Connecting people and resources to accelerate discovery by empowering the science gateway community

SGCI

Using Keycloak to Provide Authentication, Authorization, and Identity Management Services for Your Gateway

Marcus Christie
Science Gateways Research Center
Indiana University
EDS Consultant

Award Number
ACI-1547611

Overview

- OpenID Connect with CILogon
- Using Keycloak for Authentication and Role-based Authorization
- Integrating Keycloak with Apache Airavata

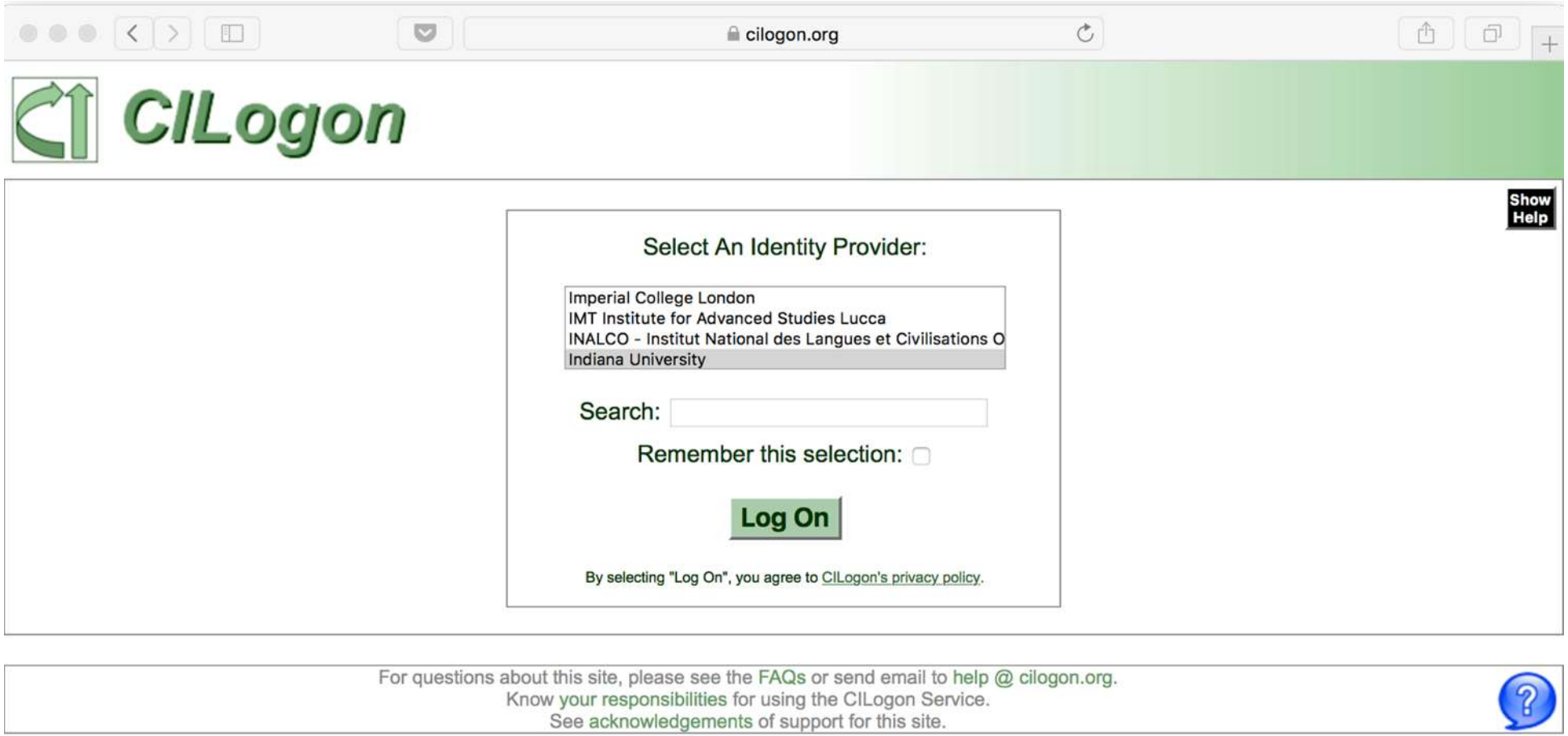
The Problem

- Gateways expose (shared) resources and APIs for manipulating those resources to a diverse set of users
- Authentication – another password to remember?
- Authorization – granting access without too much administrative burden
- Multiple gateways, multiple gateway clients
  - SciGaP (Science Gateway Platform as a Service)
- Campus gateways

Authentication

- Many options here but easiest thing is use an account the user already has
- CILogon
  - A service that allows users to authenticate to their institutions in order to retrieve credentials for accessing cyberinfrastructure.
  - Also provides a **OpenID Connect** (OIDC) interface to this federated authentication
- OpenID Connect
  - Built on top of OAuth2
  - Allows user to sign in with their identity provider
  - User grants client the right to access some of the user's information

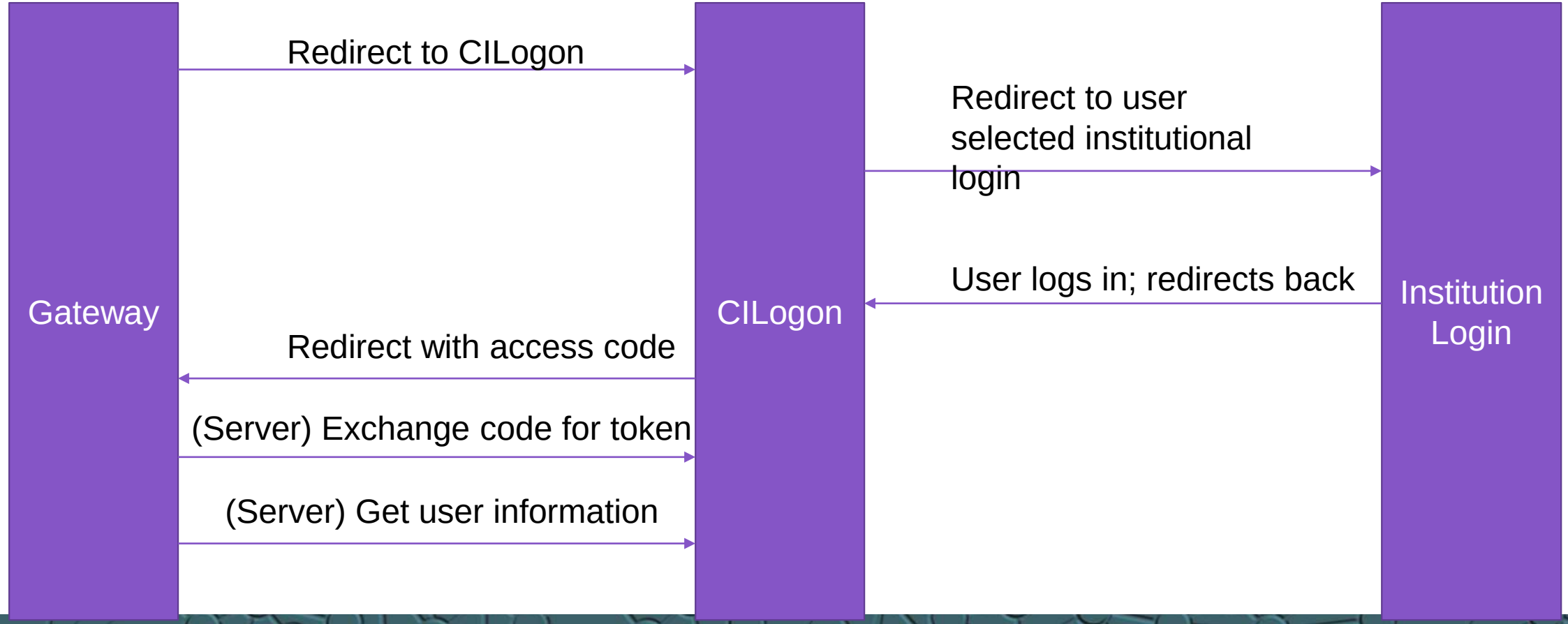
CILogon OpenID Connect



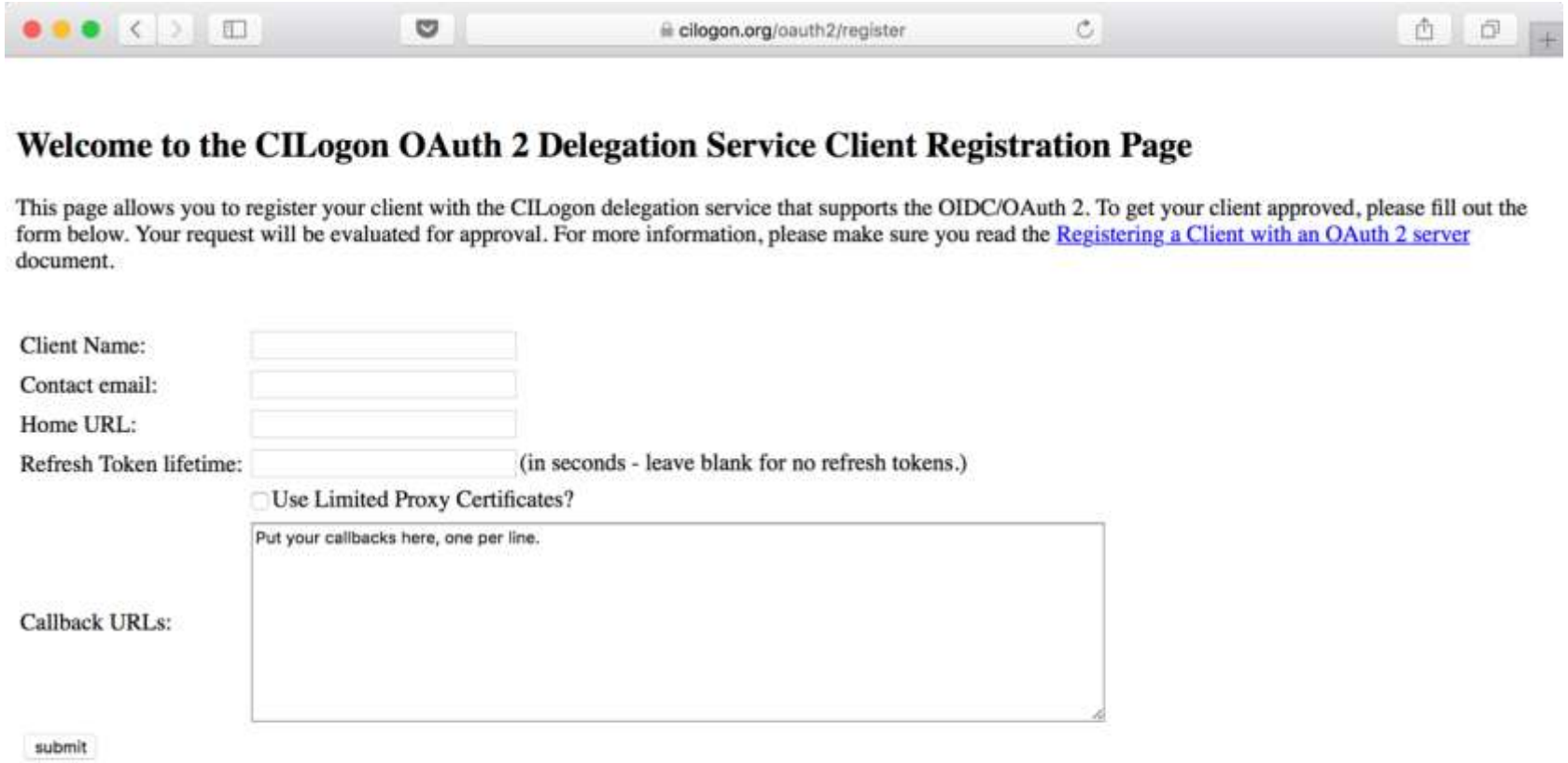
OpenID Connect

- One time initial setup
  - Register a client: client id and client secret
- 1 - Redirect to CILogon
  - Parameters: client id, redirect_uri, scope=openid
- 2 - User signs in
- 3 - Redirect back to gateway
  - Exchange access code for authorization token
- 4 - Get user info
  - Use authorization token to call user info endpoint

OpenID Connect flow



One time setup: Registering a client with CILogon



Step 1: OIDC Redirection

```
https://cilogon.org/authorize?response_type=code
  &client_id=CLIENT_ID
  &redirect_uri=https://mygateway.org/callback
  &scope=openid
```

Step 3: OIDC Exchange code for token

```
https://mygateway.org/callback?
  code=https%3A%2F%2Fcilogon.org%2Foauth2%2FauthzGrant%2F331685b3f...
```

```
POST /oauth2/token HTTP/1.1
Host: cilogon.org
Authorization: Basic Base64($client_id + ":" + $client_secret)

grant_type=authorization_code&code=...&redirect_uri=...
```

```
{
  "access_token": "eyJhbGciOiJ...",
  ...
}
```

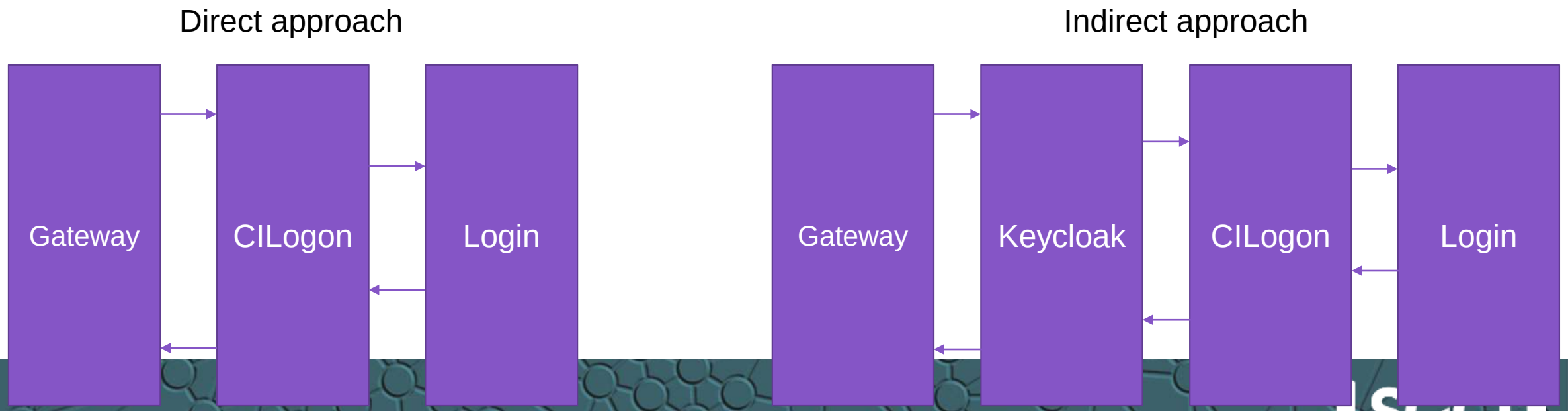
Step 4: OIDC fetch userinfo using access token

```
GET /oauth2/userinfo HTTP/1.1
Host: cilogon.org
Authorization: Bearer $access_token
```

```
{
"sub":"http://cilogon.org/serverA/users/123",
"name":”Marcus Christie",
"given_name":”Marcus",
"family_name":”Christie",
"email":”machrist@iu.edu”,
...
}
```

Integrating gateway with CILogon

- Direct integration
- Indirect integration through an Identity and Access Management (IAM) Server (for example, Keycloak)



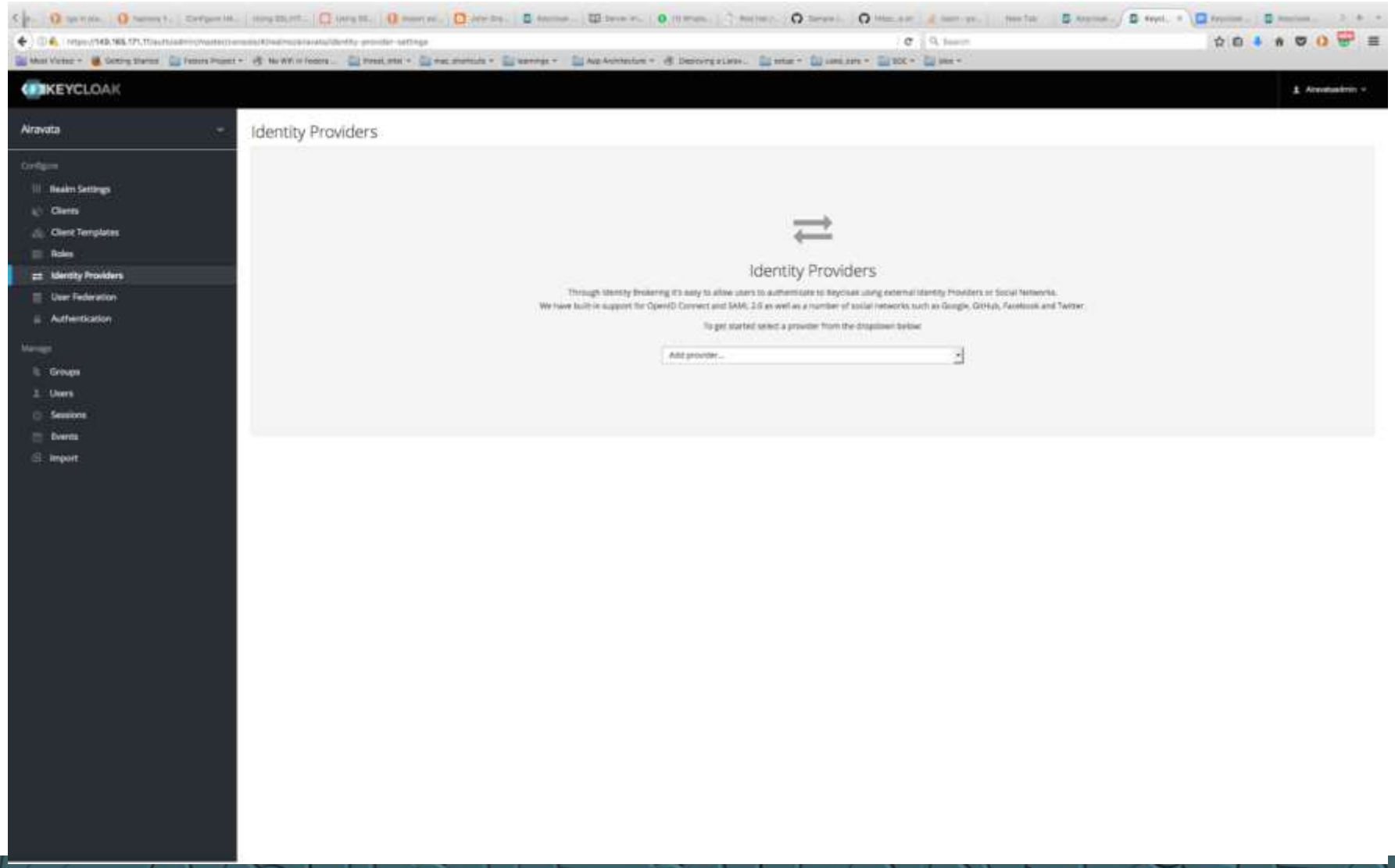
Advantages of using IAM server

- Flexible Authentication options
- Gateway client software integration is done once
- Authorization services

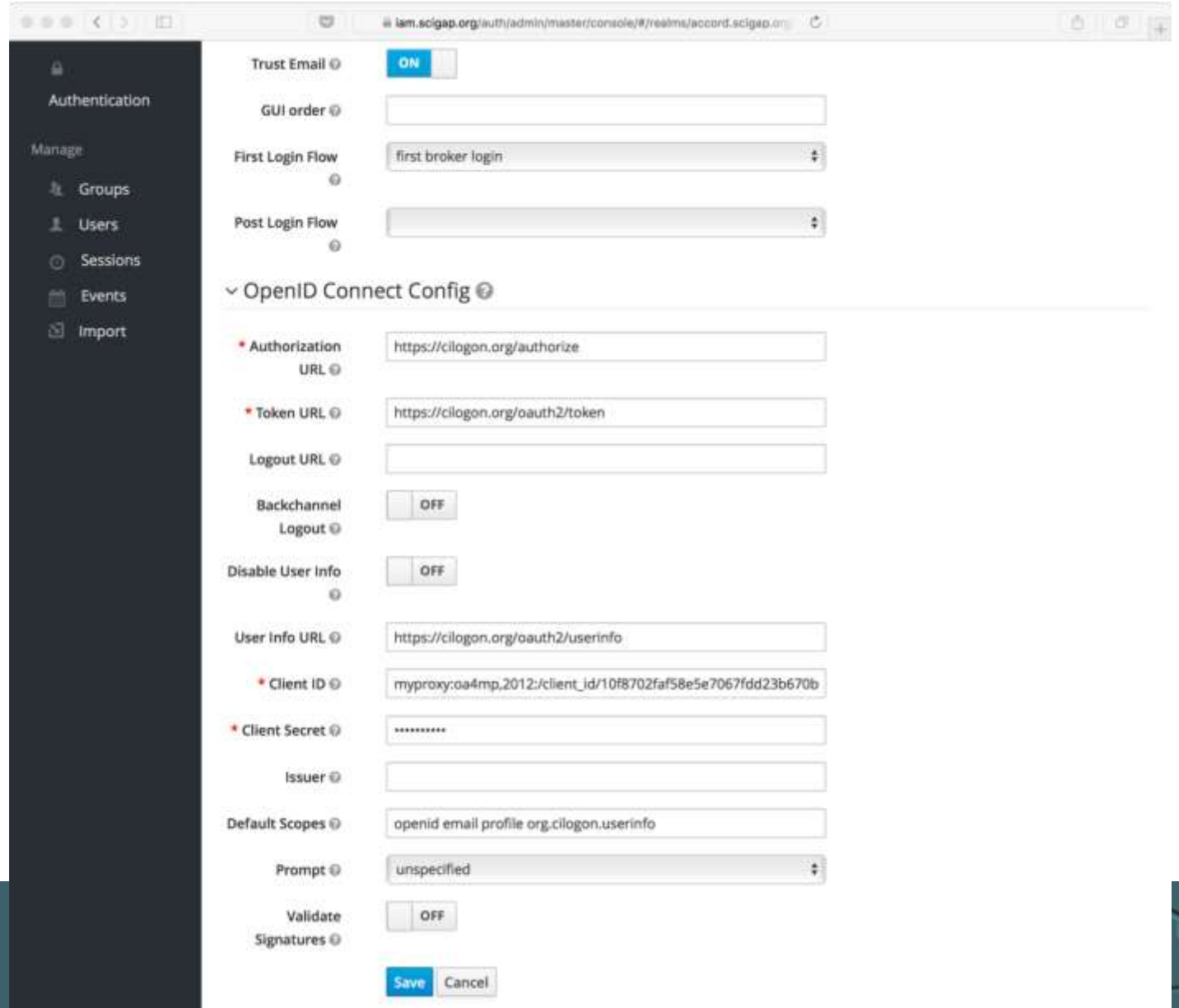
Keycloak

- http://www.keycloak.org/
- Red Hat, 200+ contributors on GH, 4,500+ merged PRs
- Open source (ALv2) identity and access management server
- Identity brokering; pluggable authentication mechanisms
- Authorization services: attribute-based, role-based, group-based, etc.
- Multi-tenanted
- Complete, well-documented REST API

Configuring Keycloak for CILogon



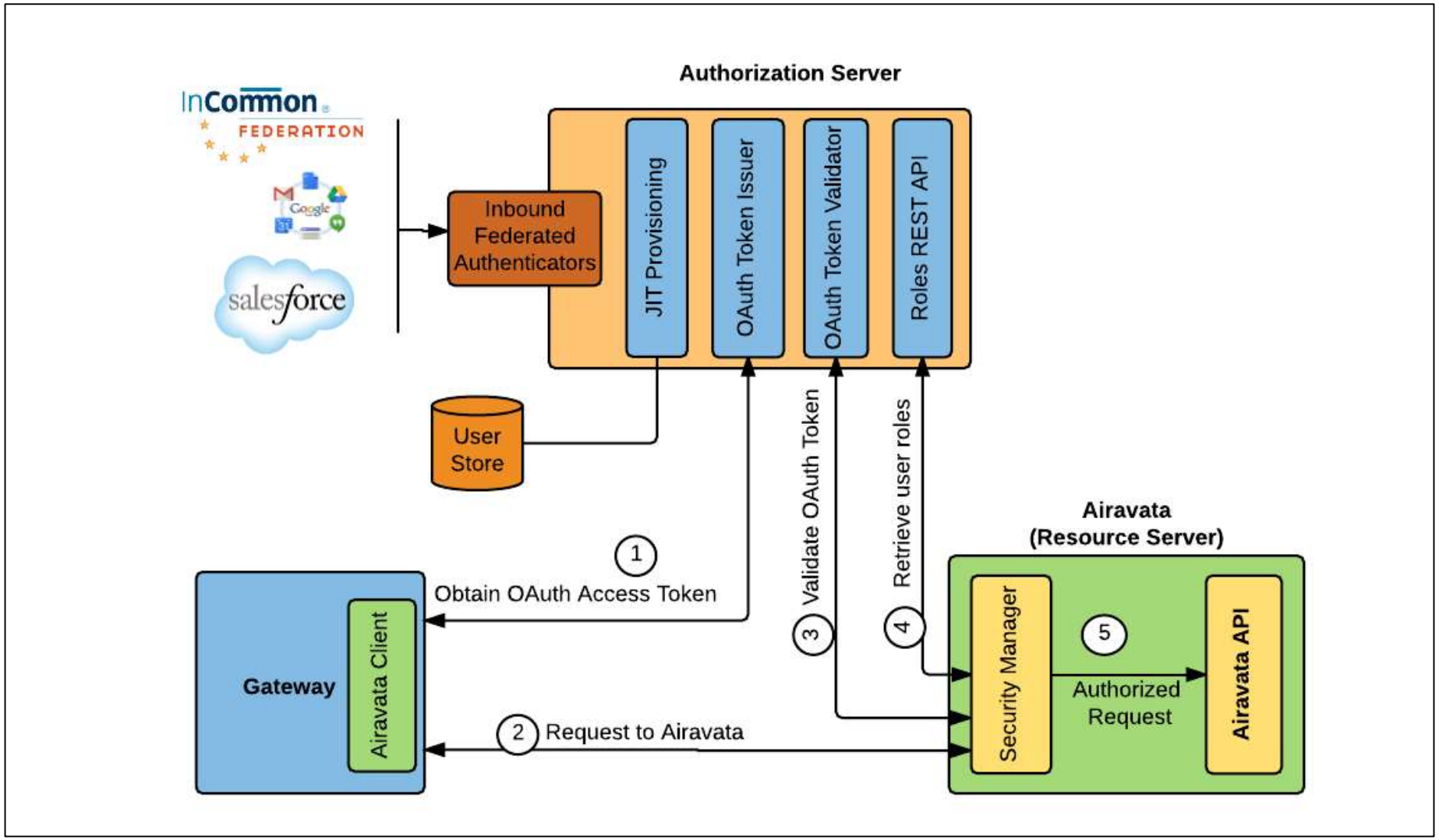
Configuring Keycloak for CILogon



How to integrate Keycloak

- Multi-tenanted: create a “realm” for each gateway
  - configure authentication options, like CILogon
- Web portal to use OpenID Connect for authentication
  - Redirect user to Keycloak to log in
- REST API for administrative and authorization integration
  - http://www.keycloak.org/docs-api/3.0/rest-api/index.html
  - E.g., Manage what roles a user has

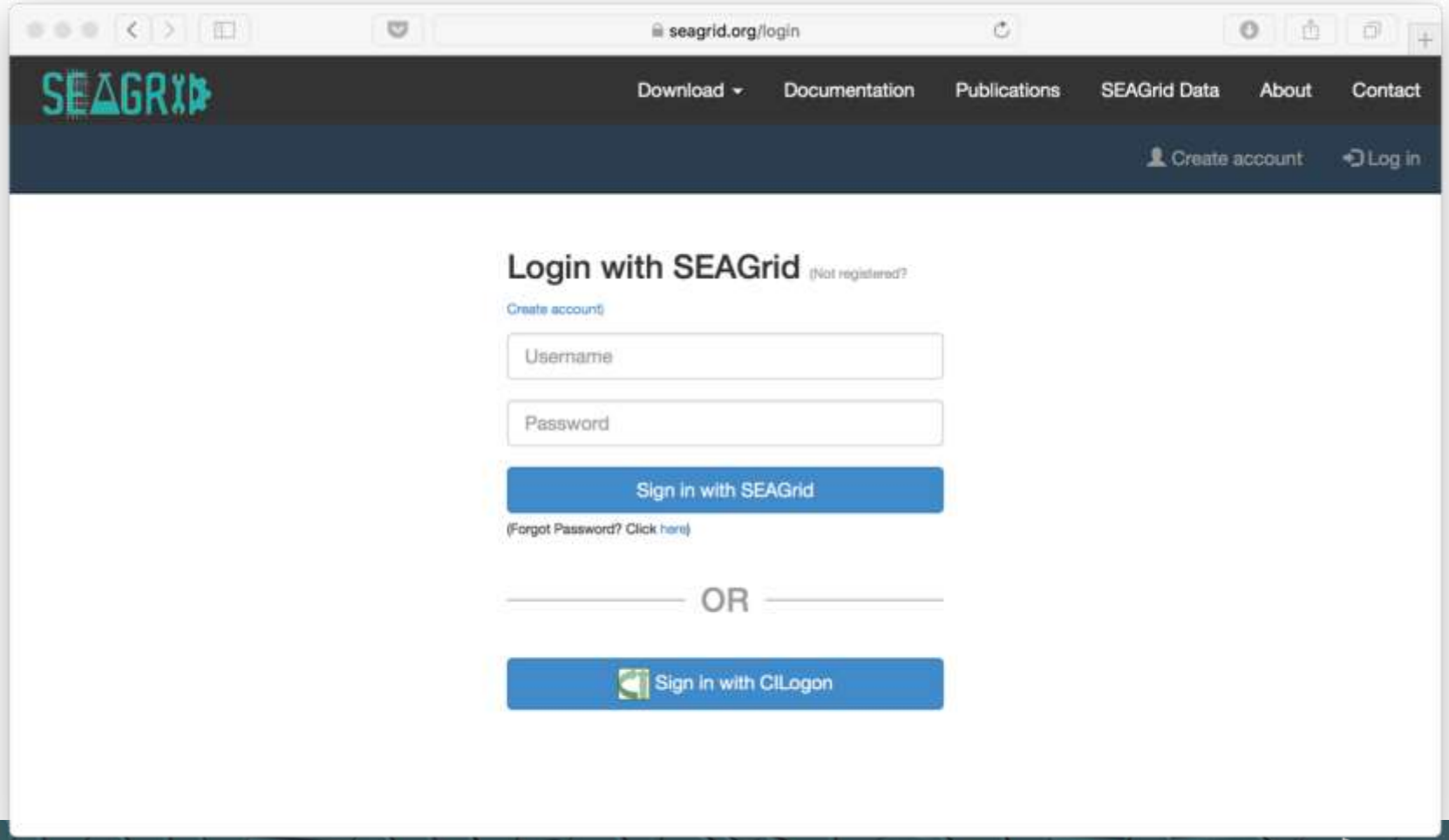
Securing Airavata



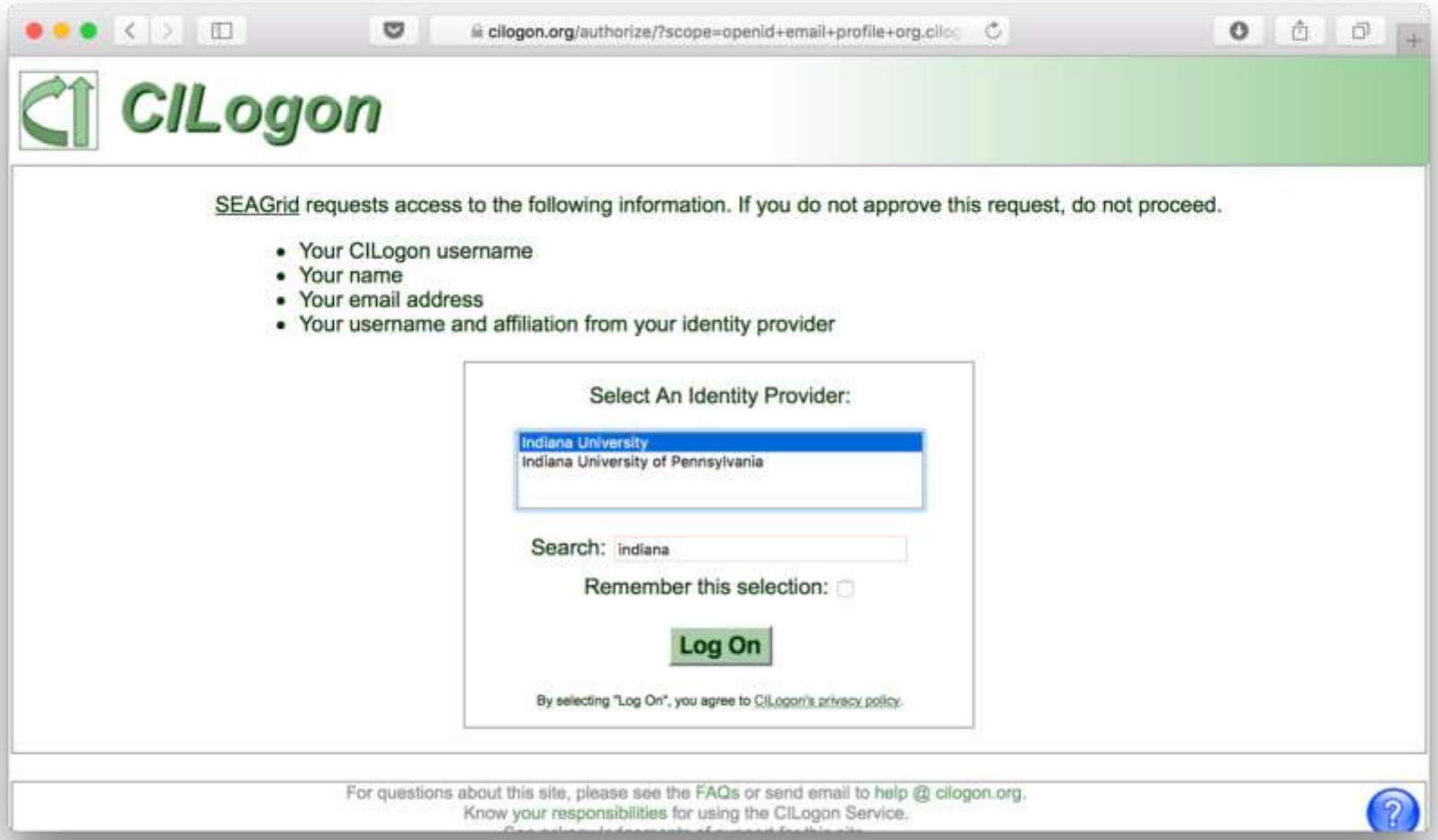
Apache Airavata – Profile Service

- Integrates with Keycloak REST API for tenant and user management
- Tenant Management
  - createTenant
    - Keycloak REST API: create Realm, create Realm Roles, create Admin user for Realm, create default client for web application
- User Management
  - createUser – Keycloak REST API: create User
  - enableUser – Keycloak REST API: update User
  - resetUserPassword – Keycloak REST API: update User with updated password
  - addRoleToUser – Keycloak REST API: add Realm Roles to User

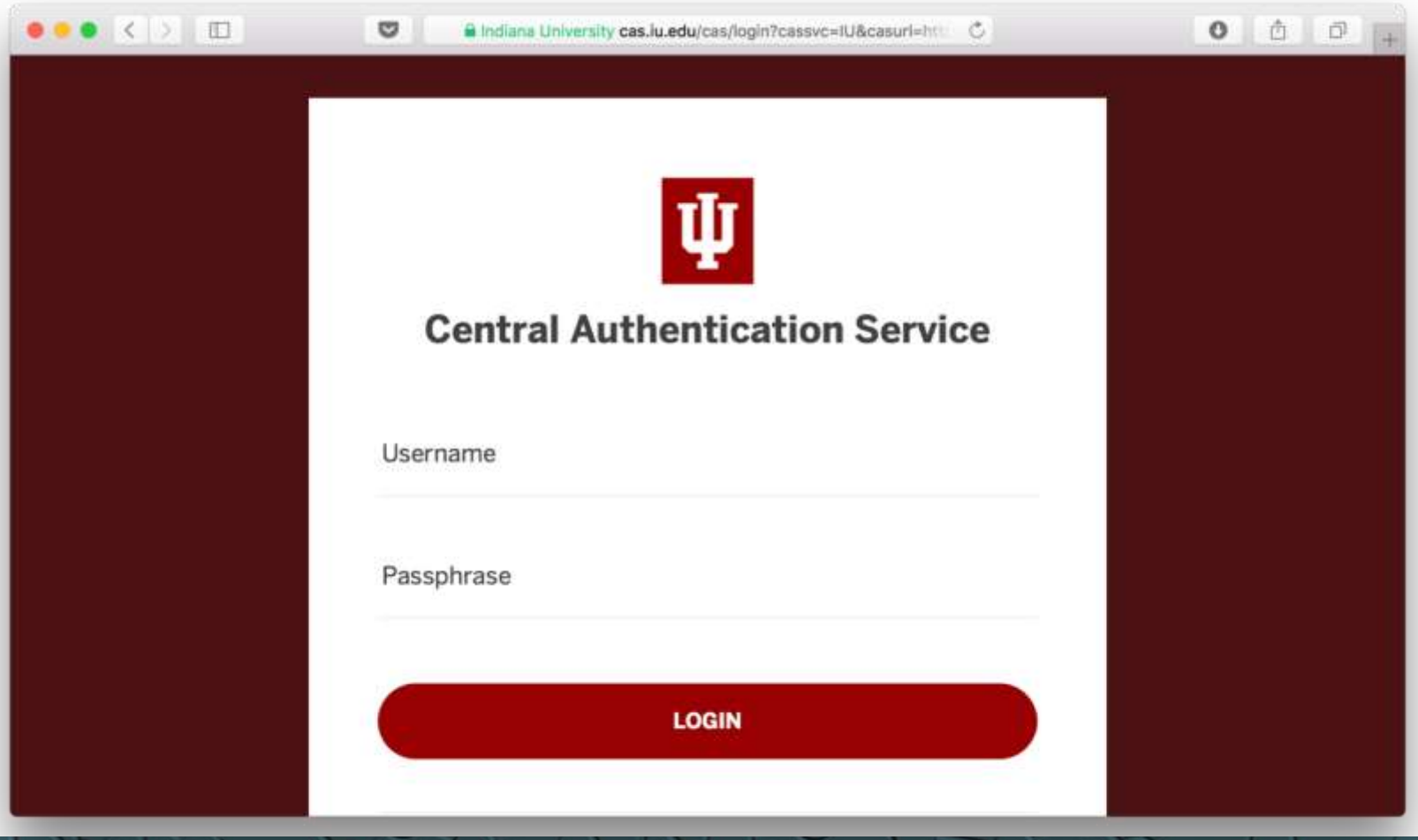
User logs in


seagrid.org/login
Download
Documentation
Publications
SEAGrid Data
About
Contact
Create account
Log in
Login with SEAGrid
(Not registered?
Create account)
Username
Password
Sign in with SEAGrid
(Forgot Password? Click here)
OR
Sign in with CILogon

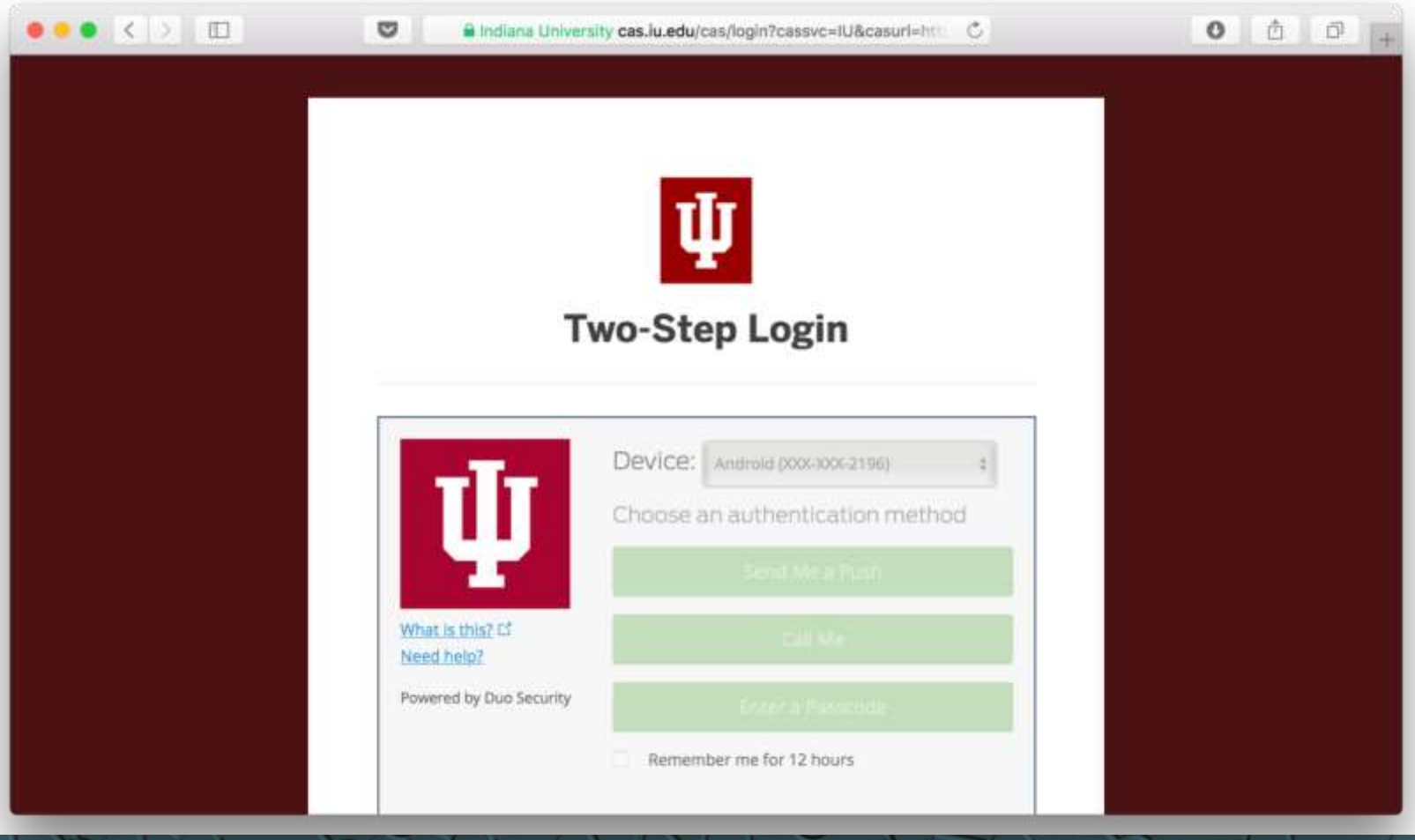
User logs in


cilogon.org/authorize/?scope=openid+email+profile+org.cilo
CILogon
SEAGrid 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:
Indiana University
Indiana University of Pennsylvania
Search: indiana
Remember this selection:
Log On
By selecting "Log On", you agree to CILogon's privacy policy.
For questions about this site, please see the FAQs or send email to help @ cilogon.org.
Know your responsibilities for using the CILogon Service.

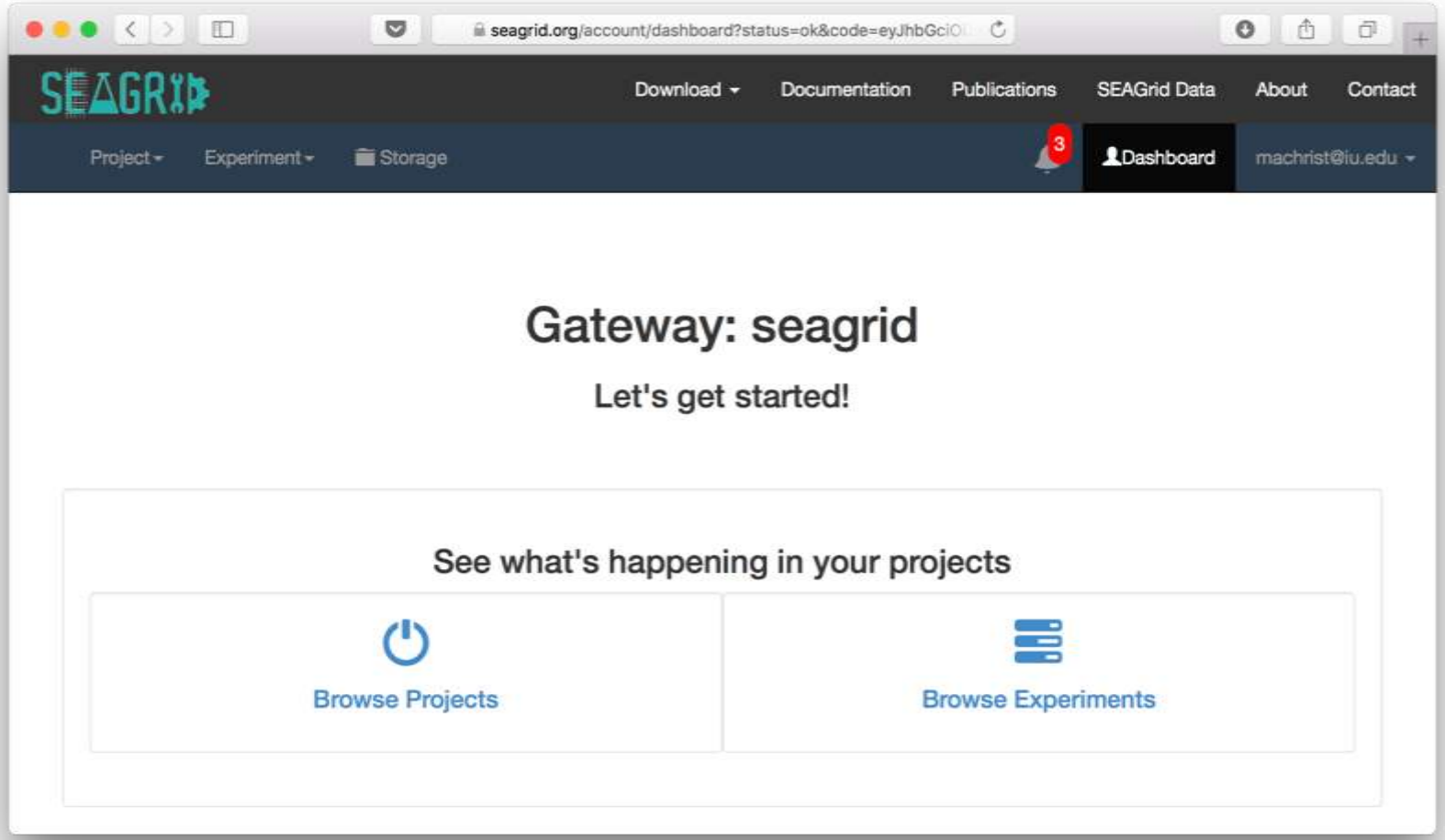
User logs in


Indiana University cas.iu.edu/cas/login?cassvc=IU&casurl=
Central Authentication Service
Username
Passphrase
LOGIN

User logs in


Indiana University cas.iu.edu/cas/login?cassvc=IU&casurl=
Two-Step Login
Device:
Android (XXX-XXX-2196)
Choose an authentication method
What is this?
Need help?
Powered by Duo Security
Remember me for 12 hours

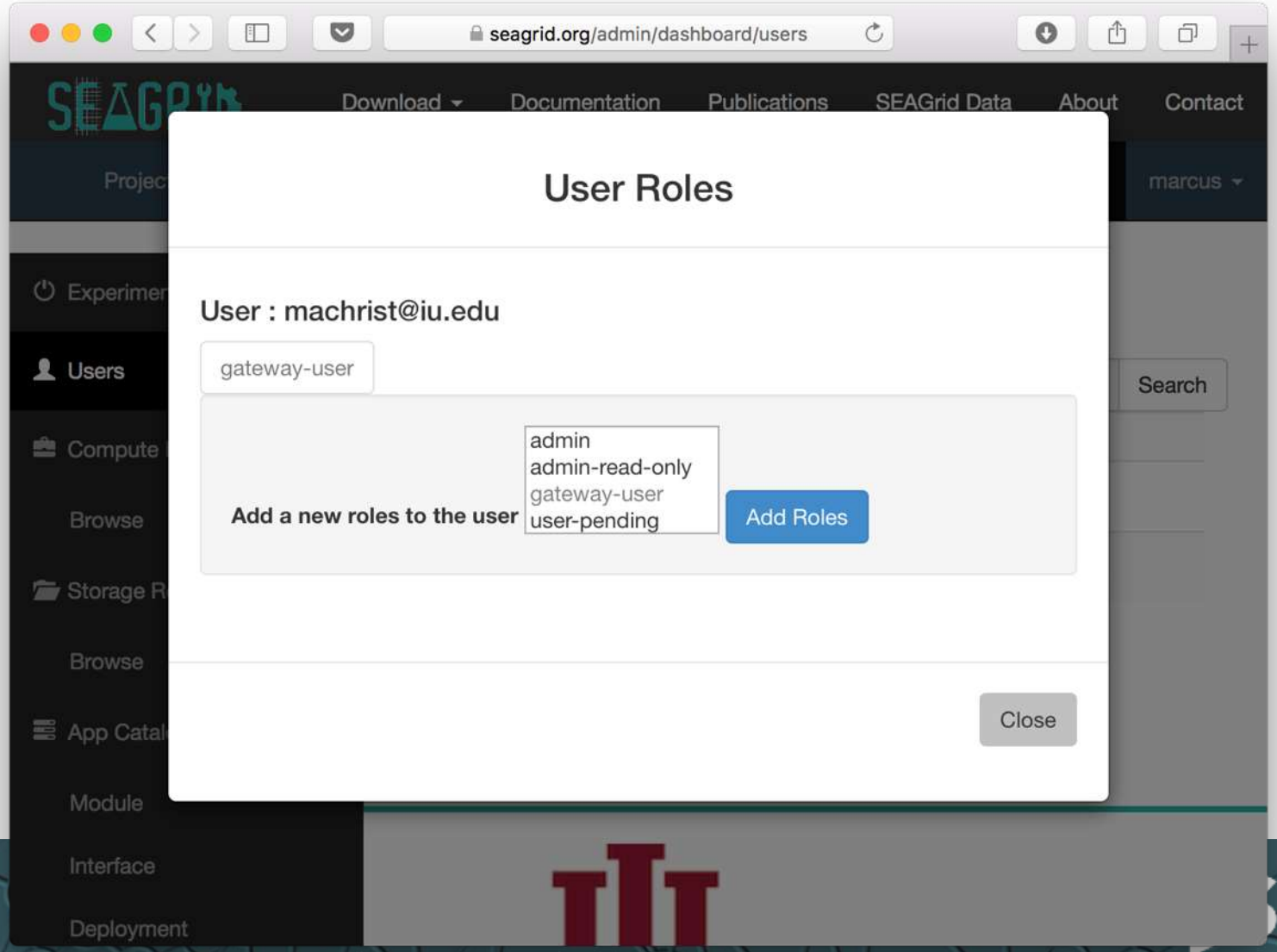
User logs in


seagrid.org/account/dashboard?status=ok&code=eyJhbGci
SEAGRID
Download
Documentation
Publications
SEAGrid Data
About
Contact
Project
Experiment
Storage
3
Dashboard
machrist@iu.edu
Gateway: seagrid
Let's get started!
See what's happening in your projects
Browse Projects
Browse Experiments

First login: user is assigned a default role

- At this point, the user is logged in but can’t do anything yet
- User is assigned to ‘user-pending’ role
- Gateway admin gets an email about a new gateway user

Admin grants role



How does the role get used?

- Portal – show/hide functionality based on role
- Airavata API server – each API method is restricted to certain roles
  - Uses OIDC endpoints to verify access token
  - Uses Keycloak REST API to get user's roles
  - Mapping from role to API methods accessible by that role

Questions?

- Feel free also to email me at machrist@iu.edu