Building the Modern Research Data Portal

Developer Tutorial
Thank you to our sponsors!

U.S. DEPARTMENT OF ENERGY

THE UNIVERSITY OF CHICAGO

NATIONAL INSTITUTES OF HEALTH

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Argonne
NATIONAL LABORATORY

powered by
amazon web services
Presentation material available at

www.globusworld.org/workshop2016

bit.ly/globus-2016
Welcome to the GlobusWorld Tour!

We're presenting a series of Globus tutorials and developer workshops across the US, building on the success of the workshop held at GlobusWorld 2016. These workshops are made possible by the various hosting institutions that generously provide meeting space and other financial support.

The following workshops are currently scheduled:

- **September 13-14, 2016** - LBNL, Berkeley, CA
- **October 12-13, 2016** - Yale University, New Haven, CT
- **Date TBD** - NCAR, Boulder, CO

*If you would like to host a workshop at your institution please contact us.*

**Motivation:** New high-speed networks make it possible, in principle, to transfer and share research data at tremendous speeds and scales—but have also proved challenging to use in practice. Two new technologies now allow us to translate this potential into reality: Science DMZ architectures provide frictionless end-

---

**Why Attend?**

- Learn how the Globus platform simplifies development of web applications for researchers
- Experiment with new Globus services and APIs
- Exchange ideas with peers on ways to apply Globus technologies
- Expand your knowledge of Globus administration features

*Workshops are free to attend and open to all, but we do require registration since*
Building the Modern Research Data Portal

Introduction
Cloud has transformed how software and platforms are delivered.

Software as a service: **SaaS**
(web & mobile apps)

Platform as a service: **PaaS**

Infrastructure as a service: **IaaS**

PaaS enables more rapid, cheap, and scalable delivery of powerful (SaaS) apps.
Research data management simplified.

135,196,155,372 MB TRANSFERRED

Researchers
Focus on your research, not IT problems. We make it easy to move, manage, and share big data.

LEARN MORE  
GET STARTED

Resource Providers
Globus gives you more control over your data infrastructure, while providing excellent ease-of-use for your researchers.

LEARN MORE
GLOBUS PROVIDER PLANS

Our Users
Researchers and resource providers are our greatest inspiration and we love it when they say nice things about Globus.

USER QUOTES
CASE STUDIES

Fast, Reliable, Secure File Transfer
Move files between your laptop, lab server, research computing center, national supercomputing facility, or any other storage system, using just a browser.

LEARN MORE ABOUT FILE TRANSFER WITH GLOBUS

UPCOMING EVENTS

October 16, 2015

Webinar: Integrating Globus into the GridChem Gateway
Globus SaaS: Research data lifecycle

1. **Researcher initiates transfer request; or requested automatically by script, science gateway.**

2. **Globus transfers files reliably, securely.**

3. **Researcher selects files to share, selects user or group, and sets access permissions.**

4. **Globus controls access to shared files on existing storage; no need to move files to cloud storage!**

5. **Collaborator logs in to Globus and accesses shared files; no local account required; download via Globus.**

6. **Researcher assembles data set; describes it using metadata (Dublin core and domain-specific).**

7. **Curator reviews and approves; data set published on campus or other system.**

8. **Peers, collaborators search and discover datasets; transfer and share using Globus.**

- Only a Web browser required
- Use storage system of your choice
- Access using your campus credentials
Demo (from end-user perspective)

- Logging into Globus with any identity
- Endpoint search
- Transfer
- HTTPS access
- Sharing with any identity
- Management Console
Platform Questions

• How do you leverage Globus services in your own applications?

• How do you extend Globus with your own services?

• How do we empower the research community to create an integrated ecosystem of services and applications?
### NCEP Climate Forecast System Version 2 (CFSv2) Monthly Products

**ds094.2**

For assistance, contact Bob Dattore (303-497-1825).

#### Data Description

<table>
<thead>
<tr>
<th>Data Description</th>
<th>Data File Downloads</th>
<th>Customizable Data Requests</th>
<th>Other Access Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Web Server Holdings</td>
<td>Subsetting</td>
<td>THREDDS Data Server</td>
</tr>
<tr>
<td><strong>Globus Transfer Service (GridFTP)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Union of Available Products

<table>
<thead>
<tr>
<th>Product</th>
<th>Web File Listing</th>
<th>Request Globus Invitation</th>
<th>Get a Subset</th>
<th>TDS Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diurnal monthly means</td>
<td>Web File Listing</td>
<td>Request Globus Invitation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regular monthly means</td>
<td>Web File Listing</td>
<td>Request Globus Invitation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Selected Parameter/Level Time Series</td>
<td>Web File Listing</td>
<td>Request Globus Invitation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### NCAR-Only Access

<table>
<thead>
<tr>
<th></th>
<th>Central File System (GLADE) Holdings</th>
<th>Tape Archive (HPSS) Holdings</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
<tr>
<td>GLADE File Listing</td>
<td>HPSS File Listing</td>
<td></td>
</tr>
</tbody>
</table>
Demo

Sample

Research Data Portal
Globus PaaS

- Data Publication & Discovery
- File Sharing
- File Transfer & Replication
- Auth & Groups
- Globus Toolkit
Prototypical research data portal

- Desktop
  - Browser
  - Globus Web Helper Pages
  - Globus Auth
  - Portal Web Server (Client)
- Firewall
  - Identity Provider
  - Globus Cloud
  - Globus Transfer Service
- Other Services
  - Other Endpoints
  - GridFTP
  - Science DMZ
  - Portal Endpoint
  - Other Endpoints

HTTPS
REST
Prototypical research data portal

1. Browser
2. Desktop
3. HTTPS
4. Firewall
5. Globus Web Helper Pages
6. Globus Auth
7. Portal Web Server (Client)
8. Globus Transfer Service
9. Other Services
10. GridFTP
11. Other Endpoints
12. Portal Endpoint
13. Science DMZ
14. Identity Provider
15. Identity Provider
16. Identity Provider
Introduction to REST APIs

• Remote operations on resources via HTTPS
  – POST ~= Create (or other operations)
  – GET ~= Read
  – PUT ~= Update
  – DELETE ~= Delete

• Globus APIs use JSON for documents and resource representations

• Resource named by URL
  – Query params allow refinement (e.g., subset of fields)

• Requests authorized via OAuth2 access token
  – Authorization: Bearer asdflkqhafsdafewk
Globus Transfer API

• Nearly all Globus Web App functionality implemented via public Transfer API

https://docs.globus.org/api/transfer/

• Overview…

• Fairly stable, but small changes coming
  – Deprecation policy
Globus Python SDK

• Python client library for the Globus Auth and Transfer REST APIs

http://globus.github.io/globus-sdk-python/

• Overview…

• Public beta, likely to change some
TransferClient class

- `globus_sdk.TransferClient` class

```python
from globus_sdk import TransferClient
tc = TransferClient()
```

- Handles connection management, security, framing, marshaling
TransferClient low-level calls

- Thin wrapper around REST API
  - `post()`, `get()`, `update()`, `delete()`

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>get()</code></td>
<td><code>get(path, params=None, headers=None, auth=None, response_class=None)</code></td>
</tr>
<tr>
<td><code>path</code></td>
<td>path for the request, with or without leading slash</td>
</tr>
<tr>
<td><code>params</code></td>
<td>dict to be encoded as a query string</td>
</tr>
<tr>
<td><code>headers</code></td>
<td>dict of HTTP headers to add to the request</td>
</tr>
<tr>
<td><code>response_class</code></td>
<td>class for response object, overrides the client’s default_response_class</td>
</tr>
<tr>
<td></td>
<td>Returns: GlobusHTTPResponse object</td>
</tr>
</tbody>
</table>
TransferClient higher-level calls

- One method for each API resource and HTTP verb
- Largely direct mapping to REST API

```python
def endpoint_search(filter_fulltext=None, filter_scope=None, num_results=25, **params)
```
Python SDK Jupyter notebook

• Jupyter (iPython) notebook demonstrating use of Python SDK

https://github.com/globus/globus-jupyter-notebooks

• Overview…

• Open source, enjoy
Endpoint Search

- **Plain text search for endpoint**
  - Searches owner, display name, keywords, description, organization, department
  - Full word and prefix match

- **Limit search to pre-defined scopes**
  - all, my-endpoints, recently-used, in-use, shared-by-me, shared-with-me

- **Returns:** List of endpoint documents
Endpoint Management

• Get endpoint (by id)
• Update endpoint

• Create & delete (shared) endpoints
• Manage endpoint servers
Endpoint Activation

- Activating endpoint means binding a credential to an endpoint for login
- Globus Connect Server endpoint that have Myproxy or MyProxy OAuth identity provider require login via web
- Auto-activate
  - Globus Connect Personal and shared endpoints use Globus-provided credential
  - An endpoint that shares an identity provider with another activated endpoint will use credential
- Must auto-activate before any API calls to endpoints
File operations

- List directory contents (ls)
- Make directory (mkdir)
- Rename

- Path encoding & UTF gotchas
- Don’t forget to auto-activate first
Task submission

- Asynchronous operations
- Get submission_id, followed by submit
  - Once and only once submission

- Transfer
  - Sync level option

- Delete
Task management

• Get task by id
• Get task_list
• Update task by id (label, deadline)
• Cancel task by id
• Get event list for task
• Get task pause info
Bookmarks

• Get list of bookmarks
• Create bookmark
• Get bookmark by id
• Update bookmark
• Delete bookmark by id

• Cannot perform other operations directly on bookmarks
  – Requires client-side resolution
Shared endpoint access rules (ACLs)

- Get list of access rules
- Get access rule by id
- Create access rule
- Update access rule
- Delete access rule
- Access manager role
Management API

- Allow endpoint administrators to monitor and manage all tasks with endpoint
  - Task API is essentially the same as for users
  - Information limited to what they could see locally
- Cancel tasks
- Pause rules
Join the Globus developer community

- Join developer-discuss@globus.org mailing lists
  
  https://www.globus.org/mailing-lists

- Python SDK is open source
  - https://github.com/globus/globus-sdk-python
  - Submit issues, pull requests
  - Discussions on developer-discuss@globus.org

- Jupyter notebook & sample data portal are also open source on github
Building the Modern Research Data Portal

Exercises:
Transfer API in Jupyter
Install Jupyter notebook

- Either locally or on EC2 instance

https://github.com/globus/globus-jupyter-notebooks.git

- EC2 instance login:
  - Username:
  - Password:
Transfer API exercises
Modify Jupyter notebook to:

1. Find the endpoint id for XSEDE Comet
2. Set all the metadata fields on your shared endpoint
3. Modify mkdir so that an existing directory does not raise an exception, but all other errors do.
4. Set access manager role on your shared endpoint, and query both roles and ACLs to see the result.
5. Perform an ls given a bookmark name.
6. Perform a transfer akin to ‘rsync –av –delete’.
7. Transfer all files in a directory named *.txt to another endpoint.
8. Perform a transfer, monitor for completion, and monitor the event log. If a fault occurs, then cancel the job for some fault types (e.g., file not found), but not others (e.g., permission denied).
9. Anything else you want to try out...
Prototypical research data portal

- Browser
- Desktop
- User’s Endpoint (optional)
- Portal Endpoint
- Science DMZ
- Gateway
- Firewall
- Globus Web Helper Pages
- Globus Auth
- Globus Transfer Service
- Other Services
- Globus Cloud
- Identity Provider
- GridFTP
- REST
- HTTPS
ESnet Slides
Prototypical research data portal

- Desktop
- Browser
- Globus Cloud
  - Globus Web Transfer Service
  - Globus Web Helper Pages
  - Globus Auth
- Firewall
- Portal Web Server (Client)
  - REST
- Portal Endpoint
- GridFTP
- Other Endpoints
- Science DMZ
- Identity Provider
- HTTPS
- Other Services
- User’s Endpoint (optional)
Challenge

• **How to provide:**
  – Login to apps
    o Web, mobile, desktop, command line
  – Protect all REST API communications
    o App → Globus service
    o App → non-Globus service
    o Service → service

• **While:**
  – Not introducing even more identities
  – Providing least privileges security model
  – Being agnostic to programming language and framework
  – Being web friendly
  – Making it easy for users and developers
Globus Auth

• Foundational identity and access management (IAM) platform service
• Simplify creation and integration of advanced apps and services
• Brokers authentication and authorization interactions between:
  – end-users
  – identity providers: InCommon, XSEDE, Google, portals
  – services: resource servers with REST APIs
  – apps: web, mobile, desktop, command line clients
  – services acting as clients to other services
Based on widely used web standards

- **OAuth 2.0 Authorization Framework**
  - aka OAuth2

- **OpenID Connect Core 1.0**
  - aka OIDC

- **Allows use of standard OAuth2 and OIDC libraries**
  - E.g., Google OAuth Client Libraries (Java, Python, etc.), Apache mod_auth_openidc
Globus Auth

- Identity and access management PaaS

https://docs.globus.org/api/auth/

- Introduction
- Reference
Globus account

• A Globus account is a set of identities
  – A primary identity
    o Identity can be primary of only one account
  – One or more linked identities
    o Identity can (currently) be linked to only one account

• Account does not have own identifier
  – An account is uniquely identified using its primary identity
Globus Auth interactions

- App (Client)
- Service (Resource Server)
- Authorization Server (Globus Auth)
- Identity Provider

Flow:
- Resource Owner
- Login
- HTTPS/REST call

Diagram:
- App (Client) -> Authorization Server (Globus Auth) -> Service (Resource Server)
Globus Auth interactions

(1) Request authorization

- For a set of scopes
  - Login: openid, email, profile
  - HTTPS/REST APIs
- User selects identity provider
Globus Auth interactions

(1) Request authorization
(2) Authenticates a resource owner

- Using existing identities:
  - XSEDE, University (via InCommon), Google, web app, etc.
- User can link multiple identities into a single Globus Account
- No Globus username & password (Globus ID) required
- Globus Auth handles naming details (e.g., ePPN vs ePTID)
Globus Auth interactions

1. Request authorization
2. Authenticates a resource owner
3. Obtains authorization (consent) for a client to access a resource

- Resource is provided by a resource server
- Limited by a scope
(1) Request authorization
(2) Authenticates a resource owner
(3) Obtains authorization (consent) for a
   client to access a resource
(4) Issues OAuth2 access_token to client

• Some grant types issue authorization code, which
  client exchanges for access token
• Access token is opaque to client
• May include a refresh token, for offline access
Globus Auth interactions

(1) Request authorization
(2) Authenticates a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity

JWT id_token:
sub: Globus Auth identity id
iss: https://auth.globus.org
name: full name
preferred_username: e.g., tuecke@uchicago.edu
e-mail: email contact
other standard OIDC claims
Globus Auth interactions

(1) Request authorization
(2) Authenticates a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity
(6) HTTPS/REST call with access_token
Globus Auth interactions

(1) Request authorization
(2) Authenticates a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity
(6) HTTPS/REST call with access_token
(7) Validates access_token for resource server, and gets additional information

RFC 7662: OAuth 2.0 Token Introspection response:
- active: true or false
- client_id
- scope
- sub: Globus Auth identity id
- username: user@example.com
- identity_set: linked identities
- email
- name
- other standard claims
Globus Auth interactions

(1) Request authorization
(2) Authorizes a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity
(6) HTTPS/REST call with access_token
(7) Validates access_token for resource server, and gets additional information
(8) Issues dependent access tokens to resource server

Allows resource server to act as client to other resource servers

Service uses request access_token to get a dependent access_token for each dependent service

Service acts as client to its dependent services
Log in with Globus

• Similar to Log in with Google and Log in with Facebook
• Using existing identities
• Providing access to community services
Demo

Log in to Jetstream App
OAuth2 / OIDC client

• Globus Auth should work with any compliant client
  – We recommend Google OAuth client libraries
  – Python, Java, PHP, Javascript, .NET

https://developers.google.com/api-client-library/
Identity id vs. username

• **Identity id:**
  – Guaranteed unique among all Globus Auth identities, and will never be reused
  – UUID
  – Always use this to refer to an identity

• **Identity username:**
  – Unique at any point in time
    o May change, may be re-used
  – Case-insensitive user@domain
  – Can map to/from id, for user experience

• **Auth API allows mapping back and forth**
Scopes

• APIs that client is requesting access to

• Scope syntax:
  urn:globus:auth:scope:<service-name>:<scope-name>

• If client requests multiple scopes
  – Token response has tokens for first scope
  – other_tokens field in response has list of token responses for other scopes
  – Client must use correct token with each request
Effective identity

- App can choose to operate only with identities from a particular identity provider
  - Globus Auth login will require an identity from that provider to be linked to user’s account
  - OIDC id_token uses this “effective identity”

- If app does not set an effective identity policy, then the primary identity of the account is used as the effective identity for that app
App registration

• **Client_id and client_secret** for service
• **App display name**
• **Declare required scopes**
  – Need long-term, offline refresh tokens?
  – May require authorization from scope admin
• **OAuth2 redirect URIs**
• **Links for terms of service & privacy policy**
• **Effective identity policy (optional)**
Demo

App Registration
Portal accounts

• Your app portal can still have portal accounts for users
• Tie portal account to Globus account identity, rather than username/password
• Associate your profile with this account
• Globus Auth handles authentication of that identity, in order to log user into your portal account
User identity vs portal identity

• User logging into portal results in portal having user’s identity and access token
  – Used to make requests on the user’s behalf

• Portal may also need its own identity
  – Access and refresh tokens for this identity
  – Used to make requests on its own behalf
Client identity

- Portal App has client_id & client_secret
- Globus Auth client_id is an identity_id
  - <client_id>@clients.auth.globus.org
- Use OAuth2 Client Credentials Grant to authenticate the client identity
  - Using client_id and client_secret
- Can use the client_id just like any other identity_id
  - Sharing access manager role, permissions, group membership, etc.
Consent

• Resource owner authorization that a client can request access to a service on the resource owner's behalf within a limited scope
  – If service has dependent scopes, they are part of the consent

• User can rescind a consent at any time
  – Invalidates all access, dependent, and refresh tokens originating from the client
Prototypical research data portal

- Desktop
  - Browser
  - User's Endpoint (optional)
  - Portal Web Server (Client)
  - Portal Endpoint
  - Identity Provider
  - Globus Web Helper Pages
  - Globus Auth
  - Globus Transfer Service
  - Other Services
  - Other Endpoints
  - Science DMZ
  - Firewall
  - HTTPS
  - REST
  - GridFTP
Adding your campus identity provider to Globus

- InCommon identity providers that release research & scholarship attributes to CILogon (free)

- OpenID Connect identity provider supported by Globus Auth (subscription)
Adding an identity provider

- If your portal has identities already:
  - Deploy OIDC server in front of it
    - Globus Python OIDC (coming soon)
    - Any standard OIDC server should work
    - Requires claim that can map to username
    - Optional claims: name, email, organization
  - Can register apps and services with an effective identity policy
    - Requires account to have identity from your identity provider when logging into your app
Branding

- Can skin Globus Auth pages
Prototypical research data portal

- Browser
  - Globus Web Helper Pages
  - Globus Auth
  - Portal Web Server (Client)
  - Portal Endpoint
  - User’s Endpoint (optional)
  - Other Endpoints
  - Science DMZ
  - GridFTP
  - REST
  - HTTPS

- Globus Cloud
  - Globus Transfer Service
  - Other Services

- Identity Provider

- Desktop

- Firewall
Introduction to Globus Helper Pages

- Globus provided web pages designed for use by your web apps
  - Browse Endpoint
  - Select Group
  - Logout

https://docs.globus.org/api/helper-pages/
Client Logout

• **Call token revocation on access tokens**
  – https://auth.globus.org/v2/oauth2/token/revoke
  – Doc: [https://docs.globus.org/api/auth/reference/](https://docs.globus.org/api/auth/reference/)
  – Note: Does not revoke dependent tokens

• **Delete access tokens**

• **Redirect to logout helper page**
  – https://auth.globus.org/v2/web/logout
  – Doc: [https://docs.globus.org/api/helper-pages/](https://docs.globus.org/api/helper-pages/)
Building the Modern Research Data Portal

Exercises:
Install and run your own sample research data portal
Sample Research Data Portal Code Walk-through
Install sample data portal

- Either locally or on EC2 instance


- EC2 instance login:
  - Username:
    Password:
Portal App exercises
Find and print to console:

1. Globus Auth URL the portal redirects to for login
2. Globus Auth URL the portal redirects to for logout
3. Username of the logged in user
4. Complete id_token of the logged in user
5. URL of the Globus Browse Endpoints helper page used by the portal
6. Endpoint and path selected by user as destination of the transfer
7. URL to submit transfer, and resulting task id
8. Complete task document returned by status
HTTPS to Endpoints

• Each endpoint HTTPS server is a Globus Auth service (resource server)

• Web page can link to file on server
  – Browser GET will cause HTTPS server to authorize request via Globus Auth (note SSO)

• Portal (client) can request scope for endpoint resource server
  – Use access token in requests
Prototypical research data portal

- Globus Cloud
  - Globus Web Helper Pages
  - Globus Auth
  - Portal Web Server (Client)
  - Other Services
- Other Endpoints
- Science DMZ
- Portal Endpoint
- Other Endpoints
- GridFTP
- REST
- HTTPS
- Desktop
  - User’s Endpoint (optional)
  - NON-Browser

Identity Provider

- Firewall
Mobile apps

• Globus Auth adding support for mobile apps
  – “Log in with Globus” in mobile apps
    o RFC 7636: Proof Key for Code Exchange by OAuth Public Clients (PKCE, pronounced “pixy”)
    o Extension to OAuth2 to allow OAuth2 Authorization Code Grant to work from mobile apps
  – Uses mobile browser for web-based login
  – Mobile apps can call any service REST APIs that use Globus Auth
  – iOS and Android
  – Same approach as used by Google, Facebook, etc.
Desktop & command line apps

- Globus Auth “Native App” PKCE support
- **Use browser if possible**
  - “OAuth 2.0 for Native Apps”
    - draft-ietf-oauth-native-apps-02
    - Use external browser if possible
    - Embed browser in app
    - Embed mini web server in app
- **Allows copy-n-paste of authorization code**
  - A little like app passwords, but OAuth2 compliant
- **Globus Python SDK and CLI will support Native App login**
- **Limited support for username/password authentication**
  - Not recommended
Prototypical research data portal
Why create your own services?

• **Front-end / back-end within your portal**
  – Remote backend for portal
  – Backend for pure Javascript browser apps

• **Extend your portal with a public REST API, so that other app and service developers can integrate with and extend your portal**
Why Globus Auth for your service?

- Outsource all identity management and authentication
  - Federated identity with InCommon, Google, etc.
- Outsource your REST API security
  - Consent, token issuance, validation, revocation
  - You provide service-specific authorization
- Apps use your service like all others
  - Its standard OAuth2 and OIDC
- Your service can seamlessly leverage other services
- Other services can leverage your service
- Implement your service using any language and framework

*Add your service to the science cyberinfrastructure platform*
Service registration

- Client_id and client_secret for service
- Service display name
- Validated DNS name for service
- One or more scopes
- Authorize clients to use each scope
  - All clients (public API), or specific clients
- Declare dependent scopes
  - Need long-term, offline refresh tokens?
  - May require authorization from scope admin
- Links for terms of service & privacy policy
- Effective identity policy (optional)
Service interactions with Globus Auth

(1) Request authorization
(2) Authenticates a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity
(6) HTTPS/REST call with access_token
(7) Validates access_token for resource server, and gets additional information

RFC 7662: OAuth 2.0 Token Introspection response:
  active: true or false
  client_id
  scope
  sub: Globus Auth identity id
  username: user@example.com
  identity_set: linked identities
  email
  name
  other standard claims
Service interactions with Globus Auth

(1) Request authorization
(2) Authenticares a resource owner
(3) Obtains authorization (consent) for a client to access a resource
(4) Issues OAuth2 access_token to client
(5) May issue OIDC id_token to client with resource owner identity
(6) HTTPS/REST call with access_token
(7) Validates access_token for resource server, and gets additional information
(8) Issues dependent access tokens to resource server

Allows resource server to act as client to other resource servers
Service uses request access_token to get a dependent access_token for each dependent service
Service acts as client to its dependent services
Typical service interactions

- **Service receives HTTPS request with header**
  - Authorization: Bearer <request-access-token>

- **Introspects the request access token**
  - Auth API: POST /v2/oauth2/token/introspect
  - Authorized by client_id and client_secret
  - Returns: validity, client, scope, effective_identity, identities_set

- **Verifies token info**

- **Authorizes request**

- **If service needs to act as client to other services:**
  - Calls Globus Auth Dependent Token Grant
    - Returns a token for each dependent service
    - Uses correct dependent token for downstream REST call

- **Responds to client HTTPS request as appropriate**
Authorization based on identity set

- Use identities_set when authorizing a request based on the resource owner associated with an access token
  - E.g., ACLs on Globus shared endpoints

- Authorizing based on set of identities is same complexity as authorizing based on group membership set
Groups

• **Globus group service is identity set aware**
  – “Tell me all groups for all identities of the logged in user”

• **Services can leverage this for authorization**
Prototypical research data portal

Browser

Globus Cloud

Globus Transfer Service

Globus Web Helper Pages

Globus Auth

Portal Web Server (Client)

Firewall

HTTPS

REST

Globus Web

Identity Provider

Other Services

Other Endpoints

User’s Endpoint (optional)

Portal Endpoint

Science DMZ

GridFTP
Dependent tokens

• Your service can act as client to other services (scopes)
  – Globus Transfer and Auth
  – XSEDE (e.g., Jetstream, XUP)
  – Other community services
  – Future: Commercial services (e.g., Google Drive)

• Entire service call tree consented by user and service owners
  – Rescinding consent revokes all dependent tokens

• Dependent tokens are restricted to a particular client, calling a particular scope, on behalf of a particular resource owner (e.g., user)
  – Restricted delegation!
Refresh tokens

• For “offline services”
  – E.g., Globus transfer service working on your behalf even when you are offline

• Refresh tokens issued to a particular client for use with a particular scope

• Client uses refresh token to get access token
  – Client_id and client_secret required

• Refresh token good for 6 months after last use

• Consent rescindment revokes resource token
Token caching

• **Service should cache tokens and related information**
  - Improves performance of service
  - Reduces load on Globus Auth

• **Access token -> introspect response**
  - Cache timeout: 1-30 seconds recommended
    - To improve performance and load related to bursty use of REST API
  - Validity: Timeout duration determines responsiveness to token revocation and rescinding consent
  - `client, scope, effective_identity`: These will never change for an access token
  - `identities_set`: This may change at any time, due to identity (un)linking. May affect authorization. Timeout duration affect responsiveness to linking changes.
  - Future: add group membership to this, which is dependent on `identities_set`

• **Access token -> dependent access tokens**
  - Cache timeout: lifetime of access token
    - To avoid costly dependent token re-issuance
  - Rescinding consent will invalidate everything

• **Refresh tokens**
  - For however long they are needed for specific operations.
Sample Research Data Portal Service Walk-through
Building the Modern Research Data Portal

Exercises:
Backend service for sample research data portal
Install sample data portal

• Either locally or on EC2 instance


• EC2 instance login:
  – Username:
  Password:
Service exercises

1. Find and print to console:
   1. Expiration time of each of dependent tokens
   2. The complete ACL rule added to the folder for the user
   3. The full response from token introspection

2. Modify cleanup to wait for files to be deleted before returning