Shibboleth/SAML: Info & Flows

by Marlena Erdos, using materials from herself & inspiration from a presentation by Marcus Mizushima, Gabriel Sroka, Gay France, Nate Klingenstein and unknown Internet2 personnel

Shibb/SAML: Raison d'etre

- Context: A protected website, accessed by users at various institutions, e.g. NIH website, and research labs (@Harvard, etc)
  - Old way: All users register at the site
  - Old way: All users login locally to the site
  - Recent example: IT Summit website

- User view: Too many distinct logins
- Resource Admin view: Too many foreign users
  - Maintain id/pwds for local plus foreign users
    - User population grows with each new partner
  - Never know when to deprovision foreign users

- Shib/SAML Solution: Users login at home institution
  - Institutions trust each other about their users

Shibb vs SAML

- Shibboleth: Code that implements SAML
- **S**ecurity **A**ssertions **M**arkup **L**anguage
  - A secure request/response protocol for
    - Authentication
    - Attributes
    - Authorization (but I don't know about this part:-))
  - An set of XML formats for the request/response
    - "Assertion" carries the info about the user
- Shibb adds on attribute management to SAML
  - A hugely important feature!

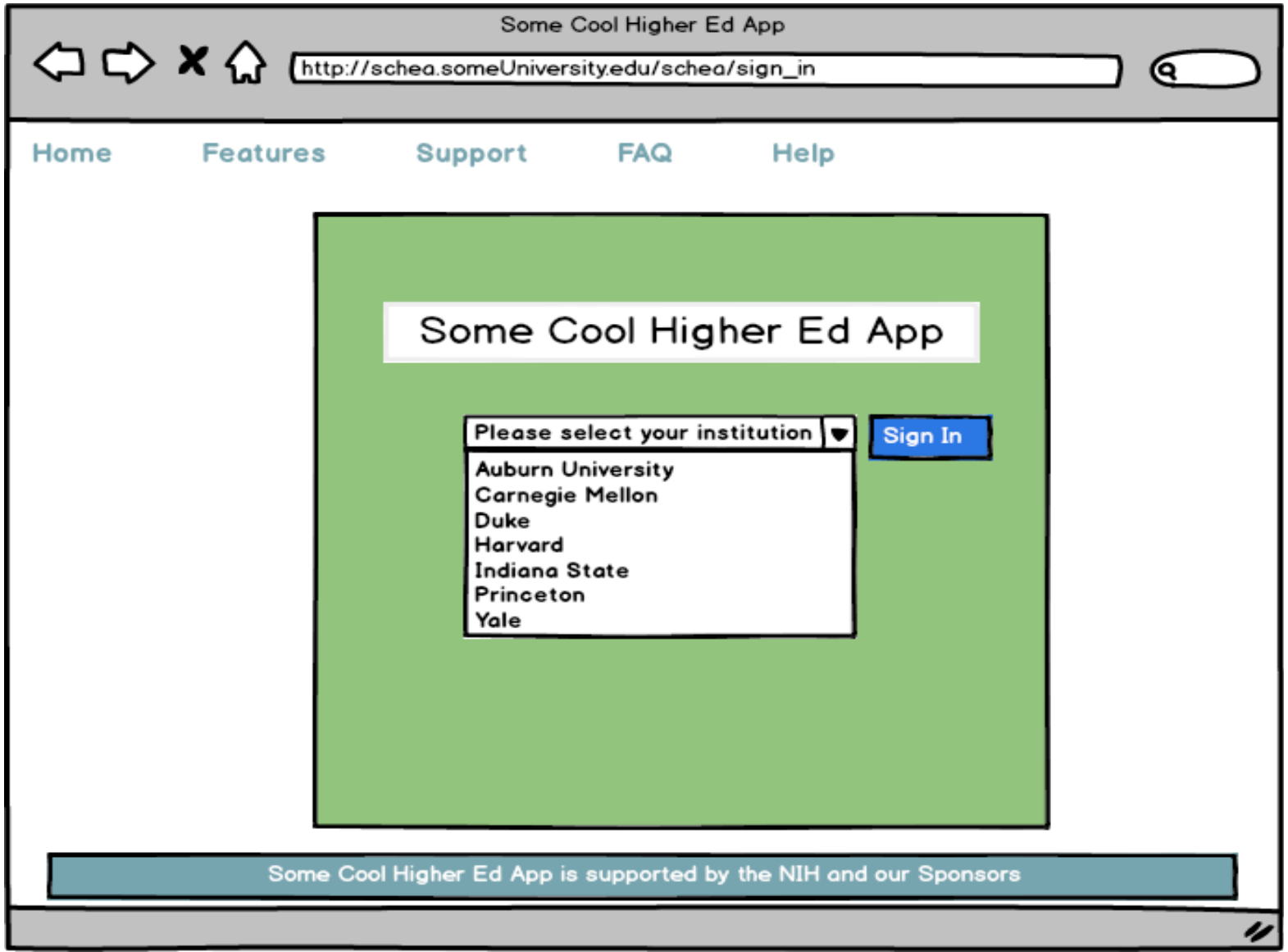
Outline

- SAML/Shib: Info and Flows (overview-y)
- SAML/Shib flows: Terms and Detailed flows
- Novel angle on SAML/Shib and PIN
- Attribute management in Shib (briefly)

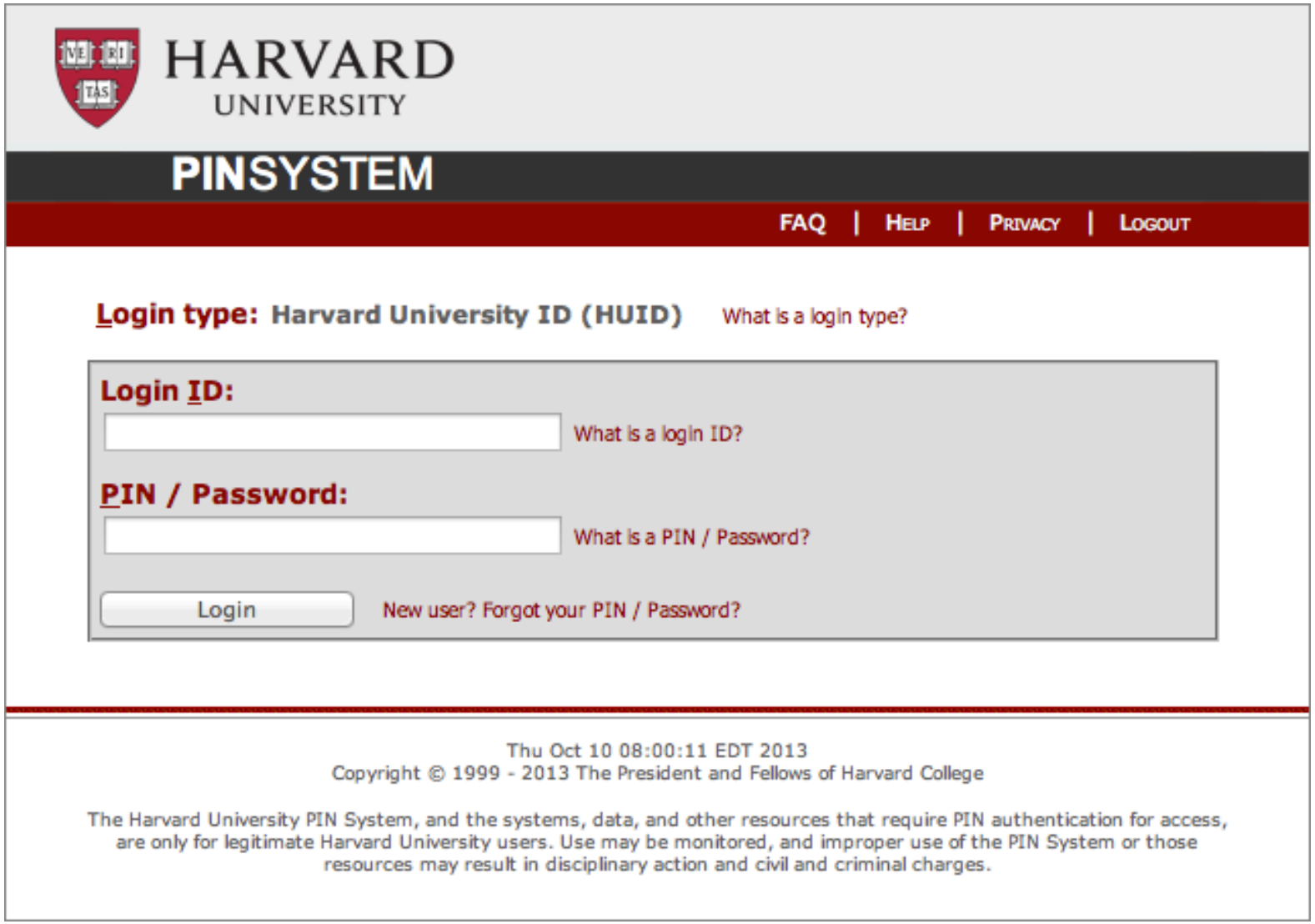
Shib Flow: User View

- The user tries to access a protected app
- App asks user “where are you from?”
- User answers
- The user sees the “home” login screen
- User provides login name & password
- User may get access to the app (or may not)

Discovery Service aka “Where Are You From?” screen example



After ”WAYF” (aka “Discovery Service”) then the good ol’ PIN login



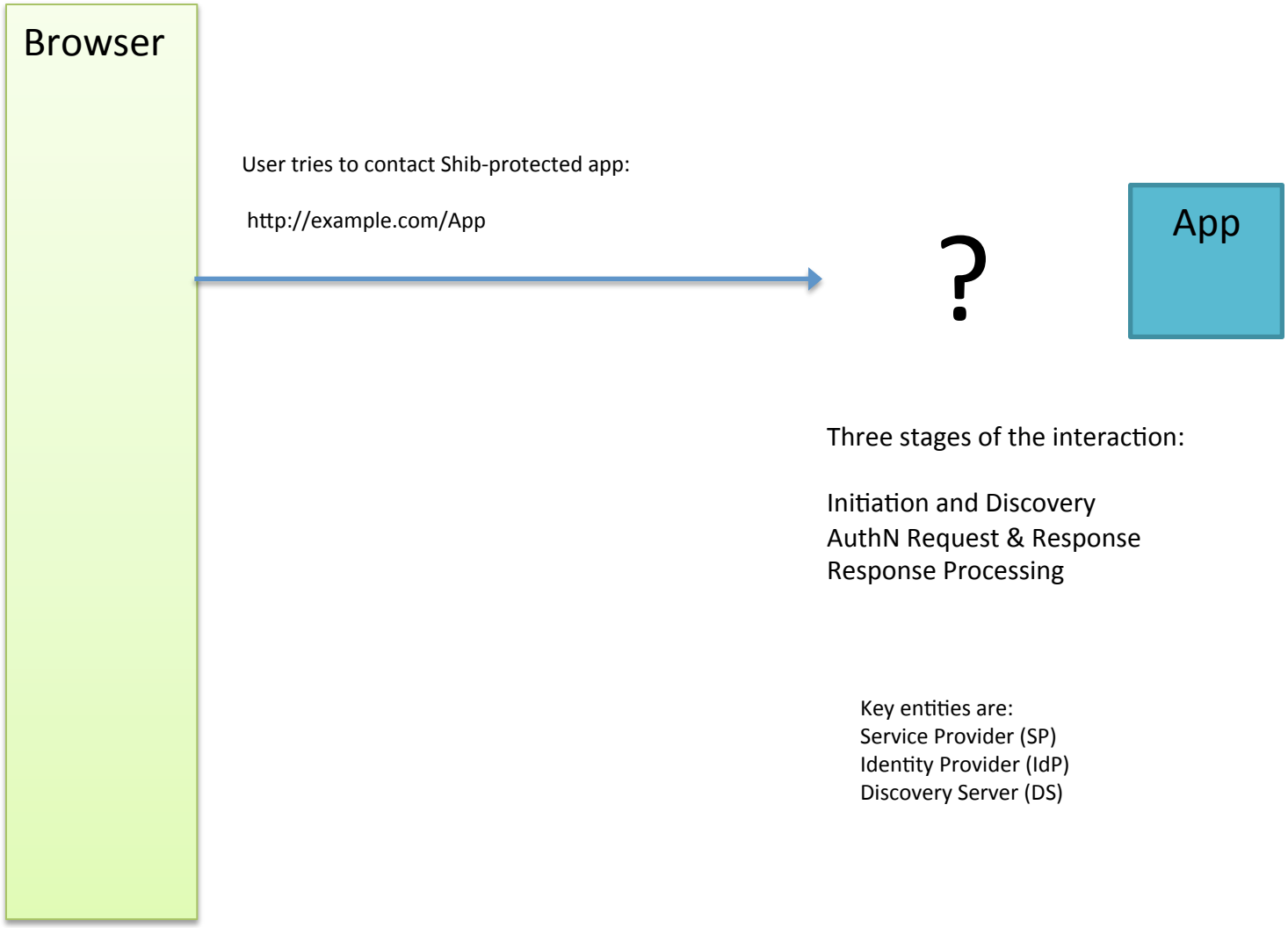
Outline

- SAML/Shib: Info and Flows (overview-y)
- **SAML/Shib flows: Terms and Detailed flows**
- Novel angle on SAML/Shib and PIN
- Attribute management in Shibb (briefly)

SAML Terms

- Service Provider (SP) – Makes authN requests on behalf of an app being accessed by a user
- Identity Provider (IdP) – provides SAML authN responses
  - Response contains an "assertion"
  - Assertion contains attributes about the user
  - IdP's digital signature on the assertion or response
- Discovery Service: Helps SPs find IdPs
- Entity ID: "Name" for each SP and IdP
  - Looks like url but isn't one
  - e.g. https://fed.huit.harvard.edu/idp

Shibboleth Detailed Flows (in four slides)



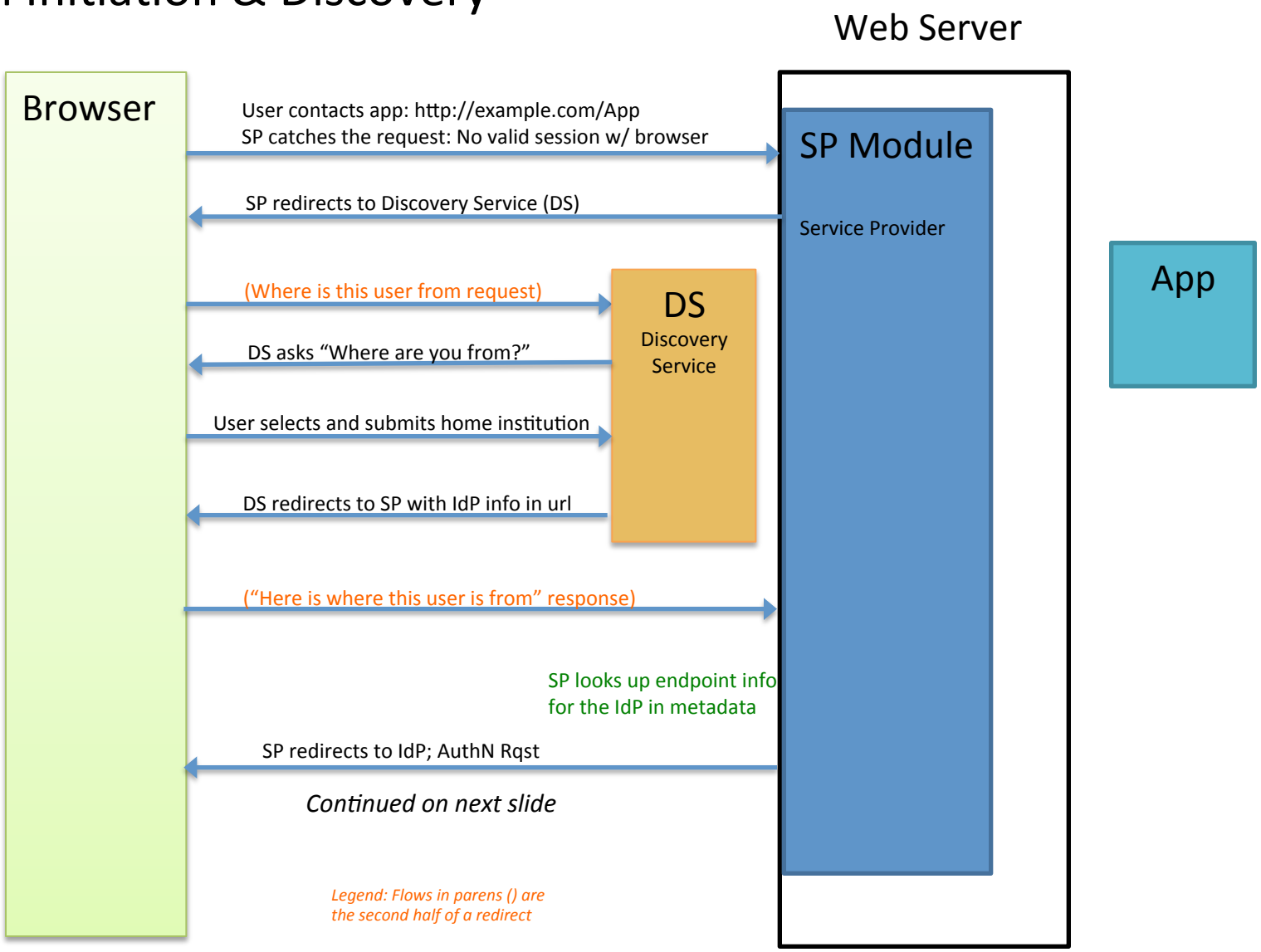
Three stages of the interaction:

Initiation and Discovery
AuthN Request & Response
Response Processing

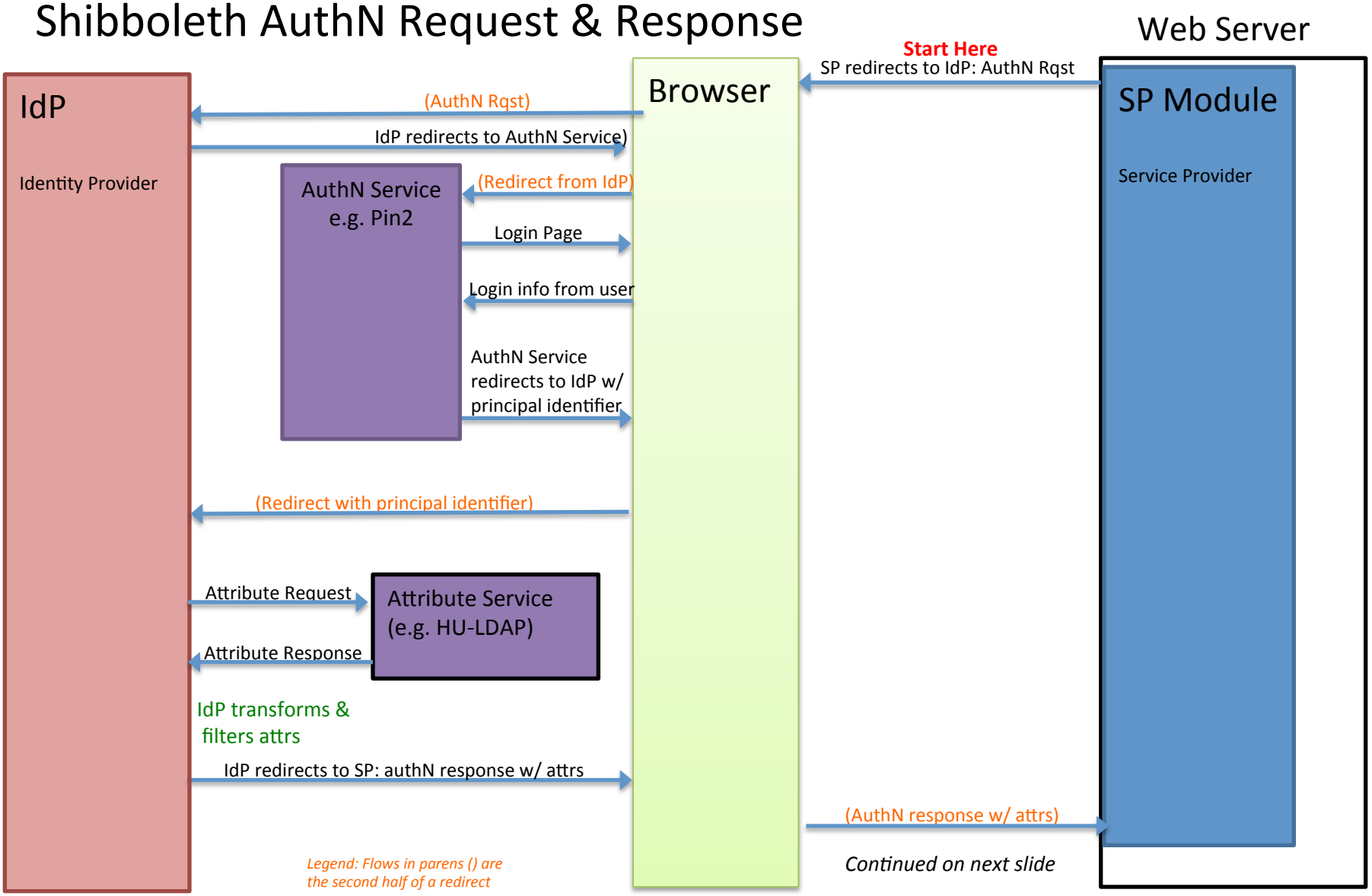
Key entities are:
Service Provider (SP)
Identity Provider (IdP)
Discovery Server (DS)

By Marlena Erdos with partial inspiration from Chris Bongaarts

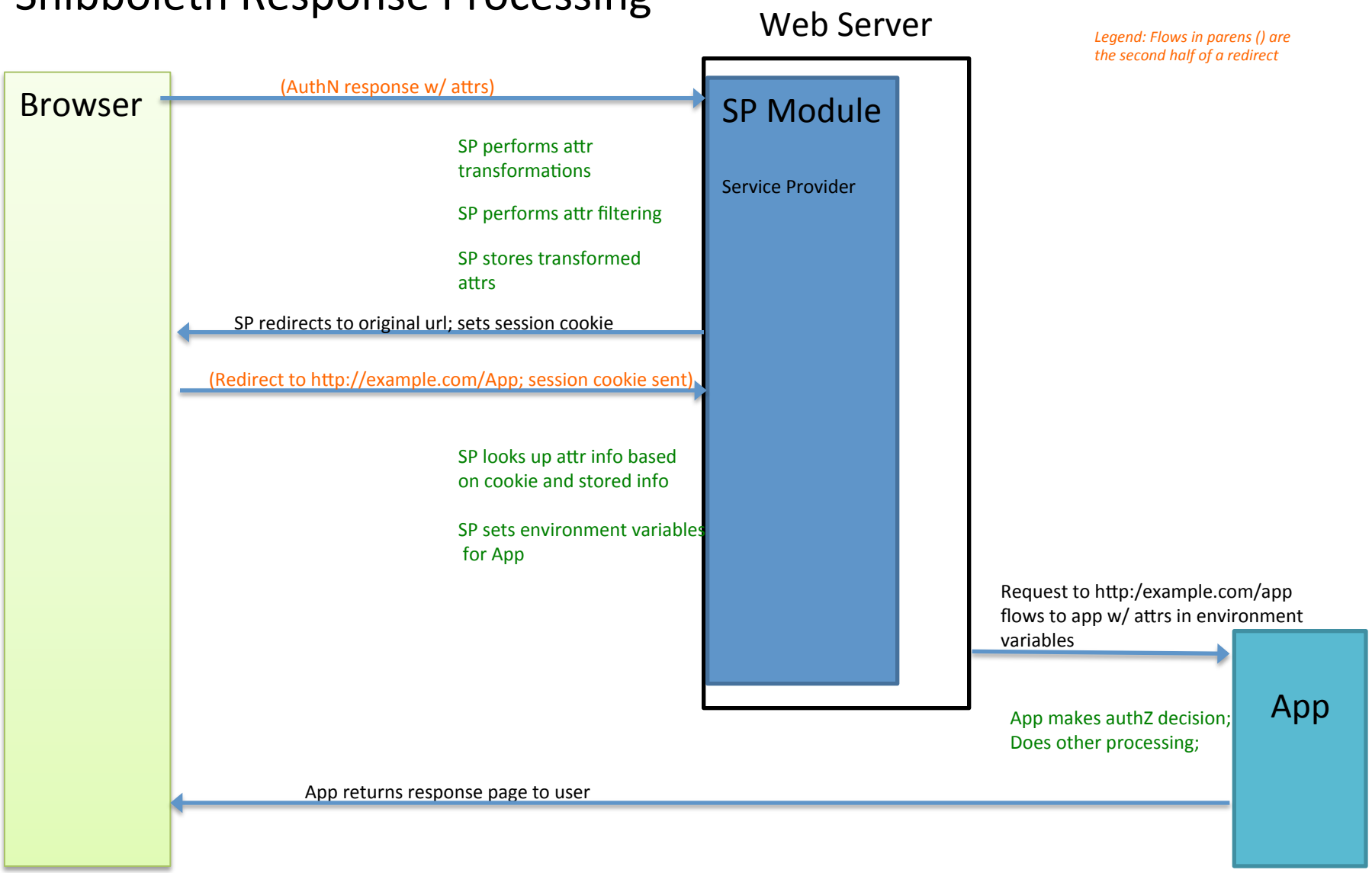
Shibboleth Initiation & Discovery



Shibboleth AuthN Request & Response



Shibboleth Response Processing

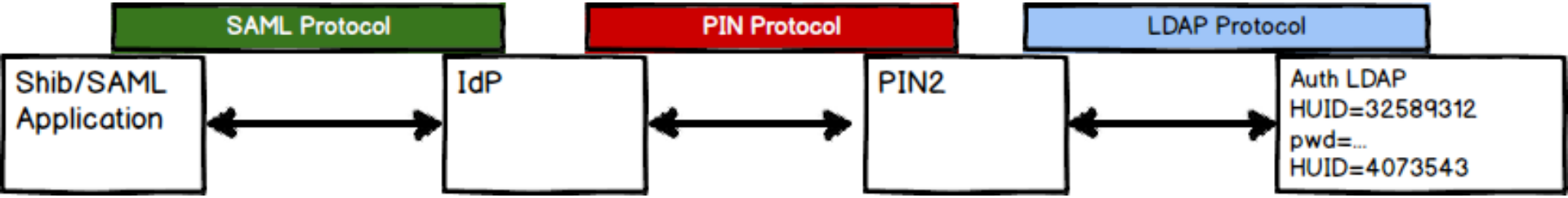


Outline

- SAML/Shib: Info and Flows (overview-y)
- SAML/Shib flows: Detailed flows
- **Novel angle on SAML/Shib and PIN**
- Attribute management in Shib (briefly)

PIN protocol as a ‘gateway’ or ‘layer’ over LDAP

SAML protocol as a ‘gateway’ or ‘layer’ over PIN …which is a gateway or layer over LDAP



Outline

- SAML/Shib: Info and Flows (overview-y)
- SAML/Shib flows: Detailed flows
- Novel angle: Both PIN and the IdP as ‘protocol gateways’ over a password repository
- **Attribute management in Shibb (overview)**

Attribute Discussion

- Attribute == a piece of information about a user
  - Examples: email address, department, start date
  - Identified by an Object ID/URN
  - Zero or more values
- IdP attribute handling
  - Retrieves attributes from configured repositories
  - Transforms input attrs into output attrs
  - Filters what gets sent to a given SP
- SP attribute handling
  - Assertion is the attr “repository”
  - Transforms and filters attributes
  - Creates env or header variables for application

How the IdP Retrieves Attributes

- Retrieval via a “data connector” definition in an IdP config file
  - config file == “attribute_resolver.xml”
- The IdP can easily be configured to retrieve attributes from LDAP directories and relational databases (and more)

Config-let for LDAP Repo

```
<resolver:DataConnector id="HULDAP" xsi:type="dc:LDAPDirectory”

    ldapURL="ldaps://hu-ldap-test.harvard.edu:636"
    baseDN="ou=people,o=Harvard University Core,dc=huid,… "

    principal ="uid=shibbidp,ou=applications,o=Harvard University …. "
    principalCredential=”NoneAUrBizNess”

    <dc:FilterTemplate>
       <![CDATA[
          (harvardeduidnumber=${requestContext.principalName})
       ]]>
    </dc:FilterTemplate>

 </resolver:DataConnector>
```

Attribute Definition

Attribute definitions allow you to

- map a source attribute (SA) into a output attr (OA)
  - e.g. “email” -> “mail”
- use the value of an SA to create a new value for an (OA)
- tell the IdP how to encode the value for transport

Attribute Definition

<resolver:AttributeDefinition xsi:type=*"Mapped" id="isStudent"* ***sourceAttributeID****="harvardedustudentstatus">*

<**resolver:Dependency ref=*"HULDAP*" />

<ad:DefaultValue>false</ad:DefaultValue>

<!-- R==Registered A==Active Class Participant F==On Leave paying fee -->

<ad:ValueMap>

<ad:ReturnValue>true</ad:ReturnValue>

<ad:SourceValue>R</ad:SourceValue>

<ad:SourceValue>A</ad:SourceValue>

<ad:SourceValue>F</ad:SourceValue>

</ad:ValueMap>

Attribute Filtering

Config file == attribute_filter.xml

Controls release of attributes in the current assertion by

- SP (i.e. recipient)
- attribute value
- User being authenticated

Syntax is powerful but a bit painful (and so not shown)

Examples to cut/paste from are on the Shibb Wiki site

https://wiki.shibboleth.net/confluence/display/SHIB2/IdPAddAttributeFilter

SP Attribute Managment

IdP resolver file => SP attribute_map.xml

IdP filter file => SP attribute_policy.xml

Map: Transforms assertion attrs into output attrs

Policy: controls what attrs get put into env variables

E.g. excise “bad” attributes

- Harvard IdP saying user is “faculty@yale.edu”

Resources URLs

IdP Home page for info: http://iam.harvard.edu/resources/idp-guide

Contains Policy Info, “Support for Support,” FAQ, and Shibboleth Flows.

The “Support” and FAQ are very much in-progress so please send us suggestions for improvements (via iam@harvard.edu).

THANK YOU!!!

What's InCommon?

InCommon is a collection ("federation") of US higher education institutions and research institutes that have agreed to cooperate with each other according to a set of rules.

More/better info here:

http://iam.harvard.edu/resources/incommon