Supporting a Widely Deployed Campus Shibboleth Implementation

Russell Beall, University of Southern California
Brendan Bellina, University of Southern California
Scott Cantor, The Ohio State University
Rob Carter, Duke University
A modern fable...
In the Beginning there was the Shibboleth IdP...

And it was Good. And readily supported.
And some Central Services came to be integrated with the Shibboleth IdP…

And it was Good. And Support was small and managed.
And then many Department Services were integrated with the Shibboleth IdP...

And it was still Good. Though Support was strained.
And then even External Providers were integrated with the IdP...

Google

ExLibris

Sharepoint

SciQuest

Digital Measures

EzProxy

Lynda.com

Aetna

TeraGrid

CollegeNet

And many, many more

And it was Still Good. And then Support ...
This could be you.
Some (not all) Things to Consider

- Applications may be central, departmental, or externally hosted.
- Applications may be internally developed, open source, or vendor developed.
- Applications may require user provisioning just-in-time or ahead of time.
- Applications may not handle identifier changes well or at all.
- Applications may not allow merges.
- Departments may lack permanent staff for SP Support and expertise. Staff may leave and they may have relied upon contractors to establish the service.
- Expectation of 24x7x365 support for web applications accessible from any time zone.
- The technology scales, but does your support infrastructure?
The Ohio State University
University Environment

- 65,000 students, 28,000 employees
- 160,000 unique accounts
- Various “traditional” populations like alumni not given access to most campus services
- IT functions highly distributed
- Minimal funding/support of centralized infrastructure
- Shibboleth in production since 2004
IdP Environment

- Upgraded to 2.x last summer, also moved from Apache+Tomcat to Jetty
- Two Linux servers at separate data center locations, lots of CPU, low on RAM
- No clustering, deployed with extensions for attribute query handling and SSO via cookie
- IdP sessions are short-lived, traffic is pinned during user login sequence
- ~150,000 – 350,000 logins per day
SP Environment

- 15-20 off-campus SPs, mostly InCommon members
- ~250 on-campus SPs
- Largely manual, email-based registration and change management process
- “Trusted” customers with larger deployments maintaining their own signed metadata files for import by IdP
Support Model

- Had .25 FTE dedicated to operating the IdP and supporting SPs, bumped to .75 FTE for staffing redundancy
- Minimal local support for SP deployment, most questions directed to public lists
- Initial configuration files tailored to OSU environment provided with custom documentation on what to change
University of Southern California
Shibboleth Support Model

Brendan Bellina
Manager
Enterprise Identity Management
Information Technology Services
University of Southern California
Los Angeles, California, USA
bbellina@usc.edu

Russell Beall
Senior Programmer/Analyst
Information Technology Services
University of Southern California
Los Angeles, California, USA
beall@usc.edu
University Environment

- 42,000 students; 17,500 employees; 4800 affiliates
- 92,000 Shibboleth-enabled accounts
  - Includes legacy student accounts retained for Google Apps
- IT infrastructure:
  - Moderately funded for centralized services
  - Highly distributed departmental IT functions as well
IdP Environment

- 2.x since late 2007
- Terracotta clustering since late 2008
- Two Sun v440 machines in the cluster
- Two zones on Sun 5240 machines in DR
- IdP sessions last 8 hours
  - Some services using short SP sessions and forceAuthn
- 30K-60K authentications per day
- 40K-100K service logins per day
SP Environment

- 100+ on-campus SPs, soon to be 130+
- 29 off-campus SPs
- Governance process required for registration
  - Requirements meeting(s) to document requirements
  - Formal campus IT-leader committee approval required
  - Blanket attribute release disallowed (perhaps until uApprove)
    - Each SP generally has different access groups and attributes anyway
Support Model - Resources

- 0.5 FTE dedicated to IdP support and managing SP integration into the IdP
- SP admin group (small fraction of their time) for support of centralized SPs
- Departments and vendors responsible for supporting their SP deployments
- 2 FTE outside of central ITS assist in managing and documenting requests and governance
- Most user issues handled directly by CSC
Support Model – Docs and Tools

- Full-fledged “start-here” document with configuration generator application specialized to USC environment.
  - [http://shibboleth.usc.edu](http://shibboleth.usc.edu)

- USC email lists for Shibboleth IdP Admins, SP Admins, and Announcements. SP Admins are required to be on the lists.

- Internally developed tool for displaying what attributes are released for a specific user to a specific SP. Helps when troubleshooting access issues.
Support Model - Policies

- Entitlement based authorization rather than attribute based. Entitlement assigned based on group memberships with a delegated tool for group administration for exceptions.

- Require department applications that consume USCID to also consume Historical USCID.

- Restrict release of non-persistent identifiers such as NetID, and instead release persistent identifiers.

- No undocumented changes are made.
Tourist Information

- University + Academic Health System + hospitals & clinics

- 6500 ugrad, 8500 grad, 3200 faculty, 30,000 staff (mostly in Health System)

- Counting alumni & affiliates, over 78,000 active accounts, 490,000 total identities (active & inactive)
IT Environment

- Roughly 50/50 central/departmental IT staff
- Central IT incorporates both OIT (spanning entire enterprise) and DHTS (Health System Central IT)
- PeopleSoft Student system & SAP HR/Payroll
- Traditional “free for all” in web space -- all clients/servers welcome
IAM Environment

- Kerberos since early 1990s
- LDAP since ca. 2002
- ERP linkages since early 2000s
- WebISO since early 2000s, starting with homegrown Webauth, transitioned to Shibboleth ca. 2005
- Grouper site since ca. 2005, now with >> 350,000 groups
- OIM since ca. 2008
Shibboleth IDP Environment

- IDP 2.x running atop Centos Linux systems w/ Apache + Tomcat + Terracotta
- 4 production primary IDPs w/ hw load balancing
- 2 Terracotta clusters -- one pair active at a time
- Dedicated LDAP attribute repository (3-way multimaster Oracle/Sun LDAP server farm)
- Back-end authN via Kerberos 5, integrated OpenID gateway, optional multifactor interface
Shibboleth IDP Environment

- Additional, unfederated dedicated IDPs for some internal purposes
- 50,000 - 120,000 logins per day (mean: 60,000) in primary IDPs, far less in unfederated internal IDPs
Shibboleth SP Environment

- 850 distinct SPs entity IDs registered (in local-sites.xml)
  - 750 in full production, ca. 100 in test or ppt
  - All but 40 are local -- roughly 1/3 of those are “cloud” services (SaaS)
  - Growth rate has increased dramatically since late 2010, when we deployed current SP support tools... 450 SPs registered between 2005 and 2011, 400 in 2011-2012 so far...
SP Owner Support Infrastructure

- Shibbolized web site for managing local SP registrations
- Owners authN with their NetIDs then see the SPs they’ve registered (or been designated as admins for) and can edit or add new registrations
- Optionally produces SP configuration XML based on basic information and some reasonable defaults
- Registrations complete in real-time -- usually up and running within a minute or two of clicking “create”
### IDP Support Web Site

**SP Registrations**

<table>
<thead>
<tr>
<th>Entity</th>
<th>Public Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="https://groupemail.cit.duke.edu">https://groupemail.cit.duke.edu</a></td>
<td>OIT Mass Email Tool</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://hr-dev.oit.duke.edu">https://hr-dev.oit.duke.edu</a></td>
<td>HR Dev Website</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-mlt.oit.duke.edu">https://idms-mlt.oit.duke.edu</a></td>
<td>Multi Factor Authentication</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-sdweb-test-01.oit.duke.edu/shibboleth">https://idms-sdweb-test-01.oit.duke.edu/shibboleth</a></td>
<td>Service Desk Portal</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-sdweb.oit.duke.edu/shibboleth">https://idms-sdweb.oit.duke.edu/shibboleth</a></td>
<td>Service Desk Portal</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-test.oit.duke.edu/shibboleth">https://idms-test.oit.duke.edu/shibboleth</a></td>
<td>OIT</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-web-dev-01.oit.duke.edu/accounts/">https://idms-web-dev-01.oit.duke.edu/accounts/</a></td>
<td>OIT</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-web-dev-01.oit.duke.edu/cgi-bin/chgmars">https://idms-web-dev-01.oit.duke.edu/cgi-bin/chgmars</a></td>
<td>OIT</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-web-dev-01.oit.duke.edu/dukid">https://idms-web-dev-01.oit.duke.edu/dukid</a></td>
<td>OIT</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
<tr>
<td><a href="https://idms-web-dev-01.oit.duke.edu/shibboleth">https://idms-web-dev-01.oit.duke.edu/shibboleth</a></td>
<td>OIT</td>
<td>Published Show Edit Destroy XML Users</td>
</tr>
</tbody>
</table>
# New SP Registration

## Service Provider Information

The key information in this section is the Entity Id or Service Provider Id. It's how the Service Provider uniquely identifies itself to the Identity Provider, IdP. Please provide the name of the application or site in the Public Name field. Provide a brief description of its function and purpose. Provide who is accessing the site and how you plan to implement authorization controls to retool eligibility requirements in the Audience field. Responsible Dept and Function owner dept are based on the Org Codes, e.g. "WebSvcs:InfoTech:Library", "TechnologyServices:DFAS:TrinityCollege". This is an auto-complete field. See the list at OU Names.

<table>
<thead>
<tr>
<th>Entity</th>
<th><a href="https://sample.duke.edu">https://sample.duke.edu</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>[ ]</td>
</tr>
<tr>
<td>Public name</td>
<td>Sample Dept</td>
</tr>
<tr>
<td>Functional purpose</td>
<td></td>
</tr>
<tr>
<td>Responsible dept</td>
<td></td>
</tr>
<tr>
<td>Function owner dept</td>
<td></td>
</tr>
<tr>
<td>Audience</td>
<td></td>
</tr>
</tbody>
</table>
**Contacts**

Technical contacts are the ones responsible for setting up the Shibboleth configuration and other administrative tasks on the Service Provider. Administrative or Functional are the site owners. Use an e-mail list address too. NetID is not required.

<table>
<thead>
<tr>
<th>Type</th>
<th>technical</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetID</td>
<td>rob</td>
</tr>
<tr>
<td>E-mail</td>
<td><a href="mailto:robert.carter@duke.edu">robert.carter@duke.edu</a></td>
</tr>
</tbody>
</table>

**Users**

The netids listed here are users that have access to update this Service Provider's information.

<table>
<thead>
<tr>
<th>NetID</th>
<th>rob</th>
</tr>
</thead>
</table>

Add a contact

Remove this contact

Add a User

Remove this user
Service Provider Details

Generate a self-signed key/cert. Copy the cert (between the header and trailer) and paste here. Use a long life span for the certificate of 5, 10 or more years.

Certificate

Assertion Consumer Service

These are where the information from the IdP gets consumed and decrypted for the Service Provider. Generally, use SAML 2 HTTP-POST for most cases. The Location for this will be in the form of 'https://your.host.duke.edu/Shibboleth.sso/SAML2/POST'. If you have more than one ACS, choose a default if you have a dependency on one

Binding

Location

Is default

Remove this assertion

Add an assertion
Shibboleth can pass directory attributes at the time of authentication for the purpose of convenience and/or authorization.

All Shibboleth instances are encouraged to implement some form of authorization (eligibility or access controls) rather than simply using NetID authentication as authorization.

You may request public attributes to be passed by Shibboleth here. Anyone requesting these attributes is responsible for safeguarding the data. These attributes may not be re-purposed or distributed to any 3rd party outside the application itself.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Value</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>eduPersonScopedAffiliation</td>
<td>Any</td>
<td><a href="mailto:staff@duke.edu">staff@duke.edu</a>, <a href="mailto:student@duke.edu">student@duke.edu</a>, <a href="mailto:faculty@duke.edu">faculty@duke.edu</a>, <a href="mailto:alumni@duke.edu">alumni@duke.edu</a>, <a href="mailto:emeritus@duke.edu">emeritus@duke.edu</a></td>
</tr>
<tr>
<td>eduPersonPrincipalName</td>
<td>Any</td>
<td><a href="mailto:noid@duke.edu">noid@duke.edu</a></td>
</tr>
</tbody>
</table>

**Add an attribute**

[Create]
Update the paths and names of these elements.

Ingredients:

- The path to your private key/cert pair.
- Duke metadata from [https://shib.oit.duke.edu/duke-metadata-2-signed.xml](https://shib.oit.duke.edu/duke-metadata-2-signed.xml) and put in `/etc/shibboleth`.

The location of files is based on the Duke distribution of the Shibboleth SP 2. Your location might vary depending on your installation and storage preferences.

<table>
<thead>
<tr>
<th>Entity Id</th>
<th><a href="https://pki-dev-03.oit.duke.edu/shibboleth">https://pki-dev-03.oit.duke.edu/shibboleth</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>pki-dev-03.oit.duke.edu</td>
</tr>
<tr>
<td>Key Path</td>
<td>/etc/shibboleth/sp-key.pem</td>
</tr>
<tr>
<td>Cert Path</td>
<td>/etc/shibboleth/sp-cert.pem</td>
</tr>
<tr>
<td>Support Contact</td>
<td></td>
</tr>
</tbody>
</table>

You probably don't need to touch any of the following values.

<table>
<thead>
<tr>
<th>Metadata Provider</th>
<th><a href="https://shib.oit.duke.edu/duke-metadata-2-signed.xml">https://shib.oit.duke.edu/duke-metadata-2-signed.xml</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>/etc/shibboleth/duke-metadata-2-signed.xml</td>
</tr>
<tr>
<td>IDP Sign Cert</td>
<td><a href="#">idp-signing.crt</a></td>
</tr>
</tbody>
</table>

Create
<SPConfig xmlns="urn:mace:shibboleth:2.0:native:sp:config">
  xmlns:conf="urn:mace:shibboleth:2.0:native:sp:config"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:samp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:ns="urn:oasis:names:tc:SAML:2.0:metadata"
  logger="syslog:logger" clockSkew="180">
  <!-- The OutOfProcess section contains properties affecting the shibd daemon. -->
  <OutOfProcess logger="shibd.logger">
    <!-- Extensions -->
    <Library path='odbc-store.so' fatal="true"/>
    </Extensions>
  </OutOfProcess>

  <!-- The InProcess section contains settings affecting web server modules/filters. -->
  <InProcess logger="native.logger">
    <ISAPI normalizeRequest="true" safeHeaderNames="true">
      <!-- Maps IIS instance ID values to the host schema/name/port. The name is required so that the proper <Host> in the request map above is found without having to cover every possible DNS/IP combination the user might enter. -->
      <Site id="1" name='sp.example.org'/>
      <!-- When the port and scheme are omitted, the HTTP request's port and scheme are used. If these are wrong because of virtualisation, they can be explicitly set here to ensure proper redirect generation. -->
      <Site id="42" name='virtual.example.org' scheme='https' port='443'/>
    </ISAPI>
  </InProcess>

  <!-- Only one listener can be defined, to connect in-process modules to shibd. -->
  <UnixListener address='shibd.sock'/>
  <!-- <TCPListener address='127.0.0.1' port='12145' acl='127.0.0.1'/> -->

  <!-- This set of components stores sessions and other persistent data in daemon memory. -->
  <StorageService type="Memory" id="mem" cleanupInterval="900"/>
  <SessionCache type="StorageService" StorageService="mem" cacheTimeOut="1600" inprocTimeOut="900" cleanupInterval="900"/>
  <ReplayCache StorageService="mem"/>
  <ArtifactMap artifactTTL="180"/>

  <!-- This set of components stores sessions and other persistent data in an ODBC database. -->
</SPConfig>
SP Infrastructure (cont’d)

- SP owners can register/deregister entirely via self-service in most cases; many can get their entire SP config from the tool.
- Default ARP passes ePPN and ePA (for unfederated SPs); handful of attributes considered “public” can be added in real-time through the self-service interface; others require approval.
- If a registration involves restricted attributes, the request is processed without, then when approved, ARP is updated.
- Note: This ONLY applies to local, not InCommon, SPs.
Questions
Contacting the Authors:

Brendan Bellina, USC
bbellina@usc.edu

Russell Beall, USC
beall@usc.edu

Scott Cantor, OSU
cantor.2@osu.edu

Rob Carter, Duke
robert.carter@duke.edu