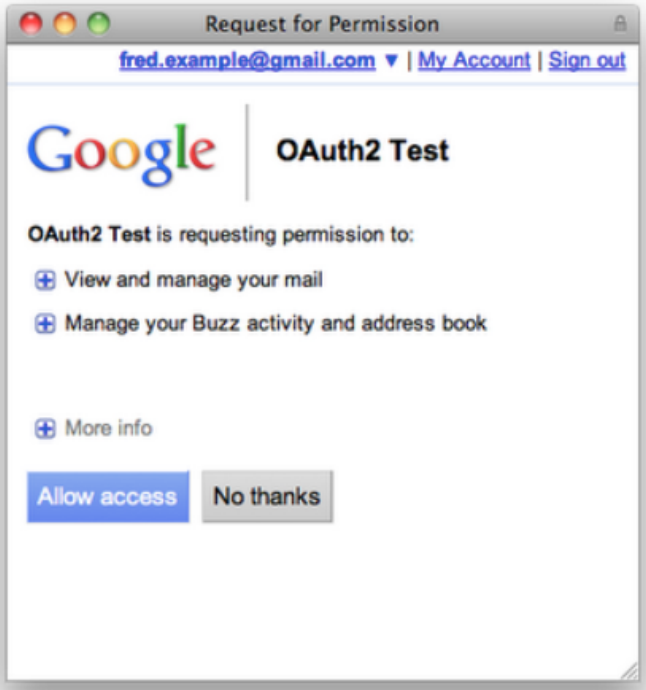
Capabilities-based Authorization Infrastructure w/ tokens

- We want to change the infrastructure to focus on *capabilities*!
  - The tokens passed to the remote service describe what authorizations the bearer has.
  - For traceability purposes, there may be an identifier that allows tracing of the token bearer back to an identity.
  - Identifier != identity. It may be privacy-preserving, requiring the issuer (VO) to provide help in mapping.
- Example: “The bearer of this piece of paper is entitled to read image files from /LSST/datasets/DecemberImages".

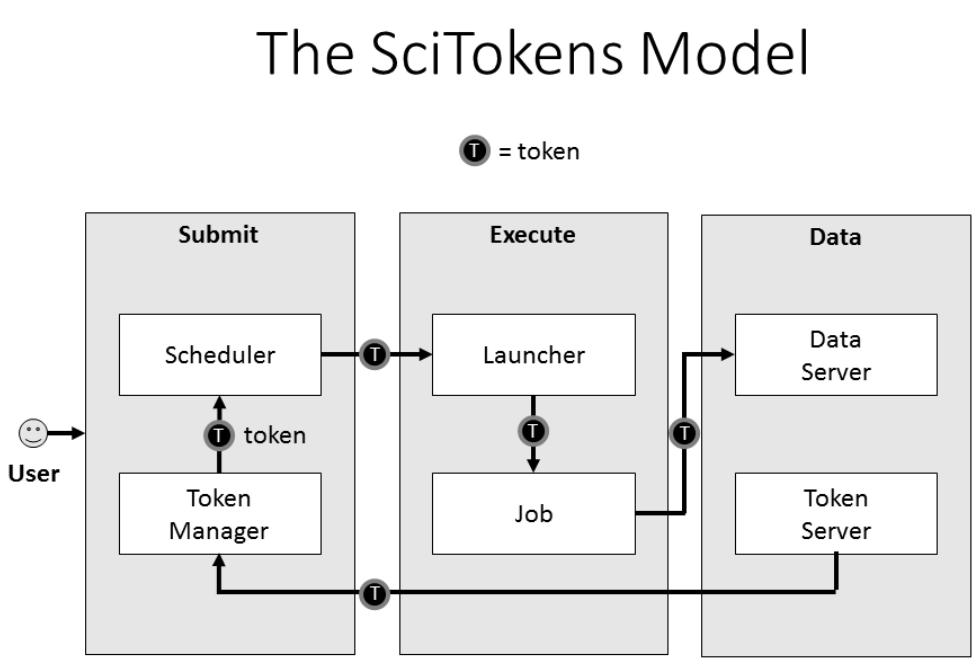

Compute Server
Job
Token: Allow read from /Images
Data Server
Token: Allow read from /Images
Submit Server
Job
Token: Allow read from /Images
SCI TOKENS

The rest of the world uses capabilities!

- The rest of the world uses capabilities for distributed services.
  - The authorization service creates a token that describes a certain capability or authorization.
  - Any bearer of that token may present it to a resource service and utilize the authorization.
- The primary way this is implemented is through OAuth2.
- When you click "allow access" on the right, the **client** at "OAuth2 Test" will receive a token. This token will permit it to access the listed subset of Google services for your account.
- OAuth2 is used by Microsoft, Facebook, Google, Dropbox, Box, Twitter, Amazon, GitHub, Salesforce (and more) to allow distributed access to their identity services.



SciTokens Project Reference Platform

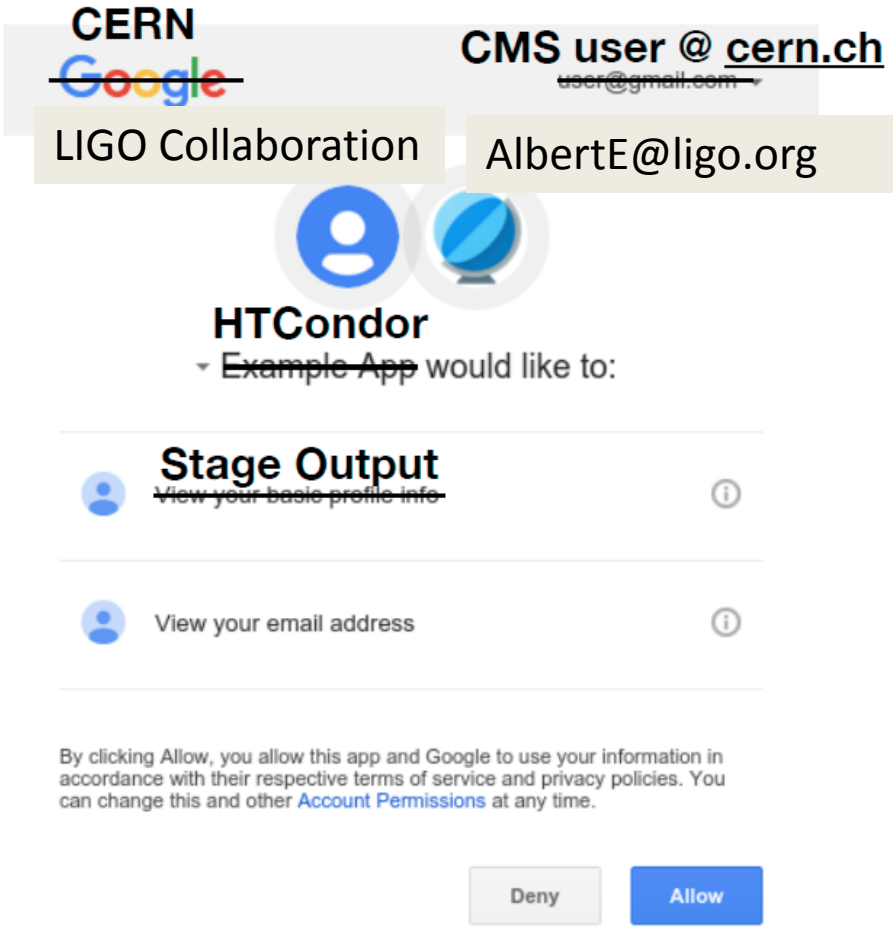


- SciTokens team working to integrate an OAuth2 client into HTCondor submit host, and enhance OAuth2 at CILogon.
- HTCondor being enhanced to manage the token lifetime (refreshing as needed), possibly attenuating it, and delivering to the job.
- Data services (CVMFS, Xrootd) are being enhanced to allow read/writes utilizing tokens instead of grid proxy certificates.

SCI TOKENS

End-Goal will look like this

- The first time you use HTCondor, you navigate to a web interface and setup your desired permissions.
  - On every subsequent job submission, HTCondor will transparently create the access token for you. *User sees nothing.*
- Replace LIGO, usernames, permissions as desired.
- Goal: our first use of OAuth will be to send job output to Box.com storage



SCI TOKENS

SCITOKENS

~~USER MANAGEMENT OF FILES~~

~~PASSWORD IN TERMINAL~~

~~SCITOKENS-PROXY-INIT~~

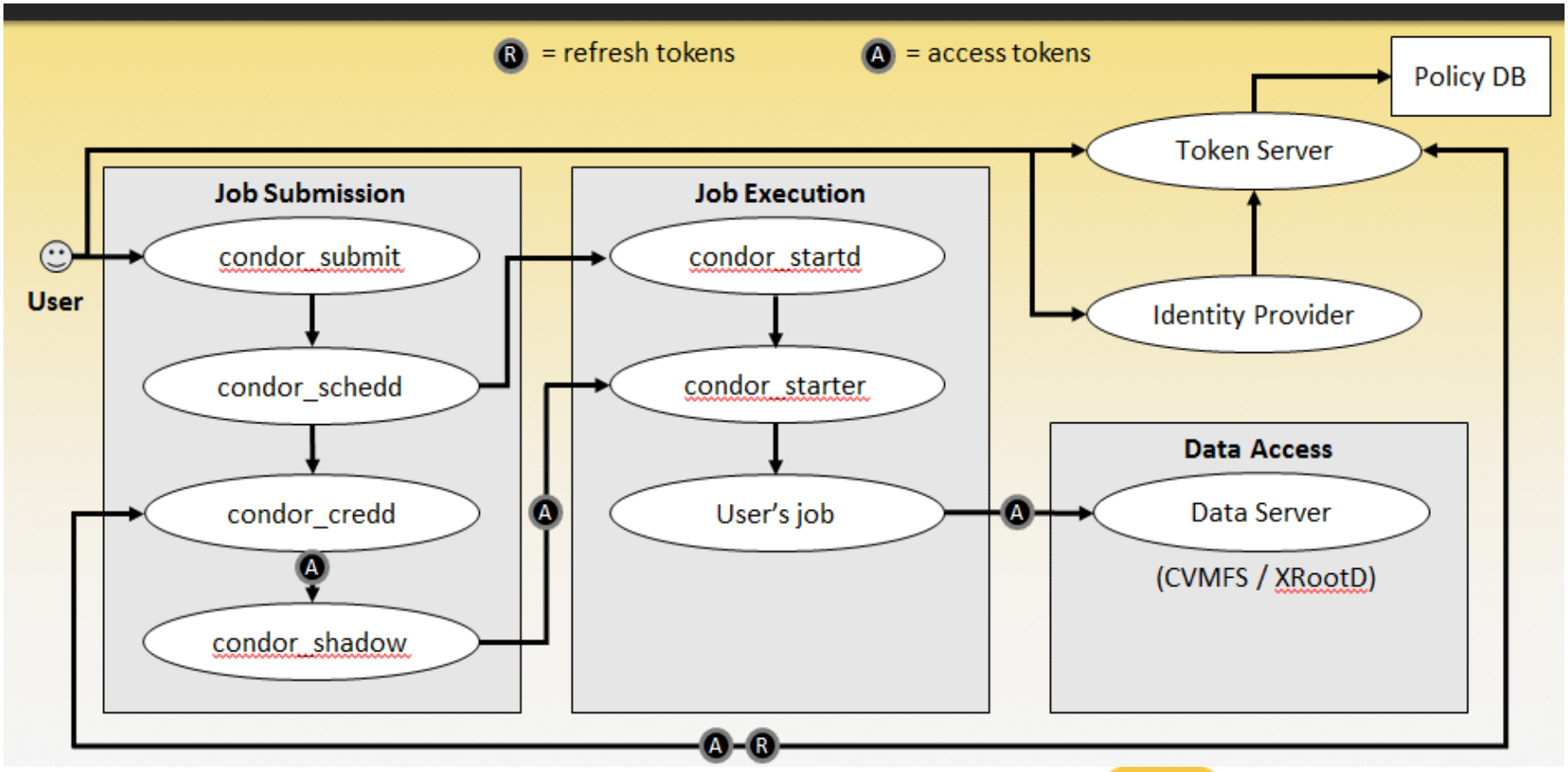
~~COPY/PASTE~~

Leverage Relevant RFC Standards

- We don’t exist in a vacuum, we'd rather adopt existing industry standards than create new ones.
- Workflows for acquiring/using tokens: **OAuth 2.0**.
  - Think of OAuth2 as describing how the various parties should interact
- Access Tokens: **JWT bearer tokens.**
  - The contents (*claims*) of JWT tokens are not standardized, so we will provide a SciTokens Claims specification and reference library implementation so tokens issued by an organization are understood by a wide variety of resources. SciTokens Library also supports
    - Distributed verification
    - Privacy preservation

SCI TOKENS

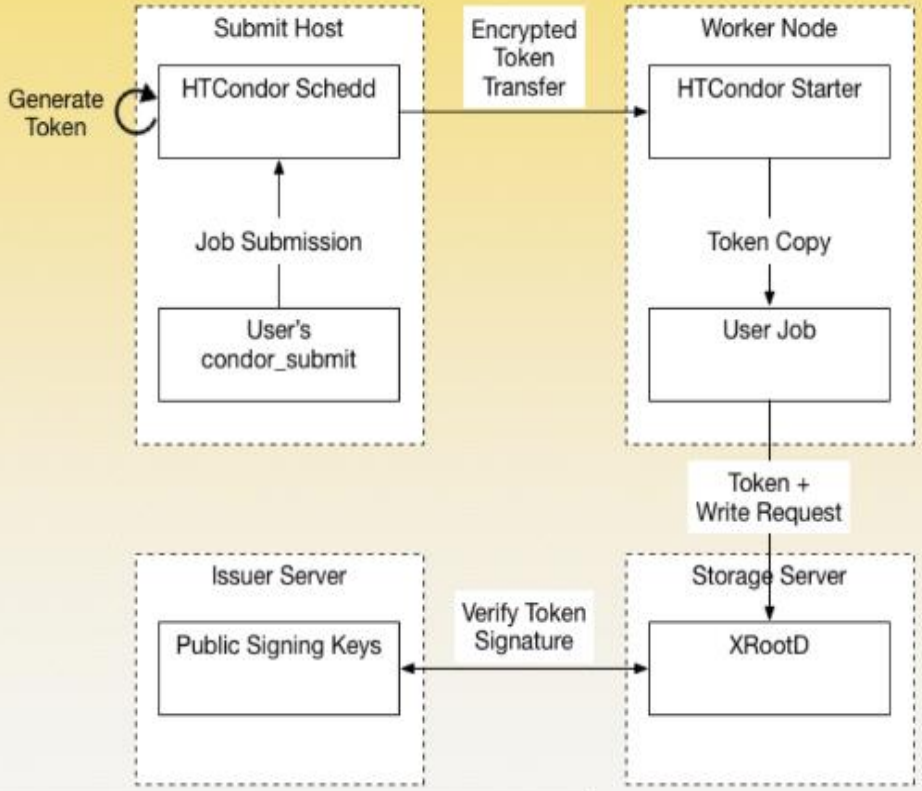
Architecture



SCI TOKENS

Early results on OSG

- 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

SCI TOKENS

Status and Next Steps

- So far we have:
  - Version 1.0 of Python and Java libraries
  - Simple HTCondor OAuth client implementation
  - XRootD token validation plugins
  - Token-based CVMFS access
  - X509-to-SciToken translation service
  - 3rd-party HTTPS FTS transfers authorized with SciTokens
- Next steps:
  - Use Java library for a dCache authorization plugin
  - Release plugin for CVMFS support
  - More fine-grained token management in HTCondor
  - Integration with LIGO LDAP
  - Enhancing HTCondor token support with OAuth flows

TL;DR

- The SciTokens project aims to:
  - Introduce a ***capabilities-based*** **authorization infrastructure** for distributed scientific computing,
  - provide a **reference platform**, combining a token library with CILogon, HTCondor, CVMFS, and Xrootd, AND
  - **Deploy this technology** to help our science stakeholders (LIGO and LSST) better achieve their scientific aims.

- Note: SciTokens does not do everything… e.g. SciTokens does not manage your identity (still need an identity management solution), nor does SciTokens provide an authorization service. But it will enable taking existing solutions and scale them out of distributed grid infrastructure.

SCI TOKENS

We are working with a lot of people

- Stakeholders: LIGO, LSST
- Technologies: HTCondor, CILogin, CVMFS, XRootD, FTS
- Discussions and Interest from Open Science Grid (OSG), LHC WLCG (Worldwide LHC Computing Grid), CMS, CERN IT
- You?

SCITOKENS

SCITOKENS

Feel free to contact us

- Web Site (includes email lists):
  https://scitokens.org
- PI/co-PI contact info appears below
- Questions and Thank You!

Jim Basney
Senior Research Scientist
NCSA / UIUC
jbasney@illinois.edu

Brian Bockelman
Asst. Research Professor, OSG Coord.
U of Nebraska-Lincoln
bbockelm@cse.unl.edu

Duncan Brown
Charles Brightman Professor of Physics
Syracuse University
dabrown@syr.edu

Todd Tannenbaum
Researcher, HTCondor Technical Lead
UW-Madison
tannenba@cs.wisc.edu

Alexander Withers
Senior Security Engineer
NCSA / UIUC
alexw1@illinois.edu