Identity and Access Management - Authentication and Authorization

Approach in overview

Assumptions

From planning workshops (April 2008- September 2010), Project Bamboo identified a number of **characteristics of humanities scholarship that point to a need for a broadly-federated and loosely-coupled identity and access management (IAM) infrastructure**. The assumptions that influenced design and implementation of Bamboo's approach to IAM included:

- Faculty, students, and staff are identifiable via ability to authenticate (log in) through their own institution's identity management (IdM) infrastructure.
- Faculty, students, and staff are often eligible for benefits (such as subscription-access to digital content) by virtue of an institutional affiliation.
- Scholars form communities in disciplinary clusters, and also in groups with shared interest in particular objects of scholarship (including digitized or born-digital collections). These communities form across institutional and international boundaries, and may include citizen-scholars who are not formally affiliated with any institution of higher education. Scholars affiliated with universities may *change* affiliation from one institution to another, e.g., when they take a position at a different university from the one where they obtained a degree.
- Products of research, including underlying data and work-in-progress, is shared by scholars in groups whose membership they themselves may wish to manage. Ideally, a scholar has full control over how narrowly or how widely she shares her work.
- Scholars utilize a diverse set of research environments, tools and/or services in the course of their work.
- Credit (attribution) for contribution to scholarly effort is important to faculty and students in academic contexts. Credit requires the ability to identify the actor (user) in any workflow for which attribution is required. Scholarly effort in the sense used here extends beyond formal, peer-reviewed publication; it includes contribution to and curation of digital resources (content) that constitute corpora that are valuable to researchers.
- The more transparently tools are able to effect and be constrained by appropriate access permissions (including intellectual property constraints), track user activity, and cross institutional borders, the less scholars are distracted from their core interests by these administrivia.

By **"broadly-federated" IAM infrastructure**, we mean an infrastructure that allows faculty and students to log in via their home institutions; *and* also permits unaffiliated scholars ('citizen scholars') to use readily-available logins they are likely to maintain already, such as social media identities established with Google, Facebook, and other social media. Use of existing logins – and, in the case of institutional identities, logins that can be trusted, more-or-less, to belong to the faculty member or student who uses the login for access – is generally considered more convenient and reliable than establishment of multiple separate logins for tools or services relevant to the spectrum of a scholar's activity. (It is for this reason that institutions of higher education in the United States have formed the InCommon trust federation to federate the *identity* aspect of *identity and access management*; similar networks have been formed elsewhere, and trust between these national federations is an ongoing, as-yet-unrealized effort. See footnote [a] below for the context in which a *Ba mboo Trust Federation* was posited by this project.)

By **"loosely-coupled" IAM infrastructure**, we meant that the same federated login ought to be applicable to a wide range of research environments, tools, and services, since the set applicable to humanities scholarship is large, diverse, and constantly evolving.

It is worth noting that no previously existing infrastructure fully addressed the goals (described below) that arise from the set of assumptions listed above at the time the Bamboo Technology Project commenced (October 2010).

Goals

1. **Users of applications enabled by the Bamboo IAM infrastructure** should:
    a. be able to log in from their home institution(s); and/or via social media identity providers (IdPs), such as Google.
    b. maintain a constant identity no matter which institution or social media IdP they are using to log in at any given time.
    c. be able to form groups to define with whom they wish to work or wish to share digital content, irrespective of those individuals' institutional affiliations.
2. **Tool and service developers and providers** should be able to integrate with the Bamboo IAM infrastructure with no or little effect or additional development requirements on their tool or service. To this end, they should be required to add *only* well-known, mature, and trusted software libraries to their technology stack.
3. Any **'Bamboo-centric' identity** should be easily associated with a scholar's preferred, globally, nationally, discipline-wide, or institutionally unique identifier (e.g., ORCIDs). Recognizing that there was no single accepted unique identifier at the time Bamboo IAM infrastructure was designed and created, the goal was to enable linkage of any user-chosen identifiers with a *Bamboo Person* profile.

Implemented IAM functionality

- A ***Bamboo Person*** may be created, and its **Bamboo-namespaced identifier** can be associated with **multiple identity providers** (IdPs), such as one or more universities and/or one or more social media IdPs.
- A Bamboo Person proves her identity by authenticating to an associated IdP; **Bamboo's IAM infrastructure does not gain access to user credentials (passwords)**. Where privileges depend on affiliation with an institution, affiliation must be proven by login to that institution's IdP in any given session to prove currency of the affiliation.
- A Bamboo Person may specify an associated set of **profile information**, including links to profile information maintained elsewhere (e.g., on a university's departmental web site).
- Any Bamboo Person may form **groups that are maintained in a central store**, and may add to their groups any other Bamboo People. A Bamboo Person may delegate responsibility for managing her group(s) to other Bamboo People.
- Policies may be defined to effect **access restrictions on resources** provided via centrally-hosted Bamboo services; group memberships may be accessed or verified by tools and services hosted separately, to help determine access permissions/restrictions on resources owned by those separately-hosted tools and services. See *Authorization and Policy* for further context and detail.
- A **research environment, tool, service, or other client may participate** as a trusted member of the Bamboo IAM ecosystem (e.g., trusted to log in users and obtain services on their behalf) by meeting these requirements:
  - **operating as a Shibboleth Service Provider** (Shibboleth SP); and,
  - entering into appropriate **operational agreements** [*these were not yet drafted for Bamboo at the time the project ended, but would have been akin to InCommon policies and practices*]; and,
  - **registering** as a member of the Bamboo Trust Federation by exchanging and publishing appropriate metadata upon satisfaction of the two requirements listed above [a]
- An **Identity Provider (institutional or social) may authenticate Bamboo users** if it meets these requirements:
  - it is a SAML-compliant IdP; or if a Social/SAML Gateway in the Bamboo Trust Federation [a] is configured to convert the IdP's native assertions to SAML assertions; and,
  - if it registers as a member of the Bamboo Trust Federation by exchanging and publishing appropriate metadata [a]; and,
  - if it releases a minimal set of user attributes[b] upon successful authentication

This functionality was implemented with as little new coding as possible. Instead, Project Bamboo sought to integrate extant software libraries and applications, including Apache Web Server (httpd), Shibboleth, Grouper, and SimpleSAMLphp. Where new coding was necessary, adherence to established standards, such as XACML, was prioritized.

[a] A "Bamboo Trust Federation" is exactly analogous to the InCommon Trust Federation that is technically defined by InCommon metadata. It is the set of all (institutional and social) IdPs trusted to authenticate users; and the set of applications trusted as Service Providers. Because only a subset of institutions participating in Project Bamboo were members of InCommon; because InCommon did not define trusted social media identity providers or support login by same; and because flexibility to register Service Providers that could or would not register with InCommon was required; Bamboo proceeded along a path that included instantiation of an independent trust federation. Future developments in federated trust – and/or a different set of service providers, included institutions, and trusted social IdPs -- may allow those who rely on the technology described here to 'piggyback' atop one or more existing trust federations rather than instantiate their own.

[b] As the Bamboo IAM infrastructure was implemented, the "minimal set of user attributes" required of an IdP includes only a user identifier unique to the IdP (a.k.a. a "scoped" user identifier); and a unique identifier of the IdP itself, generally the domain name of an institution or social media provider, such as *berkeley.edu* or *google.com*. A discussion of N-tier authentication and user attribute release in relation to Project Bamboo's work can be found in this documentation on the page *Authentication - Current Limitations and Future Direction*.

✓ It is noteworthy that here and elsewhere within the scope of this documentation the Bamboo 'branding' is used as a matter of convenience. The technology implemented and integrated may be differently-named by adopters as desired, within the requirements imposed by the open-source ECL2 software license that governs all intellectual property produced by Project Bamboo.

Limits to implemented IAM functionality

The *Bamboo Technology Project* was conceived as a three-year software development effort, to be conducted in two phases. Only the first phase was funded. During this first phase, the **underlying services to effect functionality described above were completed**. In addition, test integrations between client applications and IAM infrastructure services, the latter intended to be hosted centrally, were conducted as proofs of functional utility. However, **end-user interfaces were not built out**; thus, provision has not yet been made for 'user-friendly' establishment and maintenance of Bamboo identities, profiles, groups, or policies. Such user-interfaces are left as work to be done by adopters of the Bamboo IAM infrastruture.

It is also noteworthy that while functional utility has been proven, instantiation of the Bamboo IAM infrastructure on servers hosted in production data centers were limited to development instances used for testing and integration purposes. Productionizing these servers, load testing, and documentation of deployment architectures was not undertaken in the funded phase of the project's technology development.

Bamboo IAM from a client application's perspective

Overview

The basic requirements for a client application to participate as a trusted member of the Bamboo IAM ecosystem are described in the *Implemented IAM functionality* section, above.

As of the close of the Bamboo Technology Project, the implemented IAM infrastructure does *not* utilize N-tier authentication, as implementation and uptake of Enhanced Client Protocol (the technology we hoped to use in early stages of the project) was judged insufficiently mature to adopt in the timeframe of the Bamboo Technology Project. Details are discussed in this documentation on the page *Authentication - Current Limitations and Future Direction*.

In overview, the implementation approach used *in lieu of N-tier* requires that client applications be *trusted* to authenticate users and to make true assertions about those users' identities and roles to centrally-hosted services. This is a less-than-ideal approach because it limits use of fine-grained user-attributes in policy decisions. "Fine-grained user attributes" that *could* be supplied by an higher ed institution might include, for example, contact information (e-mail address, phone number, postal address, etc.); or affiliation type (e.g., faculty, student, staff, alumna). Use of such attributes might, for example, permit faculty members to access content licensed by an institution for faculty use but not for use by students or staff.

The reason that Bamboo's approach limits use of these fine-grained attributes is that passing user attributes between independent or loosely-coupled applications / platforms would violate the norms of trust agreements between Identity Providers and Service Providers.

Implementation and service-call responsibilities

SourcedId requirements in calls to the Person Service

To comply with the norms just described a "SourcedId" that uniquely identifies a user's login via a particular Identity Provider, irrespective of the client application that manages the authentication, is constructed of two distinct elements:

1. **unique user identifier**: the unique user identifier supplied by an IdP is always encrypted by a trusted client, using a one-way hash (SHA-256) before it is transmitted to the Bamboo *Person Service* as input to a request for an existing or new Bamboo Person Identifier (BPId), thus circumventing the need to share an IdP-released attribute among independent applications/platforms; and,
2. **unique IdP identifier**: the IdP identifier is constructed according to a convention followed throughout the Bamboo Ecosystem, by all participants in the Bamboo Trust Federation (more on this at *Conventions for representing Identity Providers in the Bamboo Trust Federation*).

The SourcedId is only passed (as elements in an XML document or in a URL) for certain requests to the *Person Service*: those requests that associate, modify, or disassociate a Bamboo user identity with a login via one or multiple Identity Providers; and those that obtain for the client a *Bamboo Person Identifier* (BPId) which is used in subsequent requests to BSP-hosted services to identify the logged-in user. The BPId is transmitted in an HTTP request header, as described in the following paragraphs.

HTTP Header requirements in all calls to services deployed on (secured instances) of the BSP

The IAM solution implemented by Project Bamboo requires a client application to add a set of HTTP headers to requests sent to IAM and other centrally-hosted services – those that, in project parlance, are *hosted on the Bamboo Services Platform*. These HTTP request headers include the following:

- **X-Bamboo-AppID**: A UUID that identifies the client research environment, application, tool, or service; this UUID is issued as part of the process of *registering* a trusted client in the Bamboo Trust Federation as described above.
- **X-Bamboo-BPID**: A UUID that identifies the logged-in user on whose behalf the request is being sent; this is obtained via a service call that occurs in time between user login and any other service request. This is normally a Bamboo Person Identifier (BPId) as described above. [c]
- **X-Bamboo-Roles**: A pipe-delimited (|) set of scoped roles asserted by the trusted client to belong to the logged-in user, of the form role@domain, which are to be evaluated as factors in the determination of whether the request satisfies policies (access restrictions) that apply to the requested resource. If a user is *authenticated*, the client is expected to include the role *undefined@domain* where *domain* identifies the organization that authenticated the user (example: *undefined@google.com* is a client app's assertion that the user authenticated to Google). This header is otherwise optional (depending on policies governing the requested resource that may require one or more scoped roles for access to be permitted). Example of multiple roles asserted in this header: *roleA@domainOne.xxx|roleB@domainTwo* [d]

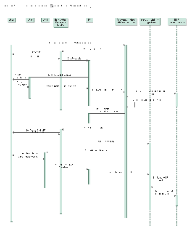
[c] The value of X-Bamboo-BPID is set to the identifier for the application itself (X-Bamboo-AppID) when a client application calls the Person Service to *crea te* a new Bamboo Person Identifier; or to *retrieve* the BPId for a user based on the identifier of the IdP with which she has logged in and an SHA-256 hash of that IdP's user identifier for the logged-in person.

[d] Client applications are responsible for extracting the conventionally-represented IdP identifier from attributes available to it following successful user authentication; the appropriate string is expected to be the top-level domain and its immediate subdomain in the Apache httpd environment variable *Shib-Identity-Provider* available to the client on successful user authentication (e.g. extract "wisc.edu" from the Shib-Identity-Provider value "https://logintest.wisc.edu/idp/shibboleth"). Policies and policy evaluation are described on the page *Authorization and Policy*. Also see *Conventions for representing Identity Providers in the Bamboo Trust Federation*.

Client Application Proof of Concept

A client application to manage Bamboo identities (create identities, associate them with logins via trusted identity providers, update profiles) was begun in Fall 2012, but halted in mid-stream when future funding for the Bamboo Technology Project was not forthcoming. The client app was a Drupal module, named the *Account Services Module*, and was implemented to a point sufficient to constitute a Proof of Concept for (a) integrating a trusted client in the Bamboo Trust Federation; (b) processing authentication via independently-hosted Identity Providers (e.g., Google, UC Berkeley, UW Madison); and (c) managing identities through trusted-client web service calls to the *Person*, *Contact*, and *Person Profile* services hosted on the Bamboo Services Platform, including management of account linking (recognizing a user who logs in from any of multiple IdPs as the same individual, i.e., the same "Bamboo Person" identified by a single BPId).

An overview of the module's design and coding approach can be found on the page *Account Services UI - Bamboo IAM Client - Drupal Module PoC*. The Project Bamboo wiki archives contain a complete set of requirements for this application, cf. *Account Services Drupal module workflow* and the linked page *Account Linking Service Draft Sequence Diagram* (a search for the page names on the wiki instance at Berkeley where the archive is housed should turn up the page if the link breaks...). The draft sequence diagram for the Account Linking use case is also included directly below this paragraph (click thumbnail for a larger view), to illustrate the types of component interactions the Account Services Module is designed to orchestrate:



Discussion of integration of a Drupal instance with the Bamboo Trust Federation can be found on the page *Client Environment-Tool-Service Integration with Bamboo IAM infrastructure*.

Authorization and Policies

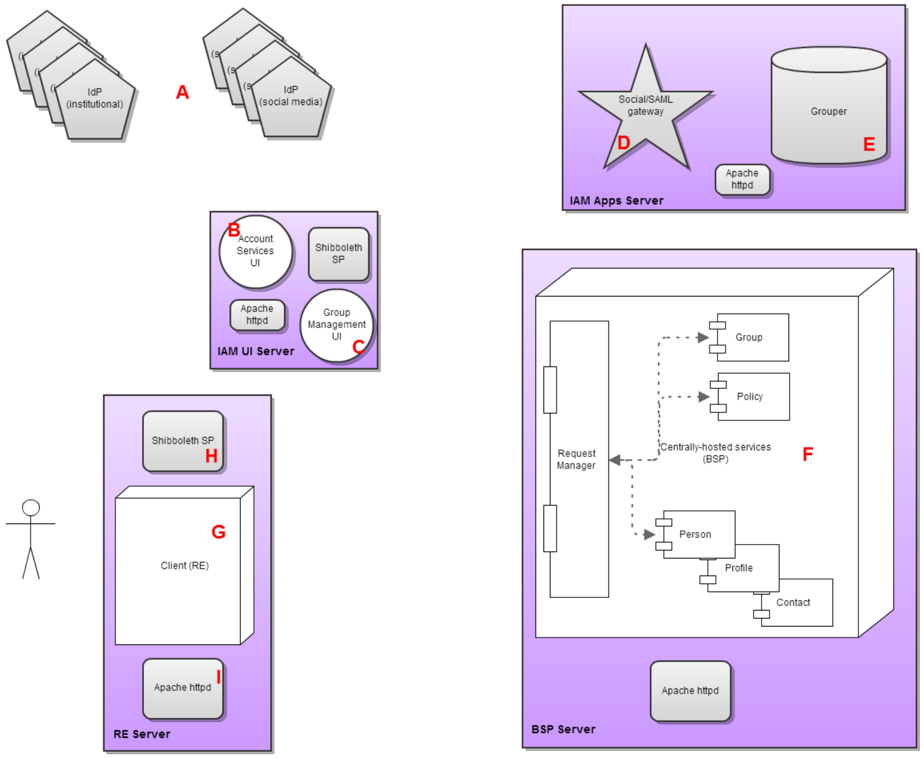
Authorization for resources served by the services running on the Bamboo Services Platform (BSP), or resources for which the BSP manages policy, is described in some detail on the page *Authorization and Policy*. That page describes:

- high-level XACML concepts around which Bamboo authorization is implemented;
- points in the processing of a web services request at which access policy may be enforced;
- factors used in deciding whether access to a service or resource will be granted;
- types of policy decisions supported;
- how user attributes are referenced in Bamboo policies; and,
- how Bamboo policies are put into effect.

Technology component overview

Bamboo's IAM infrastructure includes a number of component technologies in addition to the services developed to run on the Bamboo Services Platform (BSP), including the FUSE ESB that provides the deployment environment for the BSP. The diagram below, its labels, and the list beneath the diagram briefly describes these technologies and their function in the infrastructure.

> ⚠ The deployment topology implied by the purple boxes -- pairing the Social/SAML Gateway with Grouper, and the Account Services and Group Management UI – is *not* required; it reflects only the implemented or planned deployment topology used for development and testing during the course of the Bamboo Technology Project.



IAM Components

A - **Identity Providers** (IdPs): These exist independent of Project Bamboo, but are participants in the *Bamboo Trust Federation* described above. Institutional IdPs are generally SAML implementations, usually Shibboleth, operated by institutions of higher education; social media IdPs are generally commercial entities, such as Google or Facebook, widely trusted to provide identities associated with social media services.

B - **Account Services User Interface**: This interface was partially built as a Drupal module, and has been used to demonstrate client integration, cf. *Account Services UI - Bamboo IAM Client - Drupal Module PoC*. The purpose of this interface is to present to end-users an easily navigated, browser-based means of self-registration, profile maintenance, and 'account linking' (the latter refers to the ability to link a single Bamboo Person identity to logins via any number of IdPs).

C - **Group Management User Interface**: This interface was *not* built in the completed phase of work. Its purpose, when built or adapted, will be to present an easily-navigated, browser-based means to create and organize, populate, and manage groups that are maintained centrally. Memberships in centrally-maintained groups are intended to be used as factors in determining permissions (access to services and resources) across multiple applications and platforms. Cf. *Group Service Contract Description - v0.9-alpha*.

D - **Social/SAML Gateway**: This gateway transforms the user-attributes supplied by social media IdPs (e.g., Google, Facebook) into SAML assertions. SAML is the authentication *lingua franca* for the Bamboo IAM infrastructure.

E - **Grouper**: An Internet2 open-source application, Grouper is used to store data about known and trusted applications in the Bamboo ecosystem, and to store data representing groups created and managed by users. *Users are expected to access these groups via the (to-be-implemented) Group Management User Interface (C, above); the Group Management User Interface and other client applications are intended to direct requests to the Group Service centrally-hosted on the Bamboo Services Platform. Grouper is an implementation choice used by the Bamboo Technology Project team to persist group-related data* behind *the Group Service. It is therefore, like the database used by services on the BSP, or the file system used for storage of some data, a technology located 'deep' in the background of the Bamboo ecosystem. Grouper is included on the diagram and listed here (while neither a database nor the BSP host's file system are referenced) because it is a less-common choice of persistence technology than a relational database or a server file system.*

F - **Bamboo Services Platform (centrally-hosted services)**: The *BSP* is an instance of Fuse ESB (based on Apache ServiceMix), used as a container (deployment host) for services intended to be run centrally as the 'hub' of Bamboo's IAM infrastructure, collection interoperability services (see *Interoperability of Digital Collections*), and proxied services that enable access to remotely-hosted tools for humanities scholarship (see *Proxied Access to Remotely Hosted Tools for Scholarship*).

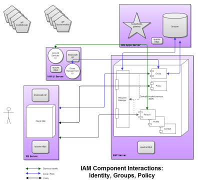
G - **Client application/tool/service**: This placeholder in the diagram may be any application, research environment, tool, or service participating in the Bamboo Trust Federation, described above. The Account Services User Interface (B in the diagram) was partially-implemented during the Bamboo Technology Project (October 2010 to September 2012) as an example of how such a client can be integrated into the Bamboo ecosystem.

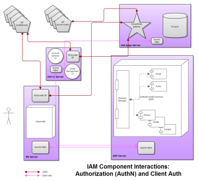
H - **Shibboleth SP**: This is a well-known and widely-deployed technology provided by the open-source Shibboleth project. As described on the project's web site, *The Shibboleth Service Provider SSO-enables and Federation-enables web applications written with any programming language or framework. It integrates natively with popular web servers like Apache and IIS. A loosely coupled integration strategy allows you to support SAML, rich attribute-exchange, and many value-added features, often without significantly changing your application code or using proprietary interfaces.*

I - **Apache Web Server (httpd)**: This is a well-known and ubiquitously-deployed open-source web server technology. In the Bamboo IAM infrastructure, it is used to establish trust, via exchange of X.509 certificates (which may be self-signed) between applications/tools/services/platforms participating in the Bamboo Trust Federation, described above.

Additional diagrams based on the one shown above sketch component interactions:

- Components that interact to enable identity, group, and policy are sketched in the diagram whose thumbnail is on the left
- Components that interact to enable Client Auth (authentication of applications and platforms in the Bamboo ecosystem) and user authentication (AuthN) are sketched in the diagram whose thumbnail is on the right.




IAM Component Interactions:
Authorization (AuthN) and Client Auth

Installing and configuring technology components to support IAM

Documentation about installation and configuration are principally located under the wiki page *Install and Configure Technology Components*.

Technology Component (s)	Documentation Link (s)	Notes
Developer toolkit	Developer Workbench Environment for BSP Service Developers	Java, Maven, Eclipse and IDE plugins, required filesystem directories, required environment variables
FUSE ESB	Developer Workbench Environment for BSP Service Developers	Core element of Bamboo Services Platform (BSP, the deployment container for centrally-hosted services whose APIs are linked from *Service APIs - Centrally-Hosted Bamboo Services*)
PostgreSQL database	Developer Workbench Environment for BSP Service Developers	Relational database providing persistence for BSP-deployed services
Core BSP-deployed services	Developer Workbench Environment for BSP Service Developers	Core services, including those that support IAM
Apache Web Server (httpd)	Configure Apache Web Server for Client Auth	httpd supports client auth (authenticating trusted client applications), as well as proxy-forwarding over AJP of BSP services
Grouper	Grouper Install - Configure - Populate	Grouper provides persistence for the "Application Catalog" (known/trusted client applications in the Bamboo Trust Federation); as well as for user-created and -managed groups
Application Catalog data	Maintaining Application Catalog Data for Trusted Clients	For an application to be trusted (a key element of gaining permission to invoke services protected by policies that restrict access), Application Catalog data must be maintained.
Trust Federation metadata	Maintaining SAML Metadata that establishes a Trust Federation	Identity providers and service providers trusted within the Bamboo Trust Federation must be identified with SAML metadata.
Social/SAML Gateway	Social/SAML Gateway to enable social media identity provision	A Social/SAML gateway must be a part of the Authentication 'machinery' if social media Identity Providers (e.g., Google) are to be used for user logins.

Clients	• Shibboleth SP Installation and Configuration for Bamboo Trust Federation Clients • Configure Apache Web Server for Client Auth (*Certificate Exchange* section)	When policies in effect restrict access to anonymous users or anonymous applications, only "Trusted Applications" can succeed in invoking the affected services. Only "Trusted Applications" can assert the identity of a user to BSP-deployed services (anonymous client apps imply anonymous users). • Clients acting as a "Trusted Application" generally do so by functioning with an authentication context backed by Shibboleth SP (Shibboleth Service Provider). Note install and configure document linked from column at left. • In addition, exchange of certificates and metadata are necessary for a client application to act as a "Trusted Application" in the Bamboo Trust Federation (this includes simple web service clients such as Firefox Poster or curl). Note Client Auth document linked from column at left, particularly the *Certificate Exchange* section of that page.. • In addition to the installation and configuration documentation linked from the column at left, see the *Client responsibilities for implementing the conventions* section of the page *Conventions for representing Identity Providers in the Bamboo Trust Federation*.

Links to architectural overview and service APIs

For an architectural overview of services on the Bamboo Services Platform, including IAM services, see *Architectural Overview of the Bamboo Services Platform*.

Service APIs for Person, Person Profile, Contact, Group, Request Manager, Protected Resource, and other centrally-hosted services, see *Service APIs - Centrally-Hosted Bamboo Services*.