Using Shibboleth to protect and access applications

Nate Klingenstein
Internet2
ndk@internet2.edu

国立情報学研究所

２００７年６月１日

This Session

- Accessing “Shibbolized” and SAML-enabled services already set up
- Setting up existing Shibboleth-enabled applications
- Using federated identity for applications that you made, or those not enabled yet

Accessing Existing Shibbolized Resources

- Once you have an IdP, this process is very technically easy
  1. Download the SP's metadata (remember, the file that describes a provider and the keys they use)
  2. Give the SP your metadata
  3. Load the SP's metadata into your IdP
  4. Set up ARP's (attribute release policies) for the SP to send the right attributes
- If you're in a federation, steps 1-3 are already done

Accessing Existing Shibbolized Resources

- Attributes needed vary by resource
- Many resources have their own privilege they want asserted
  - Because “who’s really a student” is too hard a question
  - Because contracts have lots of variety

Existing Shibboleth-Enabled Resources

- Let's look at some examples
  - Many, many more, including Blackwell-Synergy, EBSCO, JSTOR, Thomson Gale, Proquest, Ovid, etc.
- EZProxy has its own Shibboleth implementation too
  - Means many others enabled indirectly

ScienceDirect

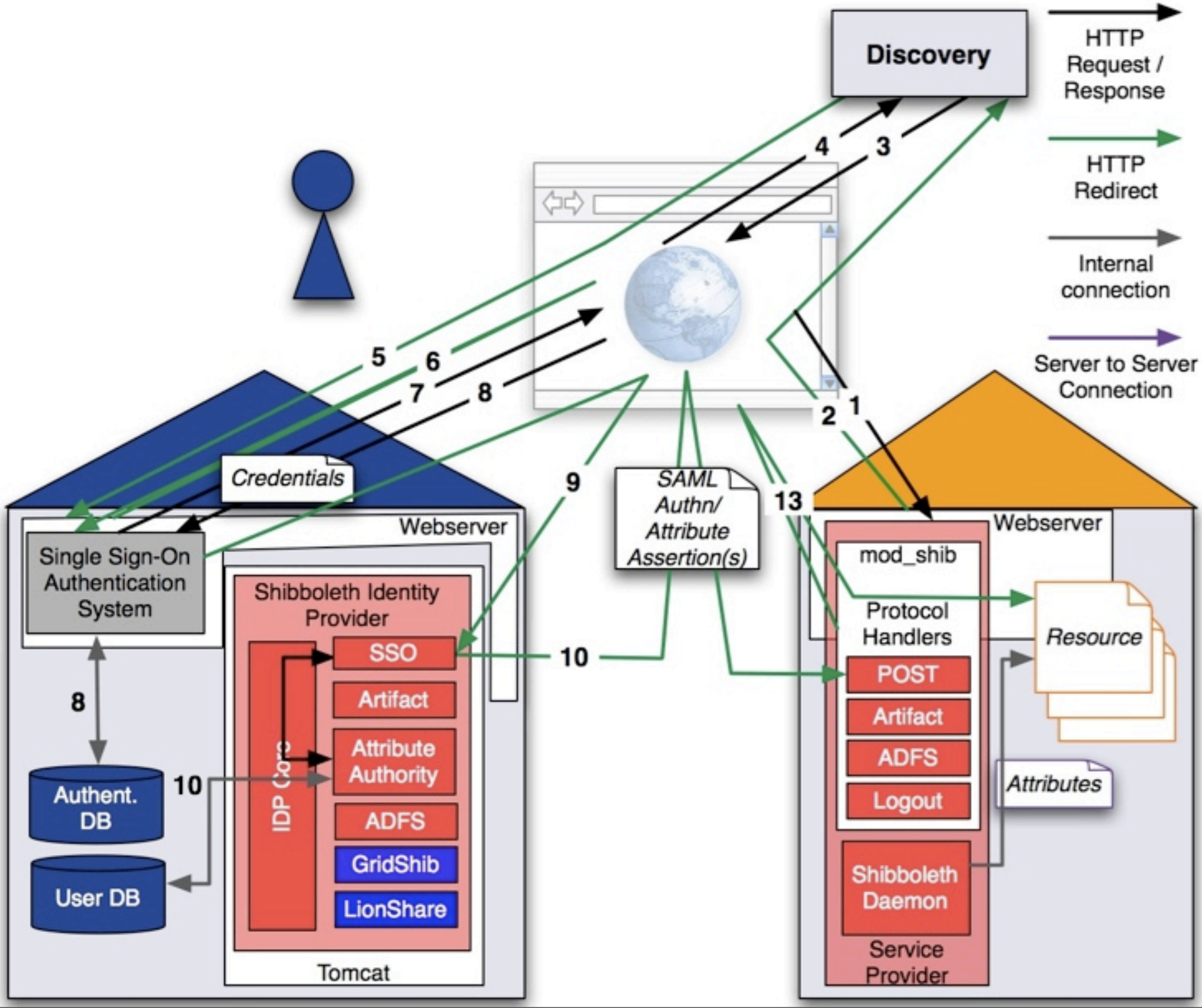
- Member of InCommon (USA), SWITCHaai(Switzerland), HEAL-Link (aai) (Greece), etc.
- Bilateral agreements with most of France, several in the UK
- Requires an eduPersonEntitlement
  - urn:mace:dir:entitlement:common-lib-terms
- Accepts a persistentID for user customization
- http://www.info.sciencedirect.com/implementing/faq/

Microsoft MSDN Academic Alliance

- Download Microsoft software for labs, classrooms, and student PC's
- Member of multiple federations
- Required attributes;
  - User's unique ID, usually as eduPersonPrincipalName
  - Home organization (automatic)
  - Home organization type (could be automatic)
  - User's affiliation (desired)

OCLC

- Many years of Shibboleth pilot, but finally is used in production
- InCommon is their initial production partner
- http://www.oclc.org/productworks/shibboleth.htm


Discovery
HTTP Request / Response
HTTP Redirect
Internal connection
Server to Server Connection
4
3
5
6
7
8
2
1
Credentials
9
SAML Authn/ Attribute Assertion(s)
13
Webserver
Single Sign-On Authentication System
Shibboleth Identity Provider
SSO
Artifact
Attribute Authority
ADFS
GridShib
LionShare
IDP Core
Tomcat
8
10
Authent. DB
User DB
10
Webserver
mod_shib
Protocol Handlers
POST
Artifact
ADFS
Logout
Shibboleth Daemon
Service Provider
Resource
Attributes

Setting up new Shibbolized resources

- Applications either implement SP functionality or use the SP itself
  - Moodle
  - Blackboard/WebCT
  - Ex Libris MetaLib
  - WebAssign
  - Confluence Wiki
  - Zope + Plone
  - etc.

Setting up new Shibbolized resources

- Applications that are already Shibbolized are very easy to set up with Shibboleth
- Shibbolizing applications that haven't been done yet can be very easy, or very hard
  - Open-source tends to be pretty easy
  - Closed-source depends on the vendor and software architecture

Accounts

- There is no technical need for accounts at the SP
  - All authentication and attribute information can be held at the IdP
- But often, some account is needed
  - Information that the SP's organization maintains
  - Applications that require information the IdP won't provide
  - Applications that expect control

SP Accounts in Federated Identity

- Keyed by a unique identifier sent by the IdP
  - SP doesn't need to authenticate the user
  - But still can maintain an account
- Usually done by the application itself
  - Wiki's, WebCT/other CMS, etc.
- Allows for bookmarks, preferences, and other application-specific or SP-provided data to be used

Service-Side Account Example Flow

1. User wants to change a Wiki page
2. Wiki decides user needs authentication, and asks the user to go home to login
3. User authenticates at home, and returns to the Wiki with a uniqueID
4. Wiki matches the uniqueID to an account it has and logs in the user

- What are common problems?
  - Account creation
  - Identifier mapping

Federated Identity vs. Identity Federation

- Confused? It's our bad English
- Identity federation is a way of linking accounts at two providers
  - An SP becomes an IdP, and vice versa
  - ID-FF 1.1, SAML 2.0, ID-FF 2.0
- Arrows in each direction are made
  - Each is unique to SP + IdP + User
  - Just like targetedId/persistentId

Account Linking

- Possible (and done often) using SAML 1.1 / eduPersonTargetedID in an ID-FF-style technique
- SAML 2.0 standardizes this, with many features beyond what SAML 1.1 and eptID could do
  - NameID Management
  - NameID Mapping

SP Accounts in Federated Identity

- Can also be done by the use of an IdP proxy
  - An IdP with accounts maintained by the SP organization
- Same architecture can be used to create virtual organizations (VO's)
  - VO handles privileges, specific attributes, etc.
  - Home organizations handle names, identifiers, and authentication

IdP Proxying Example Flow

1. User accesses service
2. Service redirects user to IdP proxy, which asks where the user's from
3. User is redirected to home IdP, and logs in
4. Home IdP sends attributes & authentication to IdP Proxy
5. IdP Proxy creates a new assertion for the SP and sends the user there

Third Parties

- Many forms already on campus; when it's only at home, it's just provisioning
  - Data Warehousing
  - Central Directories/Databases
- Proxies
  - Because there are NAT's for IP
- Portals
- Attribute aggregation
- Delegation
- Client issuance
  - Provider/User Agent Convergence
- Scope vs. Issuer

Saving and Destroying Information

- Information is inevitably destroyed
  - Where did this attribute start?
  - How did it get to me?
  - Who was trusted as it got to me?
  - What else does this data depend on?
    - Successful user authentication
    - Successful server authentication
- Privacy and secrecy vs. knowledge
  - Your needs will change, but you should know how much you know

Enabling your Own Applications

- Shibboleth SP has no API; it instead puts attributes in standard places
  - HTTP Header Variables
  - Apache Subprocess Environment Variables
  - Attributes on the Java HttpSession object
- All attributes can be named anything you want
  - Extra attributes like IdP name also always available

Shibboleth built-in access control

- Two choices
  - Web server rules
    - require valid-user
    - require affiliation staff
  - XML-based access control in the <RequestMap>
    - This slide is too small
- Applications can make their own decisions too

Protecting Things using your Application

- Which applications can receive which attributes can be restricted
- If you want to use the raw SAML, you can
- Many examples are available for all scripting languages
  - PHP, ColdFusion, JavaScript, Perl, etc.

Shibboleth SP Software

- Apache module for Apache 1.3.x, 2.0.x, and 2.2.x
- ISAPI Filter for Microsoft IIS 5+
- All versions have an attribute querying daemon
  - shibd
- Much work was put into the separation so the web server would never have the SP's private key

Shibboleth SP Software

- However, Apache can connect to most other environments
  - Especially Tomcat via mod_jk, mod_jk2, or mod_ajp_proxy
- And many vendor products are Apache-based
  - Those that aren't almost all support SAML natively

Enabling Applications

- English saying: “You can bring a horse to water, but you can’t make it drink.”
- The same is true of web-based applications and attributes
  - You can almost always give attributes to an application
  - Whether it will use them is different

Practical Approach to Shibbolizing a System

1. Learn who needs to know what, who can say what, and what can't be said
   - Metadata can help
2. Decide on protocols & bindings
   - Shibboleth makes this easy
3. Check whether someone has already defined the attributes you need
4. If so, use them; if not, choose wise names and values, and write them down
5. Create the new attributes if they don't exist; set release and access control policies

Basic Example

- A store wants to sell discount books and school shirts to university students
  - Who, exactly, is a student?
    - How precisely do you care?
- The university and store collaborate to craft the trust agreement
- If eduPersonScopedAffiliation isn't good enough, http://www.cheapbooks.edu/attributes/ourstudent or an eduPersonEntitlement
  - The university provisions the attribute to eligible users
- Attribute information is released to the store, which maintains attribute-based access control
  - Beats accounts and IP Addresses

Basic Example

- System of record: SIS
- Attributes needed: eduPersonScopedAffiliation
- Other information needed:
  - Check issuer against attribute scope so OSU can’t buy Florida shirts?
- Access control rule:
  - require scopedaffiliation *.edu

Always remember to:

- Attribute-enable applications
- Be pragmatic and trusting
  - Because it's easy to audit and punish
- The fewer total attributes in the world, the more powerful federated identity is
  - Recycle, reduce, re-use
- Name everything properly
- Use strings whenever possible
  - Applications and people seem to like them
- Keep flows as simple as possible