Federated Identity Management for Research Organizations

2017 AGU New Orleans
Your Hosts Today

- Rachana Ananthakrishnan
  - Globus Project (www.globus.org)
  - University of Chicago
  - Argonne National Lab
Your Hosts Today

- Jim Basney
  - University of Illinois
  - National Center for Supercomputing Applications (NCSA)
  - CIlogon (www.cilogon.org)
  - Center for Trustworthy Scientific Cyberinfrastructure (trustedci.org)
Your Hosts Today

- Scott Koranda
  - University of Wisconsin-Milwaukee (UWM)
  - Laser Interferometer Gravitational-wave Observatory (LIGO)
  - Spherical Cow Group (SCG)
  - Center for Trustworthy Scientific Cyberinfrastructure (trustedci.org)
<table>
<thead>
<tr>
<th>Time</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00</td>
<td>Preface and Introduction</td>
</tr>
<tr>
<td>09:10</td>
<td>Research Identity Management Process Needs</td>
</tr>
<tr>
<td>09:30</td>
<td>Federated Identity for Authentication: SAML</td>
</tr>
<tr>
<td>10:00</td>
<td>Federated Identity for Authentication: OIDC</td>
</tr>
<tr>
<td>10:30</td>
<td>Break</td>
</tr>
<tr>
<td>10:45</td>
<td>Federated Identity for Authentication: Globus Auth</td>
</tr>
<tr>
<td>11:15</td>
<td>Hands-On Exercises</td>
</tr>
<tr>
<td>12:30</td>
<td>Lunch</td>
</tr>
<tr>
<td>Time</td>
<td>Topic</td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>14:00</td>
<td>Federation Topics: SAML Federations, IdP Discovery, Joining InCommon, Attribute Release, Unaffiliated IdPs</td>
</tr>
<tr>
<td>15:00</td>
<td>Non-Browser Clients and Federated Identity</td>
</tr>
<tr>
<td>15:10</td>
<td>Participant Lifecycle Management</td>
</tr>
<tr>
<td>15:30</td>
<td>Break</td>
</tr>
<tr>
<td>15:45</td>
<td>Hands-On Exercises</td>
</tr>
<tr>
<td>16:45</td>
<td>Wrap Up</td>
</tr>
<tr>
<td>17:00</td>
<td>End</td>
</tr>
</tbody>
</table>
Workshop Materials

● Slides available at https://registry.vo.idm.training/slides

● Handout with your training VM host and credentials

● Helpful to have today:
  ○ Google account
  ○ Browser you can use and "clear all history" repeatedly
    ■ Preferably Firefox
  ○ SSH key and SSH client to use with it
    ■ RSA or DSA key
Research Identity Management
Process Needs
What Is A Collaboration?

- A cross-organizational collection of people who come together for a particular interest
- Virtual Organizations (VOs) are one type of Collaboration
  - Here the terms can be used interchangeably
Characteristics of a Collaboration

- Participants from many institutions
  - Some with federated identities
  - Some with social identities (Google, etc)
  - Some with no identities
- One institution often hosts collaboration infrastructure
What A Collaboration Needs

● Common Tools
  ○ Wikis / Document Repositories
  ○ Mailing Lists
  ○ Calendaring / Scheduling

● Domain Specific Tools
  ○ Application portals
  ○ Analysis workflow tools
  ○ Data viewers
Types of Collaborations

- Large pseudo-enterprises (e.g., big science) with a long time frame
- Researchers from a dozen schools working together on a fixed-duration project
- Members of the community partnering with a University to provide a service
- A handful of people from different departments within the same institution
The Long Tail of VOs

![Graph showing the long tail of VOs with points for LIGO and Dr. Lee's Research Group.](image-url)
Collaboration Identity Management Processes

- **Onboarding**
  - **Invitation**
    - Collaboration initiated, participant accepts
  - **Self Signup**
    - Participant initiated
  - **Application**
    - Participant initiated, approval required
  - **Conscription**
    - Collaboration initiated
Collaboration Identity Management Processes

- Offboarding
  - Expiration (Friendly, Involuntary)
    - Based on term attached at enrollment (e.g., 1 year funded position)
    - Grace period may apply before deprovisioning
  - Termination (Unfriendly, Involuntary)
    - Collaboration initiated
    - Deprovisioning usually immediate
Collaboration Identity Management Processes

- Offboarding
  - Resignation (Friendly, Voluntary)
    - Participant initiated
    - Grace period may apply before deprovisioning
  - Retirement (Friendly, Voluntary)
    - Participant initiated
    - Grace period may apply before deprovisioning
    - Some services may remain available
Collaboration Identity Management Processes

- Offboarding
  - Leave of Absence (Friendly, Voluntary)
    - Participant initiated
    - May or may not be desirable to maintain access to services
  - Desertion (Unfriendly, Voluntary)
    - Treated as Termination or Retirement
Collaboration Identity Management Processes

- Lifecycle Management
  - Transfers
  - Account Linking
  - Attribute Updates
  - Group Management
- Overall status may be calculated from more than one role
Collaboration Identity Management Processes

- Identity Infrastructure
  - Onboarding / Offboarding / Lifecycle Management
    - Transfers and Account Linking
  - Group Management

- Attributes to drive authorizations
  - From organizational sources, but hard to get
    - R&S? But even 10 attributes is not enough

- Collaboration attributes
The Long Tail of VOs

Quantity

Business Process Complexity

LIGO

Dr Lee's Research Group
Regardless of size, the basic IdM need remains the same:

1. Grant access to services when needed
2. Remove access when done
Identity Management Infrastructure

- Authentication / Credentialing Services
  - External: Federated, Social
    - Proxies, Gateways, Direct Integration
    - Discovery Services
  - Internal: Kerberos, LDAP
  - User-Centric: SSH Keys, Certificates
  - Multi-factor: RSA, Google Auth, Duo, U2F
Identity Management Infrastructure

- **Lifecycle/Access Management and Authorization**
  - Person Registry
    - Identity Matching / Linking
  - Group Registry
- **Provisioning and Application Integration**
  - Directory Services
  - Messaging Services
Federated Identity for Authentication
Federated Identity Definition

"...the means of linking a person's electronic identity and attributes, stored across multiple distinct identity management systems."

"...ultimate goal of identity federation is to enable users of one domain to securely access data or systems of another domain seamlessly, and without the need for completely redundant user administration."

Welcome to the REFEDS Wiki

The aim of REFEDS (Research and Education FEDerations) is to exchange Identity Federation processes, practices and policies and to discuss ways to facilitate inter-federation work. REFEDS aims at collecting, disseminating and (when possible) harmonising the procedures and policies followed by the participating federations.
Select your identity provider

English | Bokmål | Nynorsk | Sámi | Dansk | Deutsch | Español | Svenska | Suomeksi | Français | Italiano | Nederlands | Lëtzebuergesch | Čeština | Slovenščina | Hrvatski | Magyar | Język polski | Português | Português brasileiro | Türkçe | 日本語 | 紫藤中文 | Ελληνικά | Lietuvių kalba | Åarjelh-saemien giele | русский язык

![Stained glass window](https://login.terena.org/wayf/module.php/disc)

<table>
<thead>
<tr>
<th>All</th>
<th>eduGAIN</th>
<th>ÆDEURus</th>
<th>Chile</th>
<th>Nordic countries</th>
<th>Spain</th>
<th>UKfederation</th>
<th>US</th>
<th>Italy</th>
<th>NZ</th>
<th>AU</th>
<th>NL</th>
<th>SAFIRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Social networks</td>
<td>Guest providers</td>
<td>Miscellaneous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A. T. Still University
AAF Virtual Home
AAI@EduHR Single Sign-On Service
Aalborg University
Aalto University
Aarhus Basic Health Care College
Aarhus School of Marine and Technical Engineering

Incremental search...
What just happened?

- Used identity from one security domain (home organization) to access resources from another security domain (REFEDs)
- Pre-established trust between the identity provider (home organization) and the service provider (REFEDs wiki)
- Leveraged SAML protocol
  - Security Assertion Markup Language
  - Facilitates federated web browser single sign-on (SSO)
  - Most common (today) protocol in higher ed and research
No Federated Identity?

Cannot find your home organization?

- Use a social identity (Google, Facebook, Twitter...)
- Or sign up for a (free) federated identity
- Try NCSA: [https://go.ncsa.illinois.edu/idp-guest](https://go.ncsa.illinois.edu/idp-guest)
- Or try United ID: [https://unitedid.org/](https://unitedid.org/)
  - Requires a second factor like Google Authenticator on your phone
SAML
SAML tracer
by Olav Morken, Jaime Perez

Debug and view SAML messages

Add to Firefox

Rate your experience

How are you enjoying your experience with SAML tracer?

Log in to rate this extension

Screenshots
Trace Your SAML SSO Flow

- SAML tracer Add-on for FireFox
- SAML DevTools extension for Chrome is also available
- Other tools useable but involve more work
  - LiveHTTPHeaders
  - Safari Web Inspector
  - Fiddler
  - Often combined with [https://www.samltool.com/]
<saml2p:Response Destination="https://spaces.internet2.edu/Shibboleth.sso/SAML2/POST"
    ID="a7a2a544ba5f52cc9b3f3e58e485f29fde"
    InResponseTo="3c59c65a2a42290ba8200dd31d482b29a"
    IssueInstant="2016-06-12T16:27:36.448Z"
    Version="2.0"
    xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol">
        ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
            <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
            <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
            <ds:Reference URI="#a7a2a544ba5f52cc9b3f3e58e485f29fde">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
                </ds:Transforms>
                <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha256" />
                <ds:DigestValue>rNZmtA7hjFiAyjROXhO7m3er7N/Pz6Xe7kJpQeUomYcE="</ds:DigestValue>
            </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>
gNxxc6R9oqGChqFDhbtFLxVb3Ms6VWdBqGqFRlnLhdbayrTQ6WhQTaSAwk6R1CnFpFyEx+jT
SPcVr7O7Fg+H8PM0/9LvRjRqfoDgXeLF49FPr19BQ42186LAM7F1c5WKL4B9C+3iiKh0CB5dDpY
Qd1t1kkplrLy2F3E2f3t3Y7k387M2EBGhQDmD9JMB9xkxWk)V
</ds:SignatureValue>
</saml2:Signature>
</saml2:Issuer>
</saml2p:Response>
SAML SP Authentication Request

```
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    AssertionConsumerServiceURL="https://spaces.internet2.edu/Shibboleth.sso/SAML2/POST"
    Destination="https://idp.uwm.edu/idp/profile/SAML2/Redirect/SSO"
    ID="_3c59c65a242980ba8200dd31d48b2b9a"
    IssueInstant="2016-08-12T16:27:34Z"
    ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Version="2.0"
    >
    <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
        https://spaces.internet2.edu/shibboleth</saml:Issuer>
    <samlp:NameIDPolicy AllowCreate="1" />
</samlp:AuthnRequest>
```

- **SAML entityID**
- **Every SP and IdP has unique entityID**
- **Best practice is URL syntax**
- **Older practice is URN**
SAML SP Authentication Request

```xml
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    AssertionConsumerServiceURL="https://spaces.internet2.edu/Shibboleth.sso/SAML2/POST"
    Destination="https://idp.uwm.edu/idp/profile/SAML2/Redirect/SSO"
    ID="_3c59c65a242980ba8200dd31d48b2b9a"
    IssueInstant="2016-08-12T16:27:34Z"
    ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Version="2.0"
>  
    <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
        https://spaces.internet2.edu/shibboleth
    </saml:Issuer>
    <samlp:NameIDPolicy AllowCreate="1"/>
</samlp:AuthnRequest>
```

- Timestamp
- Prevent replay attacks
- Most systems tolerate some clock skew
SAML IdP Response

- IdP uses HTTP-POST binding to send response to SP
  - Base64 encoded XML payload returned to browser
  - Browser does the POST
- Most IdPs include Javascript to automate the POST
  - Turn off Javascript and you will see a button to click to force the POST
SAML IdP Response

- Response is usually digitally signed (XML digital signature)
  - SP can verify and trust the response
  - Prevent tampering
- Includes an assertion about the authentication event
  - Assertion usually encrypted (XML encryption)
  - Encrypted using the SPs SAML key
  - Hides details about user from snooping browsers
  - TLS transport not usually required but usually used
SAML2 Subject NameID vs Subject Attributes

- SAML2 spec defines a limited set of NameID formats
- Not as flexible as using attributes about subject
- SAML2 NameID much less used by research organizations
  - Most often seen with commercial vendor services or SPs
    - WebEx, Google Apps, electronic journals, ...
  - Due to limitations of their SAML implementations
SAML AttributeStatement

<saml2:AttributeStatement>
  <saml2:Attribute FriendlyName="sn" Name="urn:oid:2.5.4.4" NameFormat=SNIP>
    <saml2:AttributeValue>User01</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute FriendlyName="displayName" Name="urn:oid:2.16.840.1.113730.3.1.241" NameFormat=SNIP>
    <saml2:AttributeValue>Test User01</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute FriendlyName="givenName" Name="urn:oid:2.5.4.42" NameFormat=SNIP>
    <saml2:AttributeValue>Test</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute FriendlyName="eduPersonPrincipalName" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" NameFormat=SNIP>
    <saml2:AttributeValue>testuser01@vo.idm.training</saml2:AttributeValue>
  </saml2:Attribute>
  <saml2:Attribute FriendlyName="mail" Name="urn:oid:0.9.2342.19200300.100.1.3" NameFormat=SNIP>
    <saml2:AttributeValue>testuser01@vo.idm.training</saml2:AttributeValue>
  </saml2:Attribute>
</saml2:AttributeStatement>
SAML2 Attributes

- Higher ed and research leverage well defined standards
  - eduPerson schema managed by MACE-DIR
  - Standard LDAP schema like sn, givenName, mail
    - Borrow OIDs rather than reinvent new
- More later on attributes in higher ed and research federations like InCommon
Deploying a SAML Service Provider
SAML Capable Applications and Services

Writing a SAML consumer is hard

- Don't do it
- Even the best SAML libraries have significant limitations
  - Often why commercial vendor implementations so limited
SAML Application Integration

Better is to integrate with *existing SAML consumers*:

Web applications that consume SAML and provision details about subject and authentication to applications
SAML Application Integration

Quite strong options from higher ed and research community:

- Shibboleth Native SP for Apache (Linux and Windows)
- Shibboleth Native SP for IIS (Windows)
- SimpleSAMLphp (any environment PHP can run in)

Today focus on Shibboleth Native SP for Apache on Linux
SAML Application Integration

Other options outside of higher education and research:
- Cloud services like Auth0, OneLogin
- Microsoft AD FS

The primary issue with these solutions is that they focus on bi-lateral federations (small "f") between a single IdP and perhaps a few SPs
SAML Application Integration

You lose much of what you can gain from Federations

- (large "F") like InCommon

Some organizations leverage AD FS but life is hard for them...

(Not the same as using Microsoft AD as person registry, which is fairly common across higher education and research)
Shibboleth SP Deployment

- Shibboleth project officially supports RHEL and CentOS with yum repositories
- No official support for Debian and Ubuntu
  - Do not use the Debian/Ubuntu standard repository versions--too old
  - Recommend the SWITCH Federation Repositories
    - No official support outside of SWITCH but you are welcome to use repositories
    - https://www.switch.ch/aai/guides/sp/installation/
Shibboleth SP Architecture

Shibboleth SP includes two primary components
1. Shibboleth daemon 'shibd'
2. Shibboleth Apache module 'mod_shib'
Shibboleth SP Architecture

1. Shibboleth daemon 'shibd'
   ○ Most of the SAML "heavy lifting"
   ○ Maintains session state
   ○ Listens on UNIX socket
   ○ Independent daemon managed with init script
Shibboleth SP Architecture

2. Shibboleth Apache module 'mod_shib'
   - Runs "inside" of Apache like any other Apache module
   - Exposes the SAML URL endpoints for consuming SAML
   - Includes semantics for authorization and access control
   - Communicates with shibd via the UNIX socket
Shibboleth SP Application Integration

The shibd configuration

```xml
<ApplicationDefaults entityID="https://sp.example.org/shibboleth"
    REMOTE_USER="eppn persistent-id targeted-id">
```

tells the SP which attributes from the IdP to search for in order and then populate into REMOTE_USER CGI environment variable
Shibboleth SP Attribute Mapping

- All attributes asserted by IdP can be made available to applications in CGI environment variables
- Attributes mapped from formal "on the wire" names
  
  "urn:oid:1.3.6.1.4.1.5923.1.1.1.6" mapped to "eppn"

- attribute-map.xml controls the mapping
Shibboleth SP Apache Module Access Control

Access control may be configured with Apache

General syntax:

```require rule-type value1 value2```

Some rules support a regular expression mode:

```
require rule ~ exp1 exp2
```
Test Shibboleth SP Apache Module Access Control

Allow anybody that can authenticate (no authorization):

```
require shib-session
```

Require particular user:

```
require shib-attr eppn testuser0N@vo.idm.training
```
Shibboleth SP: Active Vs Passive Protection

"Active" protection of resources

- User must have a valid SP session to access resource
- Session may not be enough--some access control might also be in place
- SP will start SAML Web SSO flow immediately if no session
"Passive" protection is also possible

- No valid session required
- SP will not start SAML SSO flow
- If valid session asserted attributes will be available
- SAML SSO flow must be triggered by application
- Also called "lazy sessions"
Application Initiated SAML SSO Flow

To trigger SAML Web SSO flow and initiate a SP session, application redirects browser to special Shibboleth endpoint:

https://sp0N.vo.idm.training/Shibboleth.sso/Login
Lazy Sessions

- Lazy sessions is a common application integration pattern
- Often used with semi-public wikis
  - Anyone can read
  - Write access requires authentication (a valid SP session)
  - Wiki is configured to send browser to the
    /Shibboleth.SSO/Login endpoint when either user clicks
    "Login" or chooses "Edit"
- Access control is fully delegated to the application
Ending SP Session

- Usually TWO sessions
  a. Session with the IdP, effectively providing SSO
  b. Session with the SP

- Lifetimes of two sessions are *completely independent*
  a. Ending one has no bearing on ending the other
  b. Each may have its own lifetime AND inactivity timeout
Ending SP Session

"Global" logout in a Federated context

- Only when the same enterprise operates the IdP and the SP(s) can you realize the notion of "global" logout or single logout (SLO) that you might expect
- Example: why should an SP operated by LIGO be able to tell the University of Wisconsin-Madison IdP to end all sessions for the user?
SAML2 Logout

- Advanced topic not covered more here today
- Concern about SSO sessions from "public" terminals
  - Need to assess your risks per-service
  - Forced re-authentication can help (but not all IdPs will respect the request)
  - Educate your users -- if your users routinely use "public" terminals not managed by an organization you trust you probably have larger issues than SAML2 SLO
Shibboleth SP Local Logout

Terminate the Shibboleth SP session by browsing to

https://sp0N.vo.idm.training/Shibboleth.sso/Logout
OpenID Connect (OIDC)
OpenID Connect: Introduction

- Third gen OpenID (after OpenID 1.0 and OpenID 2.0)
- Adopted by Amazon, Google, Microsoft, and many others
- Auth layer on top of OAuth 2.0 authz framework (RFC 6749)
- Adds new token type: ID Token
- Adds new OAuth resource: UserInfo
- Standard claims in ID Token and UserInfo response
- Defines scope values for requesting claims
- Specifications: https://openid.net/connect/
SAML or OIDC?

Gateways mean you don't need to support both in each application

- OIDC to SAML (e.g., Cirrus)
- SAML to OIDC (e.g., CILogon)

Choose based on application/platform/language support
### SAML and OIDC: Terminology

<table>
<thead>
<tr>
<th>SAML</th>
<th>OIDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity Provider (IdP)</td>
<td>OpenID Provider (OP)</td>
</tr>
<tr>
<td>Service Provider (SP)</td>
<td>Relying Party (RP)</td>
</tr>
<tr>
<td>Attributes</td>
<td>Claims</td>
</tr>
<tr>
<td>Attribute Bundle</td>
<td>Scope</td>
</tr>
<tr>
<td>Authentication Assertion</td>
<td>ID Token</td>
</tr>
</tbody>
</table>
OpenID Connect: Standard Claims

- sub
- email
- email_verified
- phone_number
- phone_number_verified
- address

- name
- given_name
- family_name
- middle_name
- nickname
- preferred_username
- profile / picture
- website
- gender / birthdate
- zoneinfo / locale
- updated_at

**scope: openid**
- sub
- email

**scope: email**
- email_verified

**scope: phone**
- phone_number
- phone_number_verified

**scope: address**
- address

**scope: profile**
OpenID Connect: Protocol Overview

1. The RP (Client) sends a request to the OpenID Provider (OP).
2. The OP authenticates the End-User and obtains authorization.
3. The OP responds with an ID Token and usually an Access Token.
4. The RP can send a request with the Access Token to the UserInfo Endpoint.
5. The UserInfo Endpoint returns Claims about the End-User.

Source: https://openid.net/specs/openid-connect-core-1_0.html
Use Google OIDC

- Update SP to use Google OIDC instead of Shibboleth

```bash
cd /etc/httpd/conf.d
mv shib.conf shib.conf.template
ln -sf openidc-google.conf.template openidc.conf
service httpd restart
```

- Browse to https://sp0N.vo.idm.training/secure
Use Google OIDC

Background:

- We’ve already registered an OIDC client at 
  https://console.developers.google.com/apis/credentials

- Google OIDC documentation is at
  https://developers.google.com/identity/protocols/OpenIDConnect

- See also Google OAuth playground at
  https://developers.google.com/oauthplayground/
Demonstrate claim-based authorization using .htaccess

```bash
mkdir -p /var/www/html/secure/mygroup
echo "hello, world" > /var/www/html/secure/mygroup/hello.txt
echo -e "require claim email:skoranda@sphericalcowgroup.com" \
  > /var/www/html/secure/mygroup/.htaccess
```

Browse to https://sp01.vo.idm.training/secure/mygroup

Background:

https://github.com/pingidentity/mod_auth_openidc/wiki/Authorization
OpenID Connect: Protocol Deep Dive

What did mod_auth_openidc do?

Let’s reproduce it using curl & the Google OAuth Playground.

- Initial browser-based AuthN/AuthZ request/response
- Get Access Token and ID Token
- Look inside ID Token
- Use Access Token to get UserInfo
Google OAuth Playground

- https://developers.google.com/oauthplayground/
Google OAuth Playground

Step 1: Select & authorize APIs

Step 2: Exchange authorization code for tokens

Once you got the Authorization Code from Step 1 click the Exchange authorization code for tokens button, you will get a refresh and an access token which is required to access OAuth protected resources.

Authorization code: 4/TcYVa3Vmgb270IC4NKcbm8J4MNw0HsZ...

Request / Response

HTTP/1.1 302 Found
Location: https://accounts.google.com/o/oauth2/v2/auth?

GET /oauthplayground/?code=4/TcYVa3Vmgb270IC4NKcbm8J4MN
Host: developers.google.com

Step 3: Configure request to API

Federated Identity Management For Research Organizations
2017 AGU New Orleans
Google OAuth Playground

Federated Identity
2017 AGU New Orleans
Google OIDC with Curl

- Copy and paste id_token at https://jwt.io
eyJhbGciOiJSUzI1NiIsImtpZCI6IjVyYmE5NDFiODM0MWNjMWM5NTk2NmI3MDgyMDk2YWl5ODc3NDc3OTgFQ.CjJpc3MiM1oJodHRwczovL2FjY291bnRzLmdvb2dsZS5jbi20iLCJhdF9oYXNoIjoiZ3d3LnB1c3NlcnYeU1NHRnZnY0QFCXoJc29ja1Z1IjZmYzY1MzY0MGE1NzU2N2I2NDQyNDUyNzg0Yi18Z2FmblZ1MjBzLmdvb2dsZVZ2ZXJja2F0ZSBjbi50LmNvb5SIsInN1YiI6InJvbW92ZXQxMTI1OTAN1JyaXNzcmV2Mi4iIiwiaW5mbWFsVXJlbmRlc3MiOjEiLCkuc2V5IjpcIjEwNzQ5MDUwMCIsIm1ldG9yeV9yZXBvc2l0ZSI6MTA1Mi4iLCJpZnJlZWQiOiJwcm92aWRlbnQ6Q3JlYXR0cmlvbWVyIiwiY2F0aXJpYyI6IjhhZG1wcm92aWRlbnQ6Q3JlYXR0cmlvbWVyIiwia2V5IjpcIjIzNjIwMjM5MzAiLCJoaXN0b3J5IjpcIjIzNjIwMjM5MzAiLCJidWNrZXRjaGVtZSI6IiJ9..
Use the access token to query Google for user info

```bash
export ACCESS_TOKEN=ya29.CjNnA0sBU6up4orY7ZrKGdvbdKYuePnAa7p

curl -H "Authorization: Bearer $ACCESS_TOKEN" \   
https://www.googleapis.com/oauth2/v3/userinfo
```
Google OIDC with Curl

Should have this form

```json
{
  "sub": "102208411259056523752",
  "name": "",
  "given_name": "",
  "family_name": "",
  "picture": "https://lh3.googleusercontent.com/-XdUIqdMkCWA/AAAAAAAAAAI/AAAAAAAAAAA/4252rscbv5M/photo.jpg",
  "email": "skoranda@sphericalcowgroup.com",
  "email_verified": true,
  "hd": "sphericalcowgroup.com"
}
```
CILogon OIDC Gateway

- CILogon (a SAML to OIDC gateway)
- Provides a SAML-OIDC gateway
- Enables InCommon IdPs with OIDC applications
- https://cilogon.org
Federated Identity Management For Research Organizations
Federated Identity Management For Research Organizations
Federated Identity Management For Research Organizations
Federated Identity Management For Research Organizations

CLILogon OIDC Gateway

- Use CLILogon for authentication

```
- cd /etc/httpd/conf.d
- rm -f openidc.conf
- ln -s openidc-cilogon.conf.template openidc.conf
- service httpd restart
```

- Browse to https://sp0N.vo.idm.training/secure/
For more info on CILogon

- Visit [http://www.cilogon.org/oidc](http://www.cilogon.org/oidc)
- Ask Jim :)
- Contact help@cilogon.org
Identity Access and Management with Globus
Globus delivers…

research data management

…as SaaS and PaaS
Globus Services

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.

- Access via web browser or command line
- Use any storage system
- Use existing identity
Security PaaS challenges

• **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

• Identity and access management (IAM) platform-as-a-service
• Simplifies 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

[docs.globus.org/api/auth]
Use Case: Log in with Globus

- Similar to: “Log in with Google” “Log in with Facebook”
- Using existing identities
- Providing access to community services
Adding your identity provider

- InCommon identity providers that release Research & Scholarship attributes to CILogon
- Any other OpenID Connect identity provider
Based on widely used web standards

- OAuth 2.0 Authorization Framework (a.k.a. OAuth2)
- OpenID Connect Core 1.0 (a.k.a. OIDC)
- Access via OAuth2 and OIDC libraries of your choice
  - Google OAuth Client Libraries (Java, Python, etc.), Apache mod_auth_openidc, etc.
  - Globus Python SDK

[docs.globus.org/api/auth](docs.globus.org/api/auth)
Use case: Portal calling services on user’s behalf

• **Examples**
  – Portal starting transfer for user
  – Portal accessing data for processing on behalf of user

• **Authorization Code Grant**
  – With service scopes
  – Can also request OIDC scopes

• **Confidential client**

• **Globus SDK:**
  – To get tokens: ConfidentialAppAuthClient
  – To use tokens: AccessTokenAuthorizer
**Authorization Code Grant**

1. Access portal
2. Redirects user
3. User authenticates and consents
4. Authorization token
5. Authenticate using client id and secret, send authorization code
6. Access tokens
7. Authenticate with access tokens to invoke operation on service as user
Example: science gateways/portal
Use case: Native apps

• **Examples (any client that cannot keep a secret)**
  – Command line, desktop apps
  – Mobile apps
  – Jupyter notebooks

• **Native app is registered with Globus Auth**
  – Not a confidential client

• **Native App Grant is used**
  – Variation on the Authorization Code Grant

• **Globus SDK:**
  – To get tokens: NativeAppAuthClient
  – To use tokens: AccessTokenAuthorizer
Native App grant

1. Run application

2. URL to authenticate

3. Authenticate and consent

4. Auth code

5. Register auth code

6. Exchange code

7. Access tokens

8. Authenticate with access tokens to invoke operation on service as user

Any Service (Resource Server)

Globus Auth (Authorization Server)

Native App (Client)

Browser
Example: Command Line Interface

```bash
$ globus
Usage: globus [OPTIONS] COMMAND [ARGS]...

Options:
- v, --verbose                   Control level of output
- h, --help                      Show this message and exit.
- F, --format [json|text]        Output format for stdout. Defaults to text
- jmespath, --jq TEXT            A JMESPath expression to apply to json output.
                                    Takes precedence over any specified '--format' and forces
                                    the format to be json processed by this expression
- map-http-status TEXT           Map HTTP statuses to any of these exit codes: 0,1,50-99. e.g. "404=50,403=51"

Commands:
  bookmark          Manage Endpoint Bookmarks
  config            Modify, view, and manage your Globus CLI config.
  delete            Submit a Delete Task
  endpoint          Manage Globus Endpoint definitions
  get-identities    Lookup Globus Auth Identities
  list-commands     List all CLI Commands
  login             Login to Globus to get credentials for the Globus CLI
  logout            Logout of the Globus CLI
  ls                 List Endpoint directory contents
  mkdir             Make a directory on an Endpoint
  rename            Rename a file or directory on an Endpoint
  task               Manage asynchronous Tasks
  transfer          Submit a Transfer Task
  version           Show the version and exit
  whoami            Show the currently logged-in identity.
```

docs.globus.org/cli
Use case: Apps that need long-lived access tokens

• Examples
  – Portal checks for transfer status when user is not logged in
  – Run command line app from script for automation

• App requests refresh tokens

• Globus SDK:
  – To get token: ConfidentialAppClient or NativeAppClient
  – To use tokens: RefreshTokenAuthorizer
Refresh tokens

1. Run application

2. URL to authenticate

Browser

3. Authenticate and consent

4. Auth code

5. Register auth code

6. Exchange code, request refresh tokens

Globus Auth (Authorization Server)

7. Access tokens and refresh tokens

9. Exchange refresh token for new access tokens

Native App (Client)

8. Store refresh tokens

10. Access tokens

Any service (Resource Server)

11. Authenticate with access tokens to invoke an operation service as user
Example: Automation – data replication

- Backup or sync files without user present
- Meant to be run via cron or other task manager
- Native app grant
Use case: App invoking services as itself

• **Examples**
  – Gateway accessing resources as itself for user
  – Robots, agents, services

• Every app is/has an identity in Globus Auth (<client_id>@clients.auth.globus.org)

• App registers with Globus to get client id/secret
  – Native app cannot do this (no client_secret)

• Uses Client Credential Grant

• Can use the client_id just like any other identity_id
  – Sharing access manager role, permissions, group membership, etc.

• Globus SDK:
  – To get tokens: ConfidentialAppAuthClient
  – To use tokens: AccessTokenAuthorizer
1. Authenticate with portal client id and secret

2. Access Tokens

3. Authenticate as portal with access tokens to invoke service

Globus Auth (Authorization Server)

Any Service (Resource Server)
Example: Data distribution or analysis portals

- App access resource as itself
- Client credential grant
  - Portal or service
  - Permission for the client id
Use Case: Securing your service’s REST API

- 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
Use Case: Dependent services

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

• 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!
Summary

- Globus Auth makes it easy to:
  - add user login to your applications
  - integrate with Globus, XSEDE, and other services
  - add OAuth2 support to your service’s REST API
  - create services to leverage other services

Globus enables an integrated ecosystem of services and applications for the research community
Identity Access and Management with Globus

Federated Identity Management for Research Organizations
AGU 2017

Rachana Ananthakrishnan
University of Chicago - Globus
ranantha@uchicago.edu
Globus delivers…

research data management

…as SaaS and PaaS
Globus Services

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.

• Access via web browser or command line
• Use any storage system
• Use existing identity
Security PaaS challenges

• **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

- Identity and access management (IAM) platform-as-a-service
- Simplifies 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

[docs.globus.org(api/auth)](https://docs.globus.org/api/auth)
Use Case: Log in with Globus

• Similar to:
  “Log in with Google”
  “Log in with Facebook”

• Using existing identities

• Providing access to community services
Adding your identity provider

• InCommon identity providers that release Research & Scholarship attributes to CILogon

• Any other OpenID Connect identity provider
Based on widely used web standards

• OAuth 2.0 Authorization Framework (a.k.a. OAuth2)
• OpenID Connect Core 1.0 (a.k.a. OIDC)
• Access via OAuth2 and OIDC libraries of your choice
  – Google OAuth Client Libraries (Java, Python, etc.), Apache mod_auth_openidc, etc.
  – Globus Python SDK

docs.globus.org/api/auth
Use case: Portal calling services on user’s behalf

• **Examples**
  – Portal starting transfer for user
  – Portal accessing data for processing on behalf of user

• **Authorization Code Grant**
  – With service scopes
  – Can also request OIDC scopes

• **Confidential client**

• **Globus SDK:**
  – To get tokens: ConfidentialAppAuthClient
  – To use tokens: AccessTokenAuthorizer
Authorization Code Grant

1. Access portal

2. Redirects user

3. User authenticates and consents

4. Authorization token

5. Authenticate using client id and secret, send authorization code

6. Access tokens

7. Authenticate with access tokens to invoke operation on service as user

Browser (User)

Modern Research Data Portal

Portal (Client)

Globus Auth (Authorization Server)

Any service (Resource Server)
Example: science gateways/portal
Use case: Native apps

• **Examples (any client that cannot keep a secret)**
  – Command line, desktop apps
  – Mobile apps
  – Jupyter notebooks

• **Native app is registered with Globus Auth**
  – Not a confidential client

• **Native App Grant is used**
  – Variation on the Authorization Code Grant

• **Globus SDK:**
  – To get tokens: NativeAppAuthClient
  – To use tokens: AccessTokenAuthorizer
1. Run application
2. URL to authenticate
3. Authenticate and consent
4. Auth code
5. Register auth code
6. Exchange code
7. Access tokens
8. Authenticate with access tokens to invoke operation on service as user

Browser

Native App (Client)

Globus Auth (Authorization Server)

Any Service (Resource Server)
Example: Command Line Interface

```
$ globus
Usage: globus [OPTIONS] COMMAND [ARGS]...

Options:
- -v, --verbose  Control level of output
- -h, --help     Show this message and exit.
- F, --format [json|text]  Output format for stdout. Defaults to text
--jmespath, --jq TEXT  A JMESPath expression to apply to json output.
                        Takes precedence over any specified '--format' and forces
                        the format to be json processed by this expression
--map-http-status TEXT  Map HTTP statuses to any of these exit codes:
                        0,1,50-99. e.g. "404=50,403=51"

Commands:
  bookmark       Manage Endpoint Bookmarks
  config         Modify, view, and manage your Globus CLI config.
  delete         Submit a Delete Task
  endpoint       Manage Globus Endpoint definitions
  get-identities Lookup Globus Auth Identities
  list-commands  List all CLI Commands
  login          Login to Globus to get credentials for the Globus CLI
  logout         Logout of the Globus CLI
  ls             List Endpoint directory contents
  mkdir          Make a directory on an Endpoint
  rename         Rename a file or directory on an Endpoint
  task           Manage asynchronous Tasks
  transfer       Submit a Transfer Task
  version        Show the version and exit
  whoami         Show the currently logged-in identity.
```

docs.globus.org/cli
Use case: Apps that need long-lived access tokens

• **Examples**
  – Portal checks for transfer status when user is not logged in
  – Run command line app from script for automation

• **App requests refresh tokens**

• **Globus SDK:**
  – To get token: ConfidentialAppClient or NativeAppClient
  – To use tokens: RefreshTokenAuthorizer
Native App (Client)

1. Run application

2. URL to authenticate

Browser

3. Authenticate and consent

4. Auth code

5. Register auth code

6. Exchange code, request refresh tokens

7. Access tokens and refresh tokens

Globus Auth (Authorization Server)

8. Store refresh tokens

9. Exchange refresh token for new access tokens

10. Access tokens

Any service (Resource Server)

11. Authenticate with access tokens to invoke an operation service as user
Example: Automation – data replication

- Backup or sync files without user present
- Meant to be run via cron or other task manager
- Native app grant
Use case: App invoking services as itself

- **Examples**
  - Gateway accessing resources as itself for user
  - Robots, agents, services

- **Every app is/has an identity in Globus Auth** (<client_id>@clients.auth.globus.org)

- **App registers with Globus to get client id/secret**
  - Native app cannot do this (no client_secret)

- **Uses Client Credential Grant**

- **Can use the client_id just like any other identity_id**
  - Sharing access manager role, permissions, group membership, etc.

- **Globus SDK:**
  - To get tokens: ConfidentialAppAuthClient
  - To use tokens: AccessTokenAuthorizer
Client credential grant

1. Authenticate with portal client id and secret

2. Access Tokens

3. Authenticate as portal with access tokens to invoke service
Example: Data distribution or analysis portals

- App access resource as itself
- Client credential grant
  - Portal or service
  - Permission for the client id
Use Case: Securing your service’s REST API

• 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
Use Case: Dependent services

- Your service can act as client to other services (scopes)
  - Globus Transfer and Auth
  - XSEDE (e.g., Jetstream, XUP)
  - Other community services

- 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!
Summary

• Globus Auth makes it easy to:
  – add user login to your applications
  – integrate with Globus, XSEDE, and other services
  – add OAuth2 support to your service’s REST API
  – create services to leverage other services

Globus enables an integrated ecosystem of services and applications for the research community
Hands-On Exercises
(see supplemental document)
Federation Topics: SAML Federations
The Role of Federations

- Enable us to scale up to 1000s of IdPs and SPs
- Publish digitally signed SAML metadata containing public keys, endpoint URLs, and other info about IdPs and SPs
- Set standards for SAML attributes, levels of assurance, etc.
- Provide support and training
InCommon: The US R&E Federation

- [https://www.incommon.org/](https://www.incommon.org/)
- Over 800 participants and growing: [https://www.incommon.org/participants](https://www.incommon.org/participants)
- 500+ IdPs / 4000+ SPs: [https://incommon.org/federation/info/all-entities](https://incommon.org/federation/info/all-entities)
- Becoming an InCommon Member: [https://www.incommon.org/join](https://www.incommon.org/join)
eduGAIN: Global InterFederation

- InCommon joined eduGAIN in 2016
  https://www.incommon.org/edugain/
- Enables SAML metadata exchange across federations, with per-entity opt-in/opt-out
eduGAIN: Global InterFederation

eduGAIN Policy Framework requires:

- Primarily serve the interests of the R&E sector
- Provide a point of contact for technical issues
- Provide processes for handling complaints and incidents
- Have a published Metadata registration practice statement

- [http://www.edugain.org](http://www.edugain.org)
- [https://technical.edugain.org/](https://technical.edugain.org/)
SAML Metadata

- InCommon metadata includes US IdPs/SPs plus international IdPs/SPs from eduGAIN: https://incommon.org/federation/metadata.html
- SAML metadata interoperability enables secure, scalable federation between IdPs and SPs
SAML Metadata

- SAML metadata is a digitally signed XML document that establishes trust in the federation:
  
  [Link](http://md.incommon.org/InCommon/InCommon-metadata.xml)
  
  [Link](https://spaces.internet2.edu/display/InCFederation/Metadata+Signing+Certificate)
<EntityDescriptor entityID="https://idp.ncsa.illinois.edu/idp/shibboleth">
  <IDPSSODescriptor errorURL="https://idp.ncsa.illinois.edu/error" protocolSupportEnumeration="...">
    <Extensions>
      <shibmd:Scope regexp="false">ncsa.illinois.edu</shibmd:Scope>
      <mdui:UIInfo>
        <mdui:DisplayName xml:lang="en">National Center for Supercomputing Applications</mdui:DisplayName>
        <mdui:Description xml:lang="en">National Center for Supercomputing Applications</mdui:Description>
        <mdui:PrivacyStatementURL xml:lang="en">...</mdui:PrivacyStatementURL>
        <mdui:Logo height="100" width="148" xml:lang="en">.../mdui:Logo>
      </mdui:UIInfo>
    </Extensions>
    <KeyDescriptor use="signing">...</KeyDescriptor>
  </IDPSSODescriptor>
  <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
      Location="https://idp.ncsa.illinois.edu/idp/profile/SAML2/Redirect/SSO"/>
  <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
      Location="https://idp.ncsa.illinois.edu/idp/profile/SAML2/POST/SSO"/>
</EntityDescriptor>
<EntityDescriptor entityID="https://cilogon.org/shibboleth">
  <SPSSODescriptor protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <Extensions>...</Extensions>
    <KeyDescriptor>...</KeyDescriptor>
    <AssertionConsumerService Binding="..." Location="https://cilogon.org/Shibboleth.sso/SAML2/POST"/>
    <AttributeConsumingService index="1">
      <ServiceName xml:lang="en">CILogon</ServiceName>
      <ServiceDescription xml:lang="en">...</ServiceDescription>
      <RequestedAttribute FriendlyName="displayName" Name="urn:oid:2.16.840.1.113730.3.1.241" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"/>
      <RequestedAttribute FriendlyName="eduPersonPrincipalName" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"/>
      <RequestedAttribute FriendlyName="eduPersonTargetedID" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.10" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"/>
      <RequestedAttribute FriendlyName="mail" Name="urn:oid:0.9.2342.19200300.100.1.3" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"/>
    </AttributeConsumingService>
  </SPSSODescriptor>
</EntityDescriptor>

Federated Identity Management For Research Organizations
2017 AGU New Orleans
InCommon SAML Federation Trust Model

- SAML metadata bootstraps the trust
- Trust federation operators to only let in "good" metadata
- Each entity (IdP or SP) consumes the metadata and has details for all other entities necessary for interoperability
- Before consuming metadata each entity must verify it is properly signed by federation operator and still valid
- TLS is not a substitute for verifying signature on metadata
Federation Topics: IdP Discovery
IdP Discovery

- SP federated with more than one IdP?
- How does the SP decide to which IdP to send the browser?
- Many bad ideas proven through experience to not work:
  - IP address of the browser
  - Different URLs as gateways to different IdPs
  - Referring pages
IdP Discovery

- Bad approaches trying to prevent the actual best solution
  - Prevent one or two extra "clicks"
  - Results in more user confusion over the long term

Best approach: Ask the user to choose the IdP!
Select an Identity Provider

Please select a trusted identity provider from the list below by beginning to type in the edit box.

Enter your organization's name

Allow me to pick from a list

CONTINUE

Help
Sign in using your

- Personal Account
- Institutional Account

Sign in with your ORCID account

Email or ID
Email or ID

ORCID Password
ORCID Password

Forgotten password?

Sign into ORCID

Sign in with a social media account

Login Required

Please choose how to login

Use a suggested selection:

LIGO

LIGO Guest

Or enter your organization's name

Allow me to pick from a list

Help
REFEDS DISCOVERY GUIDE

REFEDS demonstrates the most effective way to present federated identity to users of your site, with best practice and examples of how to provide the best experience.

BEST PRACTICE GUIDE

In just 4 simple steps you can learn the key recommendations from the NISO ESPRESSO report and find out how to implement the best practice guide with visual demonstrations of how to, and how not to, use federated login effectively.

DISCOVERY DEMO

See a guided demo of how to implement the best practice guide with visual demonstrations of how to, and how not to, use federated login effectively.
Primary question for research organizations: Use a centralized discovery service shared by many SPs or embedded (per-SP) approach?
IdP Discovery: Research Organizations

Centralized:
- One discovery service leveraged by multiple SPs
- Less operational overhead
- Easier to manage user state: "Use a previous choice..."
- Single point of failure
- Can lead to a jarring user experience
  - Switching between the UIs of the application, discovery service, and IdP
IdP Discovery: Research Organizations

Embedded (per-SP):

- Easier to preserve "look and feel" of the application
- Less jarring visual experience for user
- No single point of failure
- More operational overhead
- Harder to manage consistent user experience across organization
IdP Discovery: Shibboleth EDS

Pairs well with the Shibboleth SP

"Embedded" but often deployed with an un-federated Shibboleth SP and used as a centralized discovery service

The Embedded Discovery Service provides a web interface allowing a user to select which Identity Provider they will use when accessing a Service Provider. This product is co-installed with a Service Provider and allows the discovery service to carry the same UI and branding.

Key Features

- Simple installation and configuration as HTML, Javascript and CSS files are deployed in the same manner as for any given web page on your site.
- Provides a smaller, easier to navigate list of Identity Providers by only presenting those known by your Service Provider.
- Supports assistive technologies such as screen readers.
Federation Topics: Joining InCommon
Approach 1: Join Directly
Join InCommon

Joining InCommon is the first step toward accessing several services:

- **federated identity management** (no additional charge)
- assurance program
- **certificate service** (additional charge applies and a legal addendum required)
- **multifactor authentication programs** (additional charge applies and legal addendum required)

Once the joining process is complete, InCommon participants can choose the services they wish to use.

1. **Are You Eligible?**

Participation in InCommon is open to:

1. **Higher Education**
   Two- and four-year, degree-granting academic institutions that are accredited by a U.S. Department of Education Regional Institutional Accrediting Agency, or some national or state accrediting agencies. If you represent an institution of higher education, continue to Step #2 below.
   1. If you are a California Community College, please go to this page for special instructions and agreement.

2. **Research Organizations**
   A Research Organization is defined as a lab, facility, or center related to a particular federal research agency and listed on an official publicly available government listing. See the joining process specific to Research Organizations. Research Organizations are eligible to sponsor partners into the federation.

3. **Sponsored Partners**
   Business, education, and research organizations who partner with higher education may join the Federation as Sponsored Partners. Sponsored Partners must be sponsored by the designated Executive of a current InCommon Higher Education Institution or Research Organization. If you are a business or other potential sponsored partner - or if you are a higher education institution looking to sponsor an entity, please review the process for sponsorship.
### Research Organizations

Annual fees for research organizations are based on full-time equivalent (FTE) staff. Research organizations should verify their FTE by sending a link to a web page listing that information, a page from an annual report, or a statement on letterhead.

<table>
<thead>
<tr>
<th>InCommon Annual Fees for Research Organizations (based on full-time equivalent staff)</th>
<th>Level</th>
<th>Annual Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTE more than 10,000</td>
<td>L1</td>
<td>$3,250</td>
</tr>
<tr>
<td>FTE 2,000-10,000</td>
<td>L2</td>
<td>$2,700</td>
</tr>
<tr>
<td>N/A</td>
<td>L3</td>
<td>$1,600</td>
</tr>
<tr>
<td>FTE less than 2,000</td>
<td>L4</td>
<td>$1,180</td>
</tr>
</tbody>
</table>
12. Insurance

Participant covenants and agrees to obtain and maintain in force, at its own expense, throughout the term of this Agreement, commercial general liability insurance coverage with a combined single limit of not less than $3,000,000.00 each occurrence or its equivalent, whether such insurance is maintained through self-insurance or through third party insurance, against claims, regardless of when asserted, that may arise out of, or result from, Participant’s participation in the Federation.
Approach 2: Leverage existing membership

- Most smaller research projects have a lead campus
- That lead campus is often an InCommon Participant
- Campuses will often publish SP metadata
  - Some require the SPs be operated by campus employees
  - Can delegate admin access if they so desire
- IdPs more problematic
  - Extra cost for more than one IdP
  - Not willing to vouch for non-campus VO members
Manage Service Provider > Service Providers

Your Service Providers

Add Entity ID

Add a New Service Provider

Existing Service Providers

https://ask.ligo.org/shibboleth-sp

https://cris-dev-liam.ligo.org/shibboleth-sp

https://gracedb.ligo.org/shibboleth-sp
Federation Topics: Attribute Release
Attributes: eduPerson

The eduPerson specification defines the attributes used in InCommon/eduGAIN:

- `eduPersonPrincipalName` (user@example.edu)
- `eduPersonTargetedID` (https://example.edu/idp!https://example.org/sp!ChF3nEYtvG3a5G4Xv0g=)
- `eduPersonUniqueld` (28c5353b8bb34984a8bd4169ba94c606@example.edu)
- `eduPersonOrcid` (http://orcid.org/0000-0002-1825-0097)
- `eduPersonAffiliation` (faculty, student, staff, alum, member, affiliate)
- `eduPersonScopedAffiliation` (faculty@example.edu, member@example.edu)
- `eduPersonEntitlement` (http://example.com/contracts/HEd123)
- `displayName` (John Smith)
- `givenName` (John)
- `sn` (Smith)
- `mail` (jsmith@example.edu)
The Reality of Attribute Release

Attributes IdP is able/willing to release varies, due to:

- Privacy concerns (US FERPA, EU/AAF Data Protection)
- Federation boundaries (eduGAIN opt-in/opt-out)
- Technical limitations (no eduPersonTargetedID support)
- Lack of data (e.g., guest accounts, no email contact info)
The Reality of Attribute Release

Attribute release can vary across the IdP’s membership:

- Students / Faculty / Staff / Guests / Affiliates
- Emeritus / Retired / Pre-matriculated / Alumni
- Directory Opt-Out / FERPA Hold / Protective Order
The Reality of Attribute Release: IDs

IdP may provide ePPN and/or ePTID to your SP

eduPersonPrincipalName (ePPN):
- May be re-assigned (usually after some hiatus period)
- Mitigate re-assignment through SP-side offboarding

eduPersonTargetedID (ePTID):
- Non-reassigned, targeted, opaque

https://example.edu/idp!https://example.org/sp!ChF3nEYtvG0S3a5G4Xv0g=
The Reality of Attribute Release: Names

IdP may provide displayName or givenName and surname

displayName:
- Single valued
- Don't assume it contains spaces

givenName and surname:
- Multi-valued

May contain arbitrary UTF-8 characters, not just ASCII
REFEDS Research and Scholarship (R&S)

Defines an attribute bundle:

● personal identifiers:
  ○ email address
  ○ person name
  ○ eduPersonPrincipalName

● pseudonymous identifier: eduPersonTargetedID

● affiliation: eduPersonScopedAffiliation
REFEDS Research and Scholarship (R&S)

Federations tag R&S entities in metadata:

- Research and Scholarship SPs
- IdPs that release the attribute bundle to R&S SPs
- Research and Scholarship Entity Category
  - https://refeds.org/category/research-and-scholarship
<EntityDescriptor entityID="https://idp.ncsa.illinois.edu/idp/shibboleth">
  <Extensions>
    <mdrpi:RegistrationInfo registrationAuthority="https://incommon.org"/>
    <mdattr:EntityAttributes xmlns:mdattr="urn:oasis:names:tc:SAML:metadata:attribute">
      <saml:Attribute xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        Name="http://macedir.org/entity-category-support"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
        <saml:AttributeValue>http://refeds.org/category/research-and-scholarship</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        Name="http://macedir.org/entity-category"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
      </saml:Attribute>
    </mdattr:EntityAttributes>
  </Extensions>
  ...
</EntityDescriptor>
<EntityDescriptor entityID="https://cilogon.org/shibboleth">
    <Extensions>
        <mdrpi:RegistrationInfo registrationAuthority="https://incommon.org"/>
        <mdattr:EntityAttributes xmlns:mdattr="urn:oasis:names:tc:SAML:metadata:attribute">
                <saml:AttributeValue>http://id.incommon.org/category/research-and-scholarship</saml:AttributeValue>
                <saml:AttributeValue>http://refeds.org/category/research-and-scholarship</saml:AttributeValue>
            </saml:Attribute>
        </mdattr:EntityAttributes>
    </Extensions>
    ...
</EntityDescriptor>
Federation Topics: Unaffiliated IdPs
Unaffiliated IdPs (or "IdP of Last Resort")

User may not be able to log in with “home campus” identity:
- University not a federation member
- University doesn’t operate an IdP
- International IdP did not opt-in to eduGAIN
- IdP does not release ePPN or ePTID

Approaches:
- External Guest IdP: Google, AAF Virtual Home, UnitedID,…
- Local Guest IdP: On-premises or cloud
Community Operated IdPs
Community Operated IdPs
Guest IdP Operated By Organization
Non-Browser Clients and Federated Identity
Non-Browser

- SAML and OIDC primarily target web browser SSO
- Both do have non-browser "profiles"
  - SAML: Enhanced Client or Proxy (ECP) profile
  - OIDC: Flows non-browser agnostic by definition
- In practice primary issue is user authentication
  - How does the client collect some federated credential?
  - Can the IdP or OP consume the credential?
Non-Browser with OIDC

- Already demonstrated flow using curl commands
  - Used Google as the OP
- But authenticated and obtained auth code using web browser
  - Google login form is browser based
  - Really does not function well outside of browser
  - Clients can "screen scrape" and POST
    - Not a scalable approach across OPs
  - Mobile and desktop clients throw up a browser
Non-Browser with OIDC

- Other authentication mechanisms possible
  - X.509 certificates or Kerberos anyone?
  - Rarely supported by OPs
- More authentication support for "non-user" flows
- Non-browser simply not a large use case
- Still, rich set of developer tools available for OIDC
  - Pick your favorite framework
  - Has or will have soon OIDC library or module
SAML ECP

- SAML Enhanced Client or Proxy (ECP) profile
  - Focus today on clients and not proxies
- "Enhanced" client must:
  - Initiate flow at the SP and receive auth request
  - Take auth request to IdP and authenticate the user
  - Receive assertion from IdP
  - Take assertion to the SP and begin session (if any)
  - Protect integrity of flow at all times
Primary issue again is authentication

- Most IdPs do not support ECP
  - Block the ECP URL endpoint
  - Cannot support anything outside "screen scraping"

- Situation improving (?) with Shibboleth IdP v3
  - "Out of the box" ECP support for most authentication flows
  - Operators have to go out of way to turn it off
  - Turns out many do...
SAML ECP

Secondary issue is no support at SP

- Well supported by Shibboleth SP
- Reasonable support by SimpleSAMLphp
  - Extra work for deployers
- Virtually no support by other SAML implementations
- Doesn't Office365 do ECP?
  - Well, sort of, ...
  - Does drive campus IdPs to expose ECP (sometimes)
Non-HTTP Clients

- OIDC and SAML use HTTP(S) as the transport (primarily)
- Research organizations rely on non-HTTP protocols
  - SSH
  - LDAP
  - GridFTP
  - Domain and application specific protocols/tools
- How can organizations leverage federated identity for these applications?
SSH and Federated Identity

Past efforts to evolve SSH to leverage federated identity

● SAML Enhanced Client SASL and GSS-API Mechanisms
  ○ Add "GSS-ECP" as another GSS implementation
    ■ After Kerberos v5 and RFC 3820 proxy certificates
  ○ Stalled: no funding and no development

● Project Moonshot
  ○ RADIUS as federation mechanism (think eduROAM)
  ○ SAML for user authz based on attributes
  ○ Stalled: no takeup outside of UK
SSH and Federated Identity Today

Most often used approach is federated identity HTTPS access to application for managing SSH keys

- Users authenticate using federated identity
- Upload SSH keys
- Infrastructure provisions as necessary
- Infrastructure manages lifecycle
- Saw this earlier with COmanage
LDAP, GridFTP, and Other

Follow SSH pattern

- Use federated identity to get to "portal" (COmanage)
- Allow user to provision password or other tokens
  - COmanage "Authenticator" functionality
- Tie de-provisioning of tokens to user lifecycle management
Participant Lifecycle Management Using Open Source Tools
What Is Participant Lifecycle Management?
Typical User Lifecycle Management (eg: Hitachi ID)

- **Onboard**
  - Slow: too much paper, too many people.
  - Expensive: too many administrators doing redundant work.

- **Manage**
  - Role changes: add/remove rights.
  - Policies: enforced?
  - Audit: are privileges appropriate?
  - Org. relationships: track and maintain.

- **Support**
  - Passwords: too many, too weak, often forgotten.
  - Access: Why can’t I access that application / folder / etc.

- **Deactivate**
  - Reliable: notification of terminations.
  - Fast: response by sysadmins.
  - Complete: deactivation of all IDs.
COmanage

- Person Registry (multi-tenant)
- Manage onboarding and offboarding
- Manage multiple roles and external affiliations
- Basic access management
- Provisioning infrastructure for application integration
- Supports complex organizational structures
- Designed for federated identity
Grouper

- Group Registry
- Sync subject data from external sources, including COmanage
- Perform complex group (set) math
- Sophisticated access management
- Message based provisioning
- More suitable for larger VOs with complex requirements
Additional Components

- Relational Database
- LDAP
- IdPs, SPs, and SAML/OIDC proxies or gateways
Introduction to COmanage

There be Jargons
COmanage Concepts: Platform Organization

- Collaboration Management Platform (CMP)
  - COmanage Registry and its associated services
- Collaborative Organization (CO)
  - eg: VO
  - A "tenant" of the CMP
- Collaborative Organizational Unit (COU)
  - Allows representation of internal CO structure/hierarchy
- Admins at each level (CMP/CO/COU)
Welcome to COmanage Registry. Please select a collaboration.

**Available Collaborations**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COmanage</td>
<td>COmanage Registry Internal CO</td>
</tr>
<tr>
<td>MESS (Not a Member)</td>
<td>Medium Energy Synchrotron Source</td>
</tr>
</tbody>
</table>
COmanage Concepts: Identities

1. Org Identity: Pat S Lee
   - Prof of Perpetual Motion
   - plee@univ.edu

2. Org Identity: P Lee
   - plee@idp.com

3. CO Person: Pat Lee
   - plee@myco.org

4. CO Person Role:
   - Researcher
   - Perpetual Motion Project
   - Pasadena, CA
   - 50% FTE

5. CO Person Role:
   - Chief Administrator
   - My CO
   - Milwaukee, WI
   - 50% FTE

6. CO Group Member:
   - my_co_admin

7. CO Group Member:
   - motion_project

8. CO Group Member:
   - my_co_member

9. CO Group Member:
   - ice_cream_fans
COmanage Concepts: Organizational Identities

- Associated with an "external" identity
  - Federated (InCommon, EduGain, etc)
  - Social (Google, Twitter, etc)
  - ORCID
  - Often, but not always, associated with a credential

- Organizational Identity Sources
  - Various backends: Files, LDAP, SQL, etc
  - Example use cases: Campus SIS, HRMS
View Scott Koranda

This Organizational Identity was created from an Organizational Identity Source (EnvSource) and therefore cannot be edited.

Name and Affiliation

Scott Koranda (Primary, Official)

<table>
<thead>
<tr>
<th>Status</th>
<th>Synced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affiliation</td>
<td>Member</td>
</tr>
<tr>
<td>Title</td>
<td></td>
</tr>
<tr>
<td>Organization</td>
<td>University of Wisconsin-Milwaukee</td>
</tr>
<tr>
<td>Department</td>
<td></td>
</tr>
<tr>
<td>Valid From</td>
<td></td>
</tr>
</tbody>
</table>
View Scott Koranda

This Organizational Identity was created from an Organizational Identity Source (EnvSource) and therefore cannot be edited.

Name and Affiliation

Identifiers
- http://cilogon.org/serverA/users/10376 (System of Record ID)
- http://cilogon.org/serverA/users/10376 (ePTID)
- skoranda@uw.edu (ePPN)

Email Addresses
- skoranda@uw.edu (Official, Verified)
COmanage Concepts: CO People

- Operational Attributes for the CO
  - Names (must always have one Primary)
  - Identifiers (more on this later)
  - Email Addresses
  - Groups
  - SSH Keys
  - Status (calculated based on Roles)
Edit Scott Koranda

Name

Scott Koranda (Primary, Preferred)

Identifiers

- T100000 (TraineeID)
- 5000 (uidNumber)
- 5000 (gidNumber)
- /home/skoranda (homeDirectory)
- skoranda (UID)

Email Addresses

- skoranda@uwm.edu (Preferred, Unverified)

URLs

Add
COmanage Concepts: CO Person Roles

- A *CO Person Role* describes a person's relation to the CO
- A CO Person can have more than one CO Person Role
- If COUs are defined, each CO Person Role belongs to exactly one COU
- Each CO Person Role has its own Status
COmanage Concepts: CO Groups

- CO Groups attach to the CO
- A CO Person can have multiple CO Group Memberships
  - Member and/or Owner
- A group can be Open (self-signup) or Closed (memberships managed by Group Owner)
COmanage Concept: Identifiers

- Can be from external sources (e.g., ePPN)
- Can be auto-generated using *Identifier Assignments*
  - Pattern based, e.g., *abc23* or *albert.einstein@ligo.org*
  - Created automatically as part of Enrollment Flows
  - Per-application identifiers is common use case
- Subject to Validation
  - Internal availability checks
  - Plugin based format or external checks
COmanage Concepts: Enrollment Flows

- An Enrollment Flow is a configurable process for enrolling a CO Person, or adding additional attributes (Roles, Org Identities) to an existing CO Person.
- The Enrollment Flow collects attributes and records them in a Petition along with other enrollment data.
- Once the flow has completed, the Petition is finalized and becomes read-only.
- Day-to-day updates happen against operational attributes.
Types of Enrollment Flows (Onboarding)

- **Invitation**
  - Collaboration initiated, participant accepts

- **Self Signup**
  - Participant initiated

- **Application**
  - Participant initiated, approval required

- **Conscription**
  - Collaboration initiated
Types of Enrollment Flows (Other)

- Additional Role Enrollment
- Account Linking (additional credential enrollment)
  - ORCID Linking
- Renewal
- Additional Eligibility
  - For use when external databases hold eligibility information
COmanage Concepts: Provisioning Targets

- Implemented via *Provisioner Plugins*
- Out of the box:
  - LDAP, Grouper, GitHub, Changelog, MediaWiki, Mailman
  - LDAP and Grouper most commonly deployed
- More than one provisioning target of a given type
- Population-specific provisioning
  - Driven by CO Group memberships
<table>
<thead>
<tr>
<th>Description</th>
<th>Plugin</th>
<th>Status</th>
<th>Order</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training LDAP Server</td>
<td>Ldap Provisioner</td>
<td>Automatic Mode</td>
<td>1</td>
<td>Edit, Configure, Delete, Reprovision All</td>
</tr>
</tbody>
</table>

Page 1 of 1, Viewing 1-1 of 1
COmanage Concepts: Plugins

- Enroller
- Identifier Validator
- LDAP Schema
- Normalization
- Organizational Identity Source
- Provisioner
- Other
Hands-On Exercises
(see supplemental document)
Wrap Up
Acknowledgments

- Heather Flanagan, Spherical Cow Group
- Terry Fleury, NCSA, CIlogon, and U of Illinois
- Benn Oshrin, Spherical Cow Group
- Anne Wilson, University of Colorado, Boulder
- Lesley Wyborn, Australian National University
Acknowledgments

- Center for Trustworthy Scientific Cyberinfrastructure and Indiana University (trustedci.org)
- United States National Science Foundation

This material is based upon work supported by the National Science Foundation under grant number 1547272. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.
Thank You!