Suggestions for Implementers

Suggestions for Implementers

Defining the Problem, and Identifying Approaches

Increasingly, groups and people operating web sites on campuses want their site to be used by members of their campus community, by people from other campuses, and by people from outside the Higher Education/Research environment. Central IT Departments have provided authentication mechanisms that address the first two groups. The last group is often addressed by adding user management (self-service?) and authentication to the application; however, more recently, web sites have moved toward relying on social identity (provided by the big Internet providers). Web site developers and deployers are now looking for ways that their applications can support both enterprise and social identities.

1. Perspectives on Handling Both Social and SAML Identities

What are each of the parties hoping that an approach, a framework, will provide for them ?

- Relying party wants to make some federated resource accessible to people who have a Social Identity Provider (Twitter, Yahoo, Google, Windows Live)
    - On one hand they want to minimize changes to their Service Provider implementation and their application code
    - On the other, they want to know which ~~and what type of~~ IdP is handling any particular access instance
    - They want the information about an authentication event presented to them in a standard format, independent of the protocol that was used. If they want identity attributes as well as an authentication assertion, they want those attributes to have consistent names and consistent value syntax. The same attribute name should not represent two different attributes nor should the value syntax vary for a given named attribute. Different implementations providing social identity support should use standardized syntax and semantics for all the provided attributes.
        - This applies whether a Social-to-SAML gateway is involved or not.
    - They do not want to be bothered with designing and implementing a complex Discovery process that can accommodate multiple protocols. They would rather hand this problem off to a gateway. However, this also offloads part of the problem onto the user experience.
    - They want the application to contain a mechanism that allows a current user to "invite" a new participant to join the site, independent of the type of identity that individual might eventually use to authenticate to the site.
    - They want the application to contain a mechanism that allows a new interested individual to join the site, or apply to join the site.

- The user expects to make a search-and-one-click selection of their IdP of choice
    - They would like to see a given social identity provider identified the same way regardless of their path to the SP
    - They might expect that they would be recognized as the same individual regardless of their choice of IdP, but in general this is not possible without some user mediated account linking on a per-SP basis

1.2 Models for Integration

- A native SP implementation supporting both SAML and Social protocols that deployers could add to their site.
    - unfortunately, such an implementation does not exist (really? simplesamlphp ?)
    - Currently, there is no single "standard" protocol used by the various social identity providers. They seem to use proprietary protocols or proprietary variants of OpenID. Change happens quickly, and is outside the normal standards processes. This churn means that groups that maintain native SP implementations will often wait for some level of consensus before providing support for a newer protocol.
    - Going forward, there seems to be broad interest in both oAuth V2 and OpenID Connect. If broad deployment and consensus appear, native SP implementations may add support.
- A central gateway, operated by a Federation, that translates incoming protocol requests to a single standard protocol which is supported by all of the SPs in the Federation.
    - unfortunately, due to the frequent churn in the protocols used by the social sites (and the fact that some of these protocols are proprietary), no Federation is willing to run such a gateway (and have to deal with the long term support of such a GW). (Actually, no -- FEIDE is doing this with Roland's GW)
- a gateway operated by a campus that translates incoming protocol requests to a single standard protocol which is supported by all of the SPs on the campus. It would be the campus' decision as to whether to limit the use of such a gateway to SPs located on the campus.
- a gateway installed by, maintained by, and operated by the SP site that supports translating incoming protocol requests to a single standard protocol which is supported by the application.
    - note, though, that the reason SP operators have been looking to use gateways is that they want to offload the responsibility for the Discovery process to the gateway.

1.3 Gateways as Necessary Evils--For a Time

- The majority of current identity federation deployments connect SAML IdPs with SAML Relying Parties (RPs, SPs)
- A growing number of R&E organizations wish to make some of their services accessible to users who prefer to use Social Identity Providers (Twitter, Yahoo, Google, Windows Live, Facebook). Some of these users will in fact not have an R&E organizational SAML identity.
- R&E application owners do not want to modify each SAML-protected SP to handle social identities

- As a result, many have been attracted to the idea of a Social-to-SAML gateway that permits a user to authenticate with their social IdP of choice and access a SAML relying party application or service
- The Social-to-SAML gateway is often designed to take the role of SAML IdP in interactions with RPs. This is driven by the application owner constraint that the solution not involve major modifications of existing RPs and SPs
- So a key function of the Gateway is to map social IdP asserted authentication events and identity attributes to a corresponding SAML assertion for consumption by the RP.

1.4 The Proposed Model

-- the SP will have to include support for Discovery

-- by configuring their Discovery mechanism, the SP decides which social providers to allow and trust. if the SP wants to exclude facebook, then the GW has to be able to enforce that policy choice....

-- the SP would also configure the address for the social-to-SAML gateway that they choose to use.

-- the browser user would select their identity Provider (eg a campus, google, yahoo, facebook, etc), and click SUBMIT.

-- the user would be redirected to the gateway; the SP would send an entityID value that identifies the requested social IDP)

-- the browser user would be redirected on to the social provider, authenticate, and be returned to the gateway.

-- the gateway would construct a response to the SPs AuthnRequest, using the guidelines described below.

-- the gateway would issue a POST back to the SP, using a standard SAML flow.

NOTE -- in this model, the user does not see the gateway ever present a GUI

1.5 Basic Gateway Usage Model

1. The browser user would select their identity Provider (eg a campus, google, yahoo, facebook, etc), and click SUBMIT.
2. The user would be redirected to the gateway; the SP would send an entityID value that identifies the requested social IDP)
3. The browser user would be redirected on to the social provider, authenticate, and be returned to the gateway.
4. The gateway would construct a response to the SPs AuthnRequest, using the guidelines described below.
5. The gateway would issue a POST back to the SP, using a standard SAML flow.

NOTE -- in this model, the user does not see the gateway ever present a GUI

Case Studies

- Penn State OpenId Implementation - mod_auth_openid wrapped around a Shibboleth IDP
- Implementation Descriptions

2. Developers

This section is intended to be a cookbook-like document for people developing applications and who want to allow authentication from social identity providers (eg the Bamboo project)

Suggestions

1. Isolate your application from the authentication protocols.
2. Choose an authentication package implementation that supports standard attribute conventions. See TN - Conventions on Attributes
3. The application would need to know both 1) the identity of the social identity provider, and 2) the identity of the gateway which is forwarding the authentication event in order to determine whether or not to trust the presented Assertion.
4. The SP should include support for the Discovery Process. By configuring their Discovery mechanism, the SP decides which social providers to allow and trust. if the SP wants to exclude facebook, then the GW has to be able to enforce that policy choice....
5. the SP would also configure the address for the social-to-SAML gateway that they choose to use.
6.
7. An application will need to map incoming identities to same internal "person object"

3. Gateway Developers

How to Build an Eventually Dispensable Gateway Service

- The goal is eventually to outgrow the need for gateways by providing native multi-protocol SPs that support both SAML and selected Social IdPs.
- This goal will be easier to achieve if the gateway functions are essentially invisible to the end user. That is, the step in which a user selects an IdP should behave the same way whether or not a Social-to-SAML gateway is involved.
- Unique EntityIDs should be defined for each SocialIdP-Gateway pair. This allows SPs to decide on trust points based on information in SAML2 metadata files if the relevant identity federation supports this.
- Mapping social IdP assertions to SAML assertions is a core gateway function

Gateways Must Honor SP Policies on Acceptability of Various Social Providers

- If an SP does not accept a certain Social IdP (e.g., they opt not to accept Facebook-based authentications), the gateway must honor that policy by not offering the prohibited IdP in the IdP selection list available to the end user.

could define policies at the SP saying who is allowed to assert which kinds of identifier; once you make it thru that layer of filtering than the app can trust it...

eg if you trust GWs X, Y, and Z, and IDPs 1, 2, and 3

4. Deployers

1. Because the SP configures in which GW it wants to use, it doesn't have to worry about people coming in from multiple GWs and using the same social provider.
2. could define policies at the SP saying who is allowed to assert which kinds of identifier; once you make it thru that layer of filtering than the app can trust it...

eg if you trust GWs X, Y, and Z, and IDPs 1, 2, and 3

Policy Issues to Consider

1. After reviewing your goals for your site, and the types of data that it will contain, identify which social identity providers 9and associated protocols) you want to trust.

Gateways Must Honor SP Policies on Acceptability of Various Social Providers

- If an SP does not accept a certain Social IdP (e.g., they opt not to accept Facebook-based authentications), the gateway must honor that policy by not offering the prohibited IdP in the IdP selection list available to the end user.

could define policies at the SP saying who is allowed to assert which kinds of identifier; once you make it thru that layer of filtering than the app can trust it...

eg if you trust GWs X, Y, and Z, and IDPs 1, 2, and 3