Leveraging the Globus Platform
Web Apps, Jupyter and more…

Rachana Ananthakrishnan - rachana@globus.org
Greg Nawrocki - greg@globus.org

Johns Hopkins University
April 11, 2019
Topics and Goals

• Globus Auth

• Jupyter Hub + Globus
  – Interactive data science

• Globus Transfer APIs
  – Globus Jupyter Hub walkthrough

• Data Portal Example
  – You may already be doing something similar
  – You may want to do something like this
Globus delivers… with applications and as a platform…

Fast and reliable data transfer, sharing, publication, and discovery…

…directly from your own storage systems…

…via software-as-a-service using existing identities.
How can I integrate Globus into my research workflows?
Globus serves as...
A platform for building science gateways, portals and other web applications in support of research and education.
Example web apps that leverage Globus
Globus Platform-as-a-Service

Data Search & Discovery

File Sharing

File Transfer & Replication

Globus Auth API

Globus Connect

Integrate file transfer and sharing capabilities into scientific web apps, portals, gateways, etc...

Use existing institutional ID systems in external web applications
PaaS Security Challenges – Globus Auth

• **How to provide:**
  – Login to apps
    o Web apps (Jupyter Notebook, Portals), Mobile, Desktop, Command line
  – Protect all REST API communications
    o App → Globus service (Jupyter Notebook, MRDP)
    o App → non-Globus service (MRDP)
    o Service → service (MRDP)

• **While:**
  – Not introducing even more identities
    o Providing a platform to consolidate those identities
  – Providing least privileges security model (consents)
  – Being agnostic to programming language and framework
  – Being web friendly
  – Making it easy for users and developers
Authorization Code Grant

1. Access portal
   - Client (Web Portal, Application)

2. Redirects user
   - Browser (User)

3. User authenticates and consents
   - Globus Auth (Authorization Server)

4. Authorization code
   - Globus Auth (Authorization Server)
   - Identity Provider

5. Authenticate using client id and secret, send authorization code
   - Globus Transfer (Resource Server)

6. Access token(s)
   - Globus Transfer (Resource Server)

7. Authenticate with access token(s) to give the client the authority invoke the transfer service
   - Client (Web Portal, Application)
JupyterHub + Globus PaaS: A Foundation for Interactive Data Science
JupyterHub

• Multi-user hub
• Manages multiple instances of Jupyter notebook server – Python, R, etc.
• Use it to serve notebooks to research team, class, etc.
• Configurable HTTP proxy

jupyterhub.readthedocs.io/en/stable
Securing JupyterHub with Globus Auth plugin

- Existing OAuth framework
- Can restrict IdP
- Custom scopes
- Tokens passed into notebook environment
- Documentation covers app registration and config

[GitHub link: github.com/jupyterhub/oauthenticator]
Visit [https://developers.globus.org/](https://developers.globus.org/) to set up your app. Ensure *Native App* is unchecked and make sure the callback URL looks like:

https://[your-host]/hub/oauth_callback

Set scopes for authorization and transfer. The defaults include:

`openid profile urn:globus:auth:scope:transfer.api.globus.org:all`

Set the above settings in your `jupyterhub_config`:

```python
# Tell JupyterHub to create system accounts
from oauthenticator.globus import LocalGlobusOAuthenticator
c.JupyterHub.authenticator_class = LocalGlobusOAuthenticator
c.LocalGlobusOAuthenticator.enable_auth_state = True
c.LocalGlobusOAuthenticator.oauth_callback_url = 'https://[your-host]/hub/oauth_callback'
c.LocalGlobusOAuthenticator.client_id = '[your app client id]'  
c.LocalGlobusOAuthenticator.client_secret = '[your app client secret]'
```

[github.com/jupyterhub/oauthenticator#globus-setup](https://github.com/jupyterhub/oauthenticator#globus-setup)
Tokens and JupyterHub

- Globus Auth access tokens passed back to JupyterHub
- Stored as a secure attribute in database
- Passed into Notebook Server environment
- Can be pulled into notebook or other code
  - Notebook can run alone – native app grant
- Use to communicate with...
  - ...Globus services (transfer)
  - ...other REST APIs secured with Globus Auth
Tokens and Jupyter Hub

{ "tokens":... }
Globus Platform
Transfer API
Using the SDK in our Jupyter Hub

https://www.globus.org/blog/using-globus-jupyter-notebooks
Globus Transfer API

- Globus Web App consumes public Transfer API
- Resource named by URL (standard REST approach)
  - Query params allow refinement (e.g., subset of fields)
- Globus APIs use JSON for documents and resource representations
- Requests authorized via OAuth2 access token
  - Authorization: Bearer asdflkqhafsdafeawk

[docs.globus.org/api/transfer](http://docs.globus.org/api/transfer)
Globus Python SDK

- Python client library for the Globus Auth and Transfer REST APIs
- `globus_sdk.TransferClient` class handles connection management, security, framing, marshaling

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

[Link to documentation](globus.github.io/globus-sdk-python)
TransferClient low-level calls

• Thin wrapper around REST API
  – post(), get(), update(), delete()

get(path, params=None, headers=None, auth=None, response_class=None)
  o path – path for the request, with or without leading slash
  o params – dict to be encoded as a query string
  o headers – dict of HTTP headers to add to the request
  o response_class – class response object, overrides the client’s default_response_class
  o Returns: GlobusHTTResponse object
TransferClient higher-level calls

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

endpoint_search(filter_fulltext=None,
               filter_scope=None,
               num_results=25,
               **params)
Walkthrough API with our Jupyter Hub

• **https://jupyter.demo.globus.org**
  – Sign in with Globus
  – Verify the consents
  – Start My Server (this will take about a minute)
  – Open folder: globus-jupyter-notebooks
  – Open folder: GlobusWorldTour
  – Run Platform_Introduction_JupyterHub_Auth.ipynb

• **If you mess it up and want to “go back to the beginning”**
  – Back down to the root folder
  – Run NotebookPuller.ipynb

• **If you want to use the notebook outside of our hub**
  – https://github.com/globus/globus-jupyter-notebooks
  – Authentication is a manual cut and paste of exchanging the authorization code for an access token
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
  – Must auto-activate before any API calls to endpoints
File operations

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

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

• Asynchronous operations
  – Transfer
    o Sync level option
  – Delete

• Get submission_id, followed by submit
  – Once and only once submission
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 endpoints and access rules (ACLs)

• Shared Endpoint – create / delete / get info / get list
• Administrator role required to delegate access managers
• Access manager role required to manage permission/ACLs
• Operations:
  – Get list of access rules
  – Get access rule by id
  – Create access rule
  – Update access rule
  – Delete access rule
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**
Walkthrough API with a Jupyter Notebook

• If you want to use the / a notebook outside of our hub
  – https://github.com/globus/globus-jupyter-notebooks
  – Authentication is a manual cut and paste of the authorization code.
  – Native App grant
• Globus pages designed for use by your web apps
  – Browse Endpoint
  – Activate Endpoint
  – Select Group
  – Manage Identities
  – Manage Consents
  – Logout

docs.globus.org/api/helper-pages
Example
Modern Research Data Portal
https://docs.globus.org/modern-research-data-portal/
Prototypical research data portal
Python SDK

Installation

The Globus SDK requires Python 2.6+ or 3.2+. If it is not installed on your computer, you can install it or view documentation by visiting: https://docs.globus.org/api

globus-sdk-python

Requirements

- You need to be in the tutorial users group for sharing: https://www.globus.org/app/groups/50b6a29c-63e4-8062-22000b92c6ec
- Installed Globus Python SDK

```
from future import print

tutorial_endpoint_1 = "dd59af0-6d04-11e5-ba46-22000b92c6ec" # endpoint "Globus-PAAS"
tutorial_endpoint_2 = "dd59af0-6d04-11e5-ba46-22000b92c6ec" # endpoint "Globus-PAAS"
tutorial_users_group = "50b6a29c-63e4-8062-22000b92c6ec" # group "tutorial"
```

Configuration

First you will need to configure the client with an OAuth2 access token. For the purpose of this tutorial, you can use the "Jupyter Notebook" option and copy the resulting text below, or click on "Globus CLI" and

```
In [16]: transfer_token = None # if None, tries to get token from ~/.globus.cfg file
```
Support resources

• **Globus documentation**: docs.globus.org
• **Helpdesk and issue escalation**: support@globus.org
• **Mailing lists**
  – https://www.globus.org/mailing-lists
  – developer-discuss@globus.org
• **Globus professional services team**
  – Assist with portal/gateway/app architecture and design
  – Develop custom applications that leverage the Globus platform
  – Advise on customized deployment and integration scenarios