Liberty Identity Web Services Framework Primer

Draft Version 1.0-04

12 April 2003

Editor:

Jonathan Tourzan, Sony

Contributors:

John Beatty, Sun
Jeff Hodges, Sun
Gary Ellison, Sun
John Kemp, IEEE-ISTO
Jason Rouault, HP
Robert Aarts, Nokia
Jukka Kainulainen, Nokia

Abstract:

This primer is a *non-normative* document intended to provide an overview of the relevant features of the Liberty ID-WSF Version 1.0 Specifications. It provides a general introduction to the Liberty ID-WSF framework, and to how it fits in with the other layers of the Liberty architecture. The reader is assumed to have some familiarity with SOAP 1.1, WS-Security, SAML and basic concepts such as namespaces and URIs.

Notice

Liberty Alliance Project
Licensing Administrator
c/o IEEE-ISTO
445 Hoes Lane
Piscataway, NJ 08855-1331, USA
info@projectliberty.org

Contents

1. Introduction

1.1 About this document

This primer is a *non-normative* document intended to provide an overview of the relevant features of the Liberty ID-WSF Version 1.0 Specifications. It should provide a general introduction to the Liberty ID-WSF framework, and to how it fits in with the other layers of the Liberty architecture.

Further details of the Liberty ID-WSF may be found in the following normative technical specification documents [ID_WSF Discovery Service, ID-WSF SOAP Binding, ID-WSF Security Profiles, ID-WSF Intereaction Service, ID-WSF Client Profiles, ID-WSD Static Conformance Requirements, and ID-WSF Data Services Template]. Definitions for abbreviations and acronyms not immediately defined in this document may be found in the Liberty Technical Glossary documents for Liberty ID-FF and Liberty ID-WSF [REF TECH GLOSSARY]. As this document is non-normative it does not use terminology “MUST”, “MAY”, “SHOULD” in a manner consistent with RFC-2119.

The goal of this paper is to provide sufficient information such that a technically competent individual may understand the Architecture defined by the ID-WSF framework and the basic usage scenarios defined for use within the framework. The paper also highlights how the ID-WSF interacts with an identity management framework (such as Liberty ID-FF).

The reader is assumed to have some familiarity with SOAP 1.1, WS-Security, SAML and basic concepts such as namespaces and URIs. The ID-WSF specifications draw upon work conducted in Oasis, W3C and IETF. Standards referenced in a normative manner include SAML, WS-Security, HTTP, WSDL 1.1, XML, SOAP 1.1, XML-ENC, XML-SIG, SSL/TLS, and WAP.

1.2 Liberty Modules

The Liberty architecture consists of a multi-level layered specification set, based on open standards including SAML and SOAP. There are three major components of the Liberty architecture:

1. The Liberty Identity Federation Framework (ID-FF) specifies core protocols, schemata and concrete profiles that allow implementers to create a standardized, multi-vendor, identity federation network.
2. The Liberty Identity Web Services Framework (ID-WSF) consists of a set of schemata, protocols and profiles for providing a basic framework of identity services, such as identity service discovery and invocation.
3. Liberty Identity Service Instance Specifications (ID-SIS) utilize the ID-WSF and ID-FF to provide networked identity services, such as contacts, presence detection or wallet services that depend on networked identity.

Figure 1 below illustrates the Liberty Modules and their corresponding functional areas.

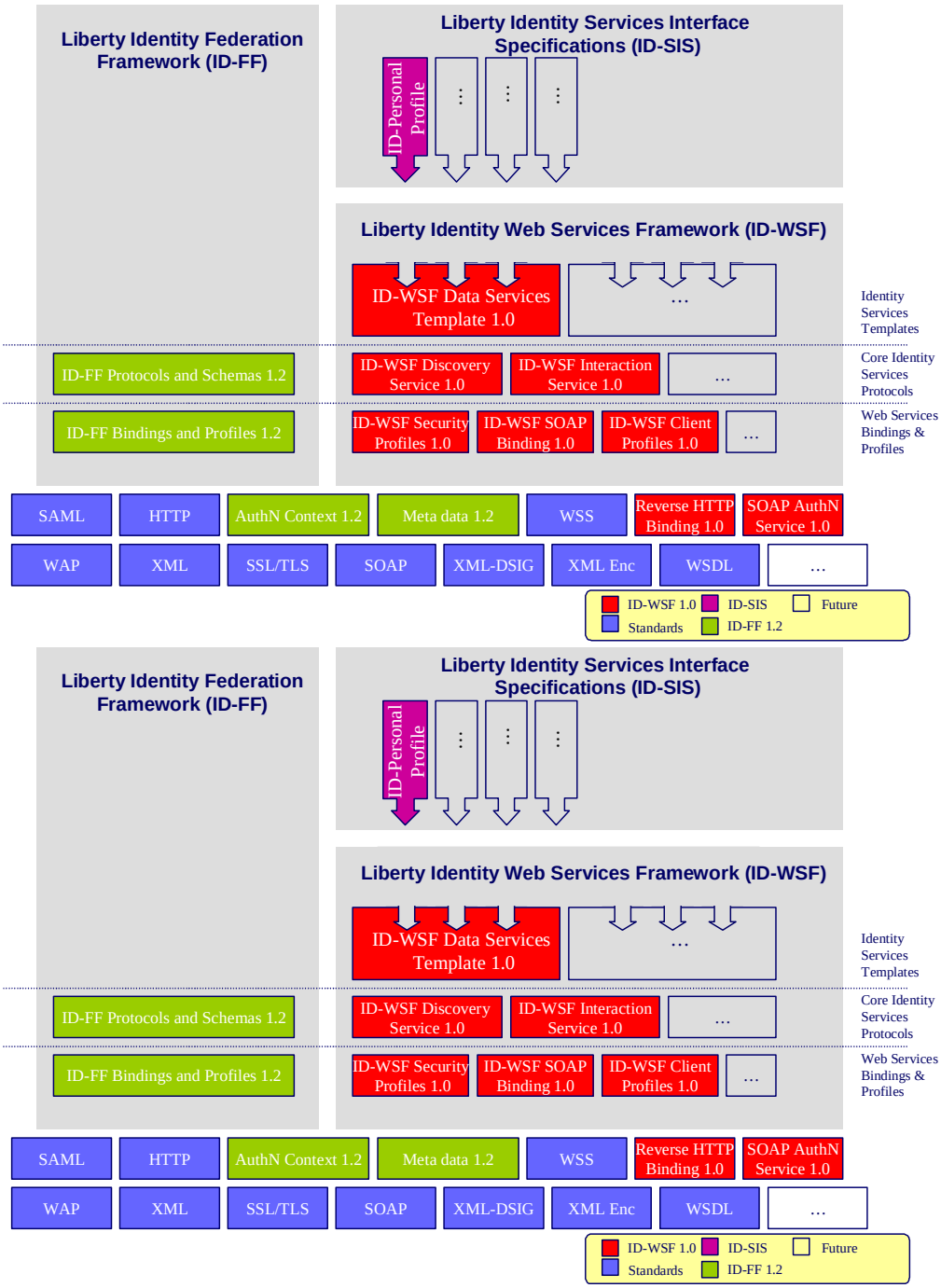


Figure 1: Liberty Modules

1.3 What is the Identity Services Framework?

The Liberty Identity Services Framework defines a SOAP based invocation framework with a layered architecture. The framework does not specify any contents for the SOAP body, allowing the development of identity services within the context of the Liberty Identity Services Framework. The layering is schematically depicted below.

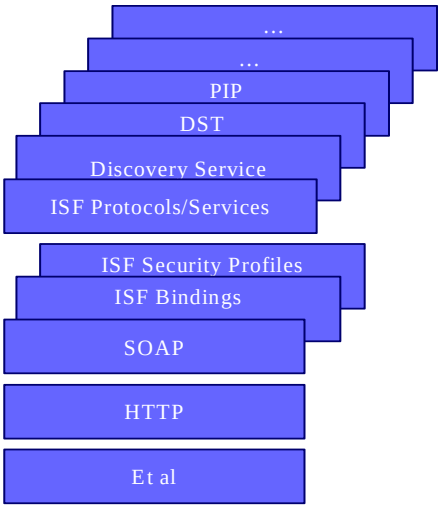


Figure 2: Liberty ID-WSF Protocol Architecture

1.4 Concepts and Architecture

The Liberty ID-WSF defines a framework for creating, discovering, and consuming identity services. The Liberty ID-WSF also defines a conceptual model that provides relevant terminology for these *identity services.* Some basic identity services, such as the discovery service, are defined in a normative manner as part of the ID-WSF Specifications. The following UML model describes the conceptual model presented in the Liberty Specifications:

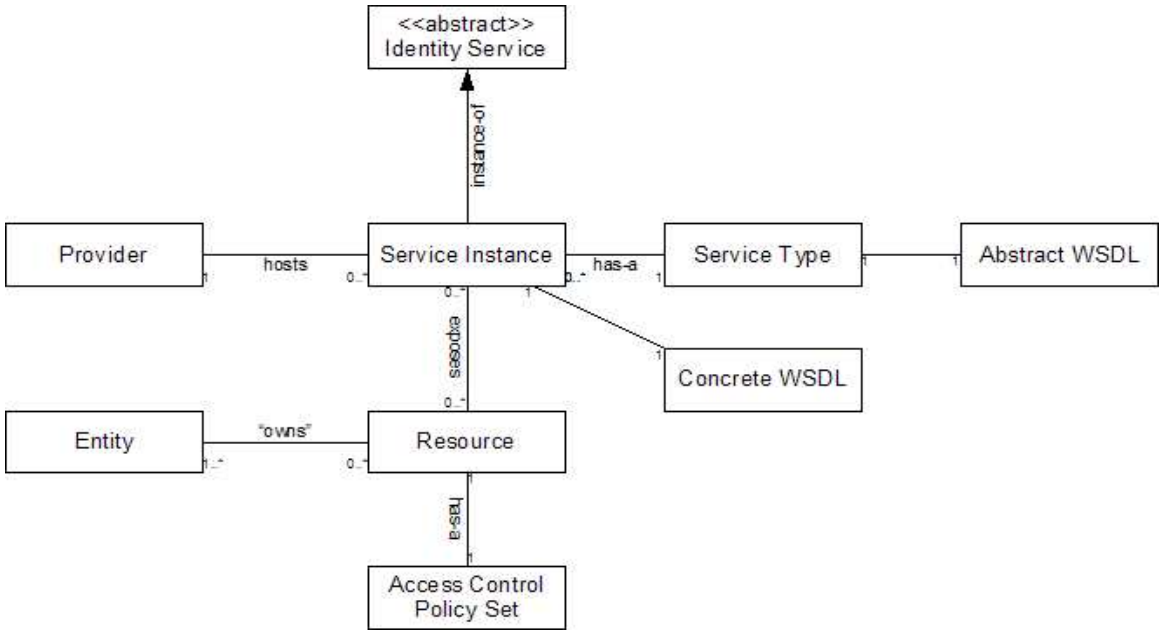


Figure 3: UML Representation of Liberty Conceptual Model

An *identity service* is an abstract notion of a web service that acts upon some resource to either retrieve information about an identity or identities, update information about an identity or identities, or perform some action for the benefit of some identity or identities.

There are different types of identity services, each of which is identified by a *service type identifier*. This service type identifier maps to exactly one *abstract WSDL* definition of a service, which contains only the type, message, and portType elements of a WSDL 1.1 description. An example of a service type is a "calendar service," which could be identified by a URI such as "urn:example:services:calendar".

A *service instance* is the instantiation of a particular type of identity service. A service instance maps to a *concrete WSDL* document (which includes the binding and service WSDL elements) that contains the *protocol endpoint* and additional information necessary for a client to communicate with the particular service instance (e.g., security policy information).

Each service instance is hosted by some *provider*, which is identified by a *provider identifier*. An example of a service instance is a SOAP endpoint offering a calendar service.

A service instance exposes a protocol interface to a set of resources. A *resource* in this specification is either data related to some identity or identities, or a service acting on behalf of some identity or group of identities. An example of a resource is a calendar containing appointments for a particular identity.

A resource commonly has *access control policies* associated with it. These access control policies are typically under the purview of the entity or entities associated with the resource (in common language, the entity or entities could be said to "own" the resource). The access control policies on a resource must be enforced by the service instance.

2 Synopsis of Specifications

ID-WSF SOAP Binding (ID-WSF/Normative)

The ID-WSF SOAP Binding provides a SOAP-based invocation framework for identity services. It defines SOAP Header blocks and processing rules enabling the invocation of identity services via SOAP requests and responses. Additionally, a usage directive container is defined for those implementations that wish to use an existing rights expression language to specify the required service and data usage policies.

ID-WSF Security Profiles (ID-WSF/Normative)

This specification describes profiles and requirements for securing the discovery and use of identity services. It includes security requirements to both protect privacy, and to ensure integrity and confidentiality of messages between service providers.

ID-WSF Discovery Service (ID-WSF/Normative)

Defines a core identity service that enables various entities (e. g., Service Providers) to dynamically discover a user's registered identity services. Given the type of service desired (e. g., Person Profile Service), the Discovery Service responds, on a permission- basis, with a service description containing WSDL for the desired identity service. The Discovery Service can also function as a security token service, issuing security tokens to the requester that the requester will use in the request to the discovered identity service.

ID-WSF Data Services Template (ID-WSF/Normative)

Provides the building blocks when implementing a data service (e. g. Person Identity Profile service) on top of the Identity Services Framework. The specification defines how to query and modify data stored in a data service and provides some common attributes for data services.

ID-WSF Resource Owner Interaction (ID-WSF/Normative)

An identity service may need to obtain permission from a user (or someone who owns a resource on behalf of that user) to allow them to share data with requesting services. The resource owner interaction specification details protocols and profiles for interactions that allow services to carry out such actions.

ID-WSF Client Profiles (ID-WSF/Normative)

Describes the profiles and requirements for Liberty-enabled clients interacting with the SOAP based authentication service.

Metadata (ID-FF/ID-WSF Independent)

With this release, schema and protocols are introduced to facilitate real-time requests for metadata (assumed out of band transfer previously). This will allow for more spontaneous conversations between Liberty-compliant entities. A mechanism is defined for publishing the metadata, and several mechanisms for retrieving the metadata are defined (DNS, well known location). The metadata architecture is designed to be flexible going forward.

Functionally, there are two primary classes of metadata:

entity core metadata, which covers the metadata elements introduced in release 1 of the protocol with additional elements introduced in this release. Core metadata includes information about cryptographic keys used by entities, SOAP related information for service endpoints, as well as IdP/SP specific information and other service related information.

entity trust metadata, which enables entities to cast business decisions based on the characteristic trust information provided in this class. This is not defined within the Alliance, but the metadata architecture could be used to publish or retrieve this data.

Reverse HTTP Binding (ID-FF/ID-WSF Independent)

Enables a normal HTTP-based user-agent to receive SOAP requests inside an HTTP *response*. This allows end users to host identity services on their devices without running an HTTP server or being IP addressable from the Internet.

SOAP Authentication Service (ID-FF/ID-WSF Independent)

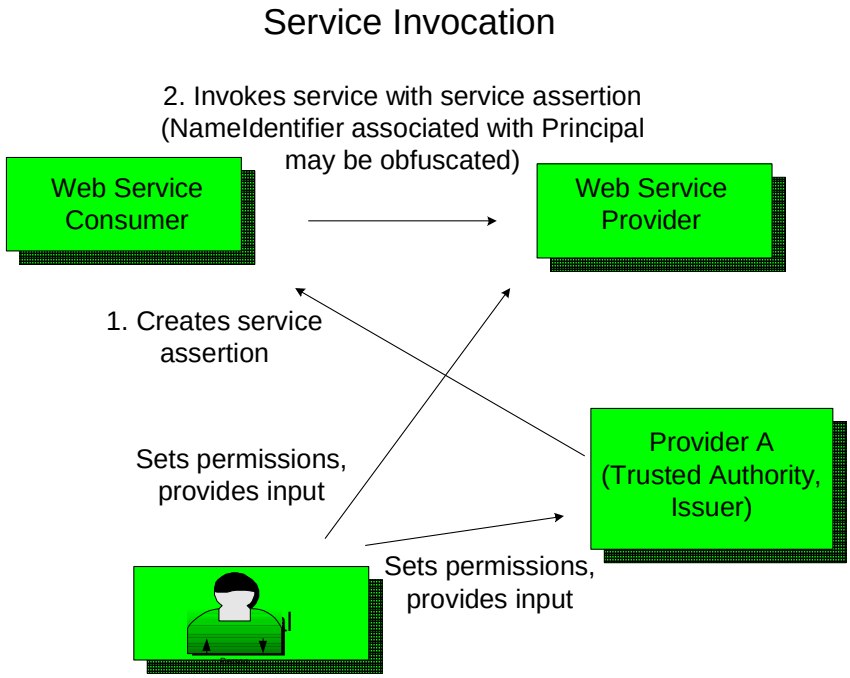
Defines how to authenticate parties who are communicating via SOAP-based messages. It leverages widely used authentication services and mechanisms, and facilitates selection of these services and mechanisms at deployment time. This specification also defines an identity-based authentication security token service, complementing the more general security token service defined by the ID-WSF Discovery Service.

3 Identity Services Framework

3.1 Service Invocation

The Liberty Identity Services Framework defines a SOAP based invocation framework that allows identity services to be discovered and invoked. Once a service has been discovered and sufficient authorization data has been received from a trusted authority, the invoking entity (Web Services Consumer) may invoke service at the hosting/relying entity (Web Services Provider). In order to convey the privilege of a system entity to access a resource, the framework defines extensions such that service invocation authorization data may be generated by a trusted authority and issued to the invoking system entity. The relying party or WSP can make access control decisions based upon this authorization data based upon its business practices and the preferences of the Resource Owner. In most cases this trusted authority is assumed to be some Identity Provider/Discovery Service.

The following diagram illustrates the entities involved in possible service invocation use cases.



198

199 **Figure 4: Service Invocation Context**

200 ### 3.1.1 Security Profiles

201 As in other web services contexts, access control policies must be enforced in an identity services context. The
202 authorization decision to invoke an identity service instance offering a specific resource may be made locally (that is at
203 the entity hosting the resource) or remotely. Regardless of whether the policy decision is distributed or not, in a
204 permissions based context or any context with security considerations, a policy enforcement must always be
205 implemented by the entity hosting the resource.

206 Identity services may rely upon a trusted third party (TTP) to make policy decisions on their behalf. In such cases, the
207 Trusted Third Party issues targeted SAML assertion to those entities. This assertion has associated conditions, such as
208 an issue instant, validity periods for the assertion. The SAML assertion also has as audience restriction(s) that provide
209 information about for whom the policy decision is intended, the relying party (a Web Service Provider) for the
210 particular assertion. The SAML assertion also contains an Authorization Decision Statement which conveys the
211 decision, information about what rights have been granted to the resource, as well as information about the Subject and
212 the Subject Confirmation Method by which the requesting entity will authenticate itself to the relying party.

213 **Please see the section on delegation for additional information. The trusted authority**
214 **digitally signs the entire assertion in order to protect the assertion from**
215 **modification**.

216 ### 3.1.2 Usage Directives

217 The Liberty ID-WSF defines extensions that allow both the invoking entity and the consuming entity to add one or
218 more Usage Directive SOAP headers to a message. A Usage Directive header in a request from the invoking entity can
219 be understood as "intended usage". It should be noted that should permissions be such that a Usage Directives level
220 in the request cannot be met, the hosting entity must either redirect the invoking entity to the user to query for
221 permission, or simply deny the service.

3.1.3 ROI Service

The Liberty ID-WSF defines a Resource Owner Interaction (ROI) protocol. This protocol provides schemas and profiles to enable an entity to interact with the owner of a resource that is exposed by that WSP. The ID-WSF defines three ways for a WSP to interact with a user:

1. The WSP may send a SOAP response with a RedirectRequest that instructs the WSC to direct the user-agent to contact the WSP at a given URL.
2. The WSP may send a UserInteractionRequest to the endpoint defined in the ROIService element.
3. The WSP may try to discover the ROI service of the resource owner, so the WSP may send a userInteractionRequest to that service.

This interaction may be for purposes of obtaining consent for a particular resource exposure (such as granting access to ID-PP), obtaining data from the user-agent, or some other purpose. The ROI protocol is an optional part of the Liberty ID-WSF. An example of how the ROI could be used would be to query the user for permissions in a web services context.

3.1.4 Delegation

The Liberty ID-WSF supports a restricted form of delegation whereby a system entity can act on behalf of the Principal to access an identity service. In order to achieve this, Liberty defines a new Subject Confirmation Method, Delegated Holder of Key, which allows delegated access to resources. The delegation functionality can be used in offline scenarios when the user is present. An example of this might be an Authorization Decision Statement allowing a delegated entity to update a calendar resource for a particular identity after a flight booking has occurred.

3.1.5 Affiliations

An affiliation allows a group of SPs organized to act as a single entity from the point of view of the customer (usually due to the group acting as a portal or acting as a single company such as TimeWarner and its affiliates). The ID-WSF Authorization Decision Statement defined in ID-WSF allows the use of Affiliation ID when a trusted authority is granting rights to a member of an affiliation group. An example of a use of affiliations in an application context might be an Authorization Decision Statement allowing Travel Affiliation X to update a calendar after a flight booking has occurred.

3.1.6 Chaining of Services/Broker

The ID-WSF architecture a broker type functionality, whereby, a WSC may make a request to a WSP who acts as a broker and makes subsequent requests (as a WSC) to other WSP(s) who have the required information. The Broker subsequently aggregates the data and responds to the originating WSC in the chain. A simple example might be that profile data is stored in various places and the broker needs to query the relevant parties for data prior to responding to a ID-PP request.

3.1.7 Anonymous Service Requests

The Trusted Third Party may obscure the subject's name identifier for purposes of confidentiality at the Web Service Consumer and any subsequent intermediaries. For this purpose, the ID-WSF specifies a mechanism for creating (at Issuer) and consuming (at relying party) encrypted name identifiers. [Notes: still some details to be resolved]

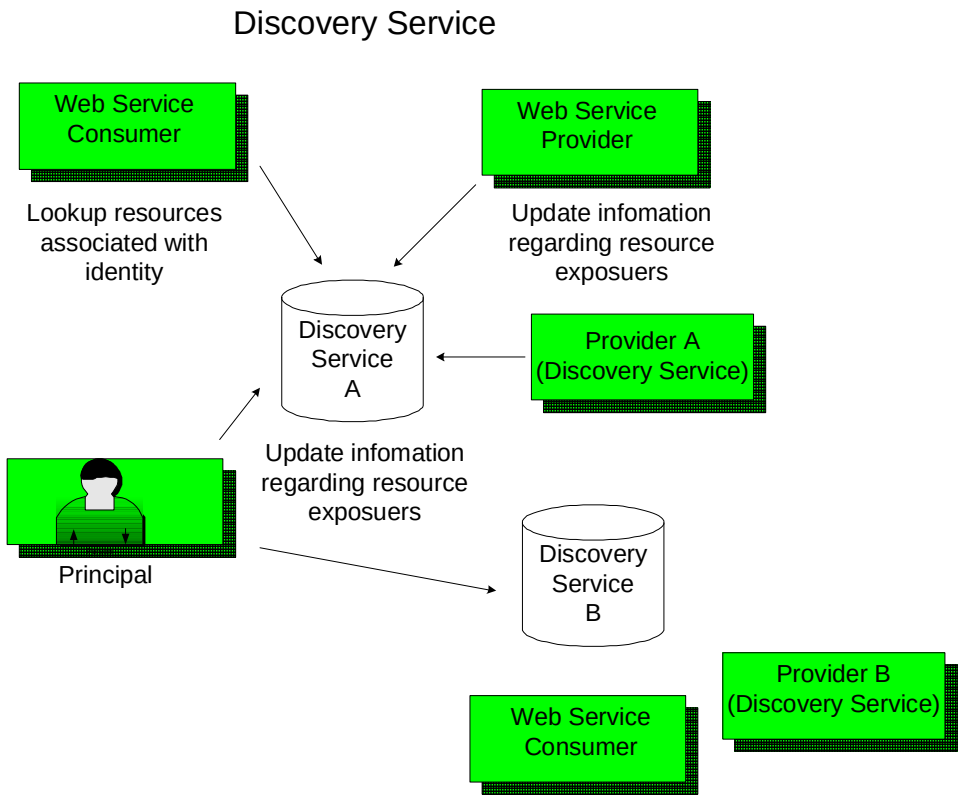
3.2 Discovery Service

The Discovery Service is a type of identity service that provides for the discovery of resource exposures associated with a given identity. An identity will typically have one or more discovery resources on the network that allow other entities to discover its identity services.

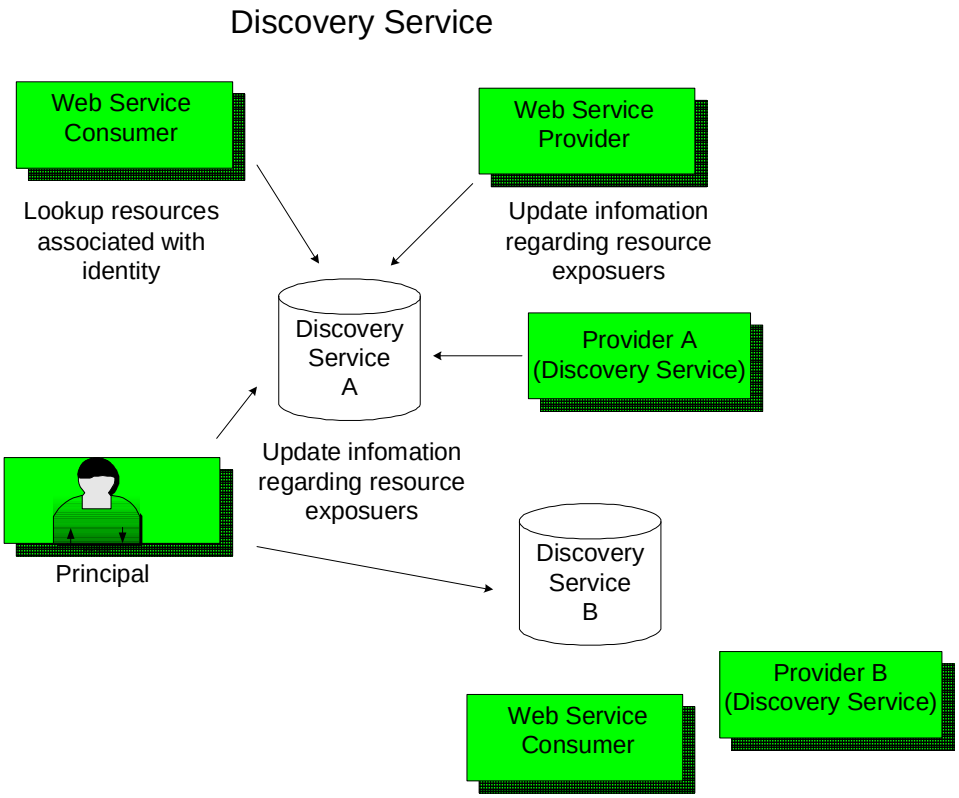
The Discovery Service offers two operations, Lookup and Update. In a web services context (browsing, etc.), a Web Services Consumer may need access to a resource exposure associated with an identity (a profile or location service). The Web Service Consumer may lookup a service instance with a Request that may include a service type element and extensible processing directives. The response message contains the relevant resources associated with the query,

266 according to the access policies set by the principal/provider, and the response may include tokens for service
267 invocation.

268 The Update Operation allows for a requester to enter and remove service instances. The Request allows the provider
269 to input information about a resource exposure, and the corresponding Response provides the status of the request. A
270 Web Service Provider that hosts the resource, the host of the Directory Service, or the Principal/Resource Owner could
271 update the resource exposure. The service registry defined by the Liberty ID-WSF is flat, so complex queries are not
272 possible (one service entry for each service type). This does not preclude having some ability to change the Lookup
273 results based upon the Access Control Policies of the host, and/or Preferences/Permissions of the resource owner. The
274 following diagram illustrates the entities involved in possible discovery service use cases.



275



276 **Figure 5: Liberty Discovery Service**

277 4 Use Cases in scope for ID-WSF

278 The Liberty Alliance defines a ID-PP (Identity - Personal Profile) service for use with the Liberty ID-WSF. The ID-
279 PP service is designed to facilitate account creation in a web services context. The ID-PP service allows a Web
280 Service Consumer to gather the information necessary to create an account or provide personalized services. The ID-
281 PP Specification provides a schema and API for queries of personal information. The ID-WSF provides ID-PP
282 deployments and other ID-SIS deployments the ability to specify and negotiate usage directives for attribute sharing,
283 to query users for permissions using the ROI, as well as the ability to provide anonymous attribute requests for non-
284 identifying ID-PP attributes (such as zip code).

285 5 Use Cases out of scope for ID-WSF, but relevant to later work

286 The Liberty Alliance anticipates that other services will be built on top of the Liberty ID-WSF. Some of these services
287 will be specified within the Alliance context, other services will be proprietary applications built on top of the Liberty
288 ID-WSF Architecture. It is anticipated that services such as wallet, calendar, messaging, presence, geo-location and
289 user groups will be useful in conjunction with the Liberty ID-WSF, and these services may be formally specified by
290 the Alliance.