Liberty Technical Glossary

Version: v1.4

Editors:
Jeff Hodges, Sun Microsystems, Inc

Contributors:
Robert Aarts, Nokia Corporation
Carolina Canales-Valenzuela, Ericsson
Peter Davis, NeuStar, Inc.
John Kemp, IEEE-ISTO
Elisa Korentayer, IEEE-ISTO
John Linn, RSA Security, Inc.
Thomas Wason, IEEE-ISTO

Abstract:

This glossary defines many of the technical terms and phrases used in the Liberty Alliance"s various specifications and documents.

Filename: liberty-glossary-v1.4.pdf

Notice

Implementation of certain elements of this document may require licenses under third party intellectual property rights, including without limitation, patent rights. The Sponsors of and any other contributors to the Specification are not, and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights. **This Specification is provided "AS IS", and no participant in the Liberty Alliance makes any warranty of any kind, express or implied, including any implied warranties of merchantability, non-infringement of third party intellectual property rights, and fitness for a particular purpose.** Implementors of this Specification are advised to review the Liberty Alliance Project's website (http://www.projectliberty.org/) for information concerning any Necessary Claims Disclosure Notices that have been received by the Liberty Alliance Management Board.

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

Revision History

Contents

1. Introduction

This document is Liberty Alliance Project glossary of normative technical terms.

This document is not an exhaustive compendium of all Liberty technical terminology because the Liberty terminology is built upon existing terminology. Thus many terms that are commonly used within this context are not listed. They may be found in the glossaries/documents/specifications referenced in the bibliography. Terms defined here that are not attributed to other glossaries/documents/specifications are being defined here.

This glossary is expected to evolve along with the Liberty Alliance Project specifications.

2. Definitions

Terms

AC

See *authentication context*.

account

A formal business agreement providing for regular dealings and services between a Principal and a service provider [Merriam-Webster].

account linkage

See *identity federation*.

AD

See *Authentication Domain*.

affiliation

An affiliation is a set of one or more entities, described by providerID's, who may perform Liberty interactions as a member of the set. An affiliation is referenced by exactly one affiliationID, and is administered by exactly one entity identified by their providerID. Members of an affiliation may invoke services either as a member of the affiliation (using affiliationID), or individually (using their providerID). *Affiliation* and *affiliation group* are equivalent terms.

affiliation group

See *affiliation*.

Affiliation ID

An *Affiliation ID* identifies an *affiliation*. It is schematically represented by the `affiliationID` attribute of the `<AffiliationDescriptor>` metadata element [LibertyMetadata].

AP

See *Attribute Provider*.

AS Consumer

See *Authentication Service Consumer*.

AS Provider

See *Authentication Service Provider*.

assertion, SAML assertion

An XML-based data structure defined by SAML. Assertions are collections of one or more statements, made by a SAML authority (also known as an *issuer*), such as an authentication statement or attribute statement. As used in Liberty, assertions typically concern things such as: an act of authentication performed by a Principal, attribute information about a Principal, or authorization permissions applying to a Principal with respect to a specified resource.

attribute

A distinct, named, characteristic of a *Principal* or other *system entity*.

attribute class

A predefined set of attributes, such as the constituents of a Principal's name (prefix, first name, middle name, last name, and suffix).

attribute container

a module comprised of a collection of attributes grouped together according to expected use patterns.

Attribute Provider (AP)

An *attribute provider* (AP) provides Identity Personal Profile (ID-PP) information. Sometimes referred to as an ID-PP provider.

authenticated identity

An *identity*, representing a *system entity*, which often is a *Principal*, that is asserted to have been the subject of a successful *authentication*.

authenticated Principal

A Principal who has had his *identity* authenticated by an *Identity Provider*.

authenticating authority

Synonymous with *authenticating identity provider* or *authenticating IdP*. An *identity provider* that authenticated a *Principal* (see also *authentication*). In [LibertyAuthnContext], the authenticating authority is identified by the first occurring `<AuthenticatingAuthority>` element instance.

authenticating entity

A *system entity* that engages in the process of authenticating itself to another *system entity*, the latter typically being an *Identity Provider* (see also *authentication*). More formally, an *authenticating system entity*.

authentication (authn)

Authentication is the process of confirming a *system entity*'s asserted *identity* with a specified, or understood, level of confidence [TrustInCyberspace].

authentication assertion

A SAML-based *assertion* that, in the Liberty specification suite, contains a `<lib:AuthenticationStatement>`. Note that the foregoing element is defined in a Liberty namespace. Also known as *Liberty authentication assertion* and *ID-FF authentication assertion*.

Liberty authentication assertions are formal XML extensions of SAML assertions [SAMLCore11].

Semantically, an assertion issuer is stating that the subject of the assertion authenticated with it (the issuer) at some point in time. Assertions are typically time-limited.

authentication authority

A *system entity* that produces authentication assertions [SAMLGloss]. In the Liberty architecture, it is typically an *Identity Provider*.

authentication context (AC)

Authentication Context is an extensible XML-based "schematic" description of authentication event characteristics [LibertyAuthnContext].

Authentication Domain (AD)

An Authentication Domain (AD) is a formal community of Liberty-enabled entities that interact using a set of well-known common rules.

authentication exchange

See *authentication protocol exchange.*

authentication mechanism

An *authentication mechanism* is a particular, identifiable, process or technique that results in a confirmation of a *system entity*'s asserted *identity* with a specified, or understood, level of confidence. See also *SASL mechanism*. An authentication mechanism may be employed in the process of generating *security tokens* attesting to the *authenticated identity* of an *authenticating entity*. The ID-WSF Authentication Protocol specifies such a process [LibertyAuthn].

authentication protocol exchange

Authentication protocol exchange is the term used in [RFC2222] to refer to the sequence of messages exchanged between the *client* and *server* as specified and governed by the particular *SASL mechanism* being employed to effect an act of *authentication*.

authentication quality

The level of assurance that a *service provider* can place in an *authentication assertion* it receives from an *identity provider*.

authentication server

The precise, specific *role* played by a *server* in the protocol message exchanges defined in the ID-WSF Authentication Protocol.

Authentication Service Consumer (AS Consumer)

A *Web Service Consumer* (WSC) implementing the *client*-side of the ID-WSF Authentication Service [LibertyAuthn].

Authentication Service Provider (AS Provider)

A *Web Service Provider* (WSP) implementing the *server*-side of the ID-WSF Authentication Service [LibertyAuthn].

authentication session

The period of time starting after A has authenticated B and until A stops trusting B's identity assertion and requires reauthentication. Also known simply as a *session*, it is the state between a successful login and a successful logout by a Principal.

authorization (authz)

The process of determining, by evaluating applicable access control information, whether a subject is allowed to have the specified types of access to a particular resource. Usually, authorization is in the context of authentication. Once a subject is authenticated, it may be authorized to perform different types of access [SAMLGloss].

bearer token

A *bearer token* is a form of *security token* having the property of connoting some attribute(s) to its holder, or *bearer*. In [LibertySecMech], bearer tokens connote *identity* and they consist essentially of *credentials* of some form, e.g. *SAML assertions* [wss-saml].

Circle of Trust (CoT)

A federation of service providers and identity providers that have business relationships based on Liberty architecture and operational agreements and with whom users can transact business in a secure and apparently seamless environment. Also known as a *Trust Circle*.

client

A *role* assumed by a *system entity* who makes a request of another system entity, often termed a *server* [RFC2828]. A client is at varying times a *sender* or a *receiver*.

CoT

See *circle of trust*.

credentials

Data that is transferred or presented to establish either a claimed *identity* or the authorizations of a *system entity*.

defederate, defederate identity

To eliminate linkage between a Principal's accounts at an identity provider and a service provider.

delegation

Enabling a system entity to operate on behalf of a principal to access an identity service.

discoverable

A *discoverable* "in principle" service is one having an *service type URI* assigned (this is typically in done in the specification defining the service). A discoverable "in practice" service is one that is registered in some discovery service instance.

ID-WSF *services* are by definition discoverable "in principle" because such services are assigned a *service type URI* facilitating their registration in *Discovery Service* instances.

Discovery Service (DS)

An *ID-WSF service* facilitating the registration, and subsequent discovery of, ID-WSF service instances [LibertyDisco], as indexed by *Principal identity*. See also *discoverable*.

Discovery Service Provider (DS Provider)

A *Web Service Provider* (WSP) implementing the *server*-side of the ID-WSF Discovery Service [LibertyDisco].

endpoint

A term used in [WSDLv1.1] — it is the short form of *protocol endpoint* — and which itself means an identified entity, at the current level of abstraction, to which a protocol message, of the same level of abstraction, may be sent. For example, at the Internet Protocol (IP) layer, an endpoint is represented by an IP address, and one may send an IP datagram (AKA a "message") to said endpoint. In contrast, at the HTTP layer, an endpoint is represented by a URL.

federate

To link or bind two or more entities together.

federation

(1) The act of establishing a relationship between two entities.

(2) An association comprising any number of service providers and identity providers.

final SASL response

The final `<SASLResponse>` message sent from the *server* to the *client* in an *authentication exchange* [LibertyAuthn].

gWS

See *generic web service*.

gWSP

See *generic Web Service Provider*.

generic web service (gWS)

A *generic web service* is defined by sense (1) of the *web service* definition.

generic Web Service Provider (gWSP)

A *generic Web Service Provider* (gWSP) an entity providing *generic web services*.

header, header block, header element

See *SOAP header block*.

ID-*

A shorthand designator referring to the Liberty ID-WSF, ID-FF, and ID-SIS specification sets. For example, one might say that the former specification sets are all part of the Liberty ID-* specification suite.

ID-* fault message

An *ID-* fault message* is a SOAP `<S:Fault>` element containing a `<Status>` element, with the attributes and attribute values of both elements configured as specified herein, or as specified in other specification(s) in the ID-WSF or ID-SIS specification sets.

ID-* header block

One of the header blocks defined in this specification, or defined in any of the other Liberty ID-* specification suite.

ID-* message

Equivalent to *ordinary ID-* message*.

ID-FF

The Identity Federation Framework (ID-FF) is the title for a subset of the Liberty specification suite which defines largely HTTP-based protocols for web single sign-on and identity federation [LibertyProtSchema].

ID-FF authentication assertion

See *authentication assertion*.

ID-PP

The "ID Personal Profile" is an *ID-SIS* -based service which can provide profile information regarding Principals, typically subject to policy established by said Principals [LibertyIDPP].

ID-SIS

Liberty Identity Service Interface specification set.

ID-SIS service

See *ID-SIS-based service*.

ID-SIS-based service

ID-SIS-based services are *identity services* typically built on *ID-WSF* — i.e. they are essentially *ID-WSF-based services* — and are often also built on the [LibertyDST] specification. [LibertyIDEP] and [LibertyIDPP] are examples of ID-SIS service specifications.

ID-WSF

Liberty Identity Web Services Framework specification set.

ID-WSF service

See *ID-WSF-based service*.

ID-WSF-based service

An *ID-WSF-based service* is an *identity service* that is at least *discoverable* in principle, and is based on [LibertySOAPBinding] and [LibertySecMech].

identity

The essence of an entity. One's *identity* is often described by one's characteristics, among which may be any number of identifiers.

A *Principal* may wield one or more identities. See also *Principal identity*.

Identity federation

Creating associations between a given *system entity's* identifiers or *accounts*.

Identity Provider (IdP)

A Liberty-enabled *system entity* that manages *identity* information on behalf of *Principals* and provides assertions of Principal *authentication* to other *providers*.

identity service

See *identity web service*.

identity web service

An abstract notion of a web service whose operations are indexed by *identity*, i.e. they are "discoverable" — see also *web service (2)*, *Discovery Service*, and [LibertyDisco]. Such a service might maintain information about, or on behalf of, Principals (as represented by their *identities*), or perform actions on behalf of Principals. Also known as an *identity service*. See also *web service*.

initial response

A [RFC2222] term referring to *authentication exchange* data sent by the *client* in the initial SASL request. It is used by a subset of SASL mechanisms. See Section 5.1 of [RFC2222].

initial SASL request

The initial `<SASLRequest>` message sent from the *client* to the *server* in an *authentication exchange* [LibertyAuthn].

IdP

See *Identity Provider*.

invocation identity

The identity of the *system entity* invoking a *service*.

LEC

See *Liberty-Enabled Client*.

LECP

See *Liberty-Enabled Client or Proxy* .

LEP

See *Liberty-Enabled Proxy*.

Liberty authentication assertion

See *authentication assertion*.

Liberty-enabled client (LEC)

An entity that has, or knows how to obtain, knowledge about the identity provider that the Principal wishes to use with the service provider.

Liberty-enabled client or proxy (LECP)

A Liberty-enabled client is a client that has, or knows how to obtain, knowledge about the identity provider that the Principal wishes to use with the service provider. A Liberty-enabled proxy is an HTTP proxy (typically a WAP gateway) that emulates a Liberty-enabled client.

Liberty-enabled Provider

An umbrella term referring to any *Provider* offering any ID-FF-, ID-WSF-, or ID-SIS-based services.

Liberty-Enabled Client and Proxy Profile

This profile specifies interactions between Liberty-enabled clients and/or proxies, service providers, and identity providers [LibertyBindProf].

Liberty-enabled Proxy (LEP)

A Liberty-enabled proxy is a HTTP proxy (typically a WAP gateway) that emulates a Liberty-enabled client.

Liberty-enabled User Agent or Device (LUAD)

A user agent or device that has specific support for one or more profiles of the Liberty specifications. It should be noted that although a standard web browser can be used in many Liberty-specified scenarios, it does not provide specific support for the Liberty protocols, and thus is not a LUAD.

No particular claims of specific functionality should be implied about a system entity solely based on its definition as a LUAD. Rather, a LUAD may perform one or more Liberty system entity roles as defined by the Liberty specifications it implements. For example, a LUAD-LECP is a user agent or device that supports the Liberty LECP profile, and a LUAD-DS would define a user agent or device offering a Liberty ID-WSF Discovery Service.

local session state

In the Liberty context, this term refers to a notion of *session state* "local" to, i.e. maintained by, a *provider*, with respect to an interaction with another *system entity*, typically a *user agent*. Note that the concrete techniques used to maintain session state vary; cookies [RFC2965], so-called "URL re-writing", and so-called "hidden form fields" are the most viable techniques in the HTTP, aka "web", world.

login

The act of a Principal proving their *identity* to a *system entity*, which typically establishes a *session*.

logout

The termination of a *session*.

LUAD

See *Liberty-enabled User Agent or Device*.

(LUAD-)WSC

A *Web Service Consumer* (WSC), that may or may not also be a *Liberty-enabled User Agent or Device*.

mechanism

A process or technique for achieving a result [Merriam-Webster].

MEP

see Message Exchange Pattern.

Message Exchange Pattern

A [SOAPv1.2] term for the overall notion of various patterns of message exchange between SOAP nodes. For example, request-reply and one-way are two *MEPs* used in this specification.

message thread

A *message thread* is a synchronous exchange of messages in a request-response *MEP* between two *SOAP nodes*. All the messages of a given message thread are "linked" via each message's `<Correlation>` header block `refToMessageID` attribute value being set, by the sender, from the previous successfully received message's `<Correlation>` header block `messageID` attribute value.

metadata

Definitional data that provides information about other data or *system entities* managed within an application or environment. In Liberty, *metadata* is *Provider* information that is necessary for interacting with Providers [LibertyMetadata].

network identity

An abstraction, consisting of a *Principal*'s global set of attributes, which is composed from a "union" of the Principal's *accounts*. See also *identity*.

Non-Transitive Proxy Capability

the ability to act for another entity based on Trusted Authority Policy. The capability is non-transferable.

opaque handle

An identifier that has meaning only in the context between a specific *identity provider* and specific *service provider*.

ordinary ID-* message

An *ordinary ID-* message* is a Liberty Identity Web Services Framework (ID-WSF) or Service Interface Specification (ID-SIS) message as defined in the [LibertyDST], [LibertyDisco], and [LibertyIDPP] specifications and others. It has the characteristics of being designed to be conveyed by essentially any transport or transfer protocol, notably SOAP [SOAPv1.1]. It is also known among the ID-* specifications as a *service request*, or an ID-WSF (service) request, or an ID-SIS (service) request.

PAOS

A Reversed HTTP binding for SOAP [SOAPv1.1] The primary difference from the normal HTTP binding for SOAP is that here a SOAP request is bound to a HTTP response and vice versa. "PAOS" is "SOAP" spelled backwards (pun intended).

PDP

See *Policy Decision Point*

PEP

See *Policy Enforcement Point*

permission

Privileges granted to a *system entity* with respect to operations that may be performed on some resource.

personally identifiable information (PII)

Any data that identifies or locates a particular person, consisting primarily of name, address, telephone number, e-mail address, bank accounts, or other unique identifiers such as Social Security numbers.

PII

See *personally identifiable information*.

policy

A logically defined, enforceable, and testable set of rules.

Policy Decision Point

A system entity that evaluates decision requests in light of applicable policy and information describing the requesting entity or entities and renders an authorization decision.

Policy Enforcement Point

A system entity that performs access control by making decision requests and enforcing authorization decisions. If the authorization decision is pushed to the PEP there will be no need for it to create a request.

Principal

Succinctly, a *principal* is a *system entity* whose *identity* can be authenticated. In Liberty usage, the term *Principal* is often synonymous with "natural person" or "user". A Principal's identity may be federated. Examples of Principals include individual users, groups of individuals, organizational entities e.g. corporations, or a component of the Liberty architecture.

Principal identity

An identity being wielded by a Principal, or that is mapped to a Principal in some fashion.

privacy

Proper handling of personal information throughout its life cycle, consistent with the preferences of the subject.

processing context

A *processing context* is the collection of specific circumstances under which a particular processing step or set of steps take place.

processing context facet

A *processing context facet* is an identified aspect, inherent or additive, of a *processing context*.

profile

Data comprising the broad set of attributes that may be maintained on behalf of an *system entity* (usually a *Principal*), over and beyond its various identifiers. At least some of this information (for example, addresses, preferences, card numbers) is typically provided by the Principal.

provider

A *provider* is a Liberty-enabled entity that performs one or more of the provider *roles* in the Liberty architecture, for example *Service Provider* or *Identity Provider*. Providers are identified in Liberty protocol interactions by their *Provider IDs* or optionally their *Affiliation ID* if they are a member of an affiliation(s) and are acting in that capacity.

Provider ID

A *Provider ID* identifies an entity known as a *provider*. It is schematically represented by the `providerID` attribute of the `<EntityDescriptor>` metadata element [LibertyMetadata].

proxy

(1) An entity authorized to act for another [Merriam-Webster].
(2) A *system entity* whose authenticated identity, according to the *recipient*, differs from that of the system entity making the invocation under consideration.

pseudonym

An arbitrary identifier assigned by the identity or service provider to identify a Principal to a given relying party so that the name has meaning only in the context of the relationship between the parties.

recipient

An entity that receives a message and acts as the message's ultimate processor.

RELs

See *Rights Expression Languages*.

receiver

A *role* taken by a *system entity* when it receives a message sent by another system entity. See also *SOAP receiver* in [SOAPv1.2].

relying party

The recipient of a message that relies on a request message and associated assertions to determine whether to provide a requested service.

requester

A *system entity* which sends a *service request* to a *provider*.

resource

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.

resource offering

The association of a resource and a service instance.

Rights Expression Language (REL)

A *Rights Expression Language* facilitates the expression of who are the "rights holders" for a resource, who is authorized to use a resource and their applicable permissions, and any constraints or conditions imposed on such permissions. They also may express "rights entities" and "rights transactions".

role

A function or part performed, especially in a particular operation or process [Merriam-Webster].

SAML (Security Assertion Markup Language)

An XML-based standard defining a means for making *assertions* about events, attributes, and policy evaluations concerning subjects [SAMLCore11]. In Liberty usage, SAML subjects are typically Principals.

SAML assertion

See *assertion*.

SAML Authority

An abstract system entity in the SAML domain model that issues assertions [SAMLGloss].

SASL

See *Simple Authentication and Security Layer*.

SASL mechanism

A *SASL mechanism* is an *authentication mechanism* that has been profiled for use in the context of *SASL* [RFC2222]. See [RFC2444] for a particular example of profiling an existing authentication mechanism — one-time passwords [RFC2289] — for use as a SASL mechanism. See also [LibertyAuthn].

security token

In Liberty, a *security token* is a collection of security-related information that is used to represent and substantiate a claim [LibertyIDWSFSecurityPrivacyGuidelines] [LibertySecMech].
Outside of Liberty, the term "security token" often refers to hardware-based devices, e.g. so-called "token cards". One should not confuse the latter and the former definitions. However, it is possible for some given *authentication mechanism* to employ token cards in the process of *authentication*.

sender

(1) A *role* donned by a *system entity* when it constructs and sends a message to another system entity. See also *SOAP sender* in [SOAPv1.2].
(1a) an initial SOAP *sender*. A sender is a *proxy* when its identity differs from the *invocation identity*.

server

A *role* donned by a *system entity* that provides a service in response to requests from other system entities called *clients* [RFC2828]. Note that in order to provide a service to clients, a server will often be both a *sender* and a *receiver*.

service

(1) A collection of *endpoints* designed to offer some service or to provide information [WSDLv1.1].
(2) Short form of *ID-WSF Service* or *ID-WSF-based Service*.

service discovery

The act of looking up a *service(s)* in the *Discovery Service*.

service instance

The physical instantiation of a service. A service instance is a running web service at a distinct endpoint.

Service Provider (SP)

(1)A *role* donned by *system entities*. In the Liberty architecture, *Service Providers* interact with other system entities primarily via vanilla HTTP.

(2) From a Principal's perspective, a Service Provider is typically a website providing services and/or goods.

service request

A *service request* is another term for an *ordinary ID-* message* sent by a *client*. Service request is also loosely equivalent to a "SOAP-bound (ordinary) ID-* message".

Service Type URI

ID-WSF-based services are assigned a *Service Type URI* as a part of each service's definition. The Service Type URI is a factor in *service discovery* [LibertyDisco].

session

[Merriam-Webster] defines *session* (in its sixth sense [sic]) as: "a meeting or period devoted to a particular activity" [as in "an Irish drinking session", Ed.]. Thus, a given interaction between some set of *system entities* may involve a notion of session, especially if one or more of the system entities maintain *session state*.

session state

If an interaction between *system entities* involves one or more of the system entities maintaining information pertaining to the interaction itself — such as who the other involved system entity(ies) are, when the interaction began, etc. — then there likely is an explicit notion of *session* and thus this information is termed *session state* information.

See also *local session state*.

Simple Authentication and Security Layer (SASL)

SASL [RFC2222] is an approach to modularizing protocol design such that the security design components, e.g. authentication and security layer mechanisms, are reduced to a uniform abstract interface. This facilitates a protocol's use of an open-ended set of security mechanisms, as well as a so-called "late binding" between implementations of the protocol and the security mechanisms' implementations. This late binding can occur at implementation- and/or deployment-time. The SASL specification also defines how one packages authentication and security layer mechanisms to fit into the SASL framework, where they are known as *SASL mechanisms*, as well as register them with the Internet Assigned Numbers Authority [IANA] for reuse.

single sign-on (SSO)

From a *Principal*'s perspective, *single sign-on* encompasses the capability to authenticate with some *system entity* — in the Liberty context, an *Identity Provider* — and have that authentication honored by other system entities, termed *Service Providers* in the Liberty context.

Note that upon authenticating with an Identity Provider, the Identity Provider typically establishes and maintains some notion of *local session state* between itself and the Principal's *user agent*. Service Providers may also maintain their own distinct local session state with a Principal's user agent.

Single Sign-On Service (SSO Service, SSOS)

An *ID-WSF-based service* providing *WSCs* a means of obtaining *ID-FF authentication assertions* [LibertyAuthn].

Single Sign-On Service Consumer (SSO Service Consumer, SSOS Consumer)

A *Web Service Consumer* (WSC) implementing the *client*-side of the ID-WSF Single Sign-On Service [LibertyAuthn].

Single Sign-On Service Provider (SSO Service Provider, SSOS Provider)

A *Web Service Provider* (WSP) implementing the *server*-side of the ID-WSF Single Sign-On Service [LibertyAuthn].

SOAP (Simple Object Access Protocol)

An XML envelope and data encoding technology used to communicate information and requests across the Web. It is typically considered the protocol used by Web services. It is actually an envelope encapsulation format that can be used with lower level Web protocols such as HTTP and FTP. See [SOAPv1.1] and [SOAPv1.2].

SOAP-bound ID-* message

A *SOAP message* conveying ID-WSF and perhaps ID-SIS header blocks and conveying either an *ordinary ID-* message* or an *ID-* fault message*. After being bound to SOAP, the resultant composite messages are referred to as an *Ordinary SOAP-bound ID-* Message* and a *SOAP-bound ID-* Fault Message*, respectively.

SOAP header block

A [SOAPv1.2] term meaning: An [element] used to delimit data that logically constitutes a single computational unit within the SOAP header. In [SOAPv1.1] these are known as simply *SOAP headers*, or simply *headers*. Liberty specifications use the SOAPv1.2 terminology.

SOAP-bound ID-* message

A *SOAP message* conveying ID-WSF and perhaps ID-SIS header blocks and conveying either an *ordinary ID-* message* or an *ID-* fault message*. After being bound to SOAP, the resultant composite messages are referred to as an *Ordinary SOAP-bound ID-* Message* and a *SOAP-bound ID-* Fault Message*, respectively.

SOAP node

A [SOAPv1.2] term describing *system entities* who are parties to SOAP-based message exchanges that are, for purposes of this specification, also the ultimate destination of the exchanged messages, i.e. *SOAP endpoints*. In [SOAPv1.1], SOAP nodes are referred to as *SOAP endpoints*, or simply *endpoints*. This specification uses the SOAPv1.2 terminology.

SP

See *Service Provider*.

SSL (Secure Sockets Layer Protocol)

An Internet protocol (originally developed by Netscape Communications, Inc.) that uses connection-oriented end-to-end encryption to provide data confidentiality service and data integrity service for traffic between a client (often a Web browser) and a server and that can optionally provide peer entity authentication between the client and the server. See *Transport Layer Security*. [RFC2828].

SSO

See *single sign-on*.

SSOS, SSO Service

See *Single Sign-On Service*.

SSOS Provider, SSO Service Provider

See *Single Sign-On Service Provider*.

system entity

An active element of a computer/network system. For example, an automated process or set of processes, a subsystem, a person or group of persons that incorporates a distinct set of functionality [SAMLGloss].

TLS (Transport Layer Security Protocol)

An evolution of the SSL protocol. The TLS protocol provides communications privacy over the Internet. The protocol allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery. See [RFC2246].

token

See *security token*.

trust circle

See *Circle of Trust*.

Trusted Authority

In Liberty, a Trusted Third Party (TTP) which issues and vouches for assertions, otherwise known as an *Identity Provider*.

Trusted Third Party

In general, a security authority or its agent, trusted by other entities with respect to security-related activities. In the context of Liberty, these other entities are, for example, *Principals* and *Service Providers*, and the trusted third party is typically the *Identity Provider(s)* involved in the particular interaction of interest.

TTP

See Trusted Third Party

URI (Uniform Resource Identifier)

A compact string of characters for identifying an abstract or physical resource. [RFC2396] defines the generic syntax of URIs. URNs and URLs are proper subsets of URIs.

URL (Uniform Resource Locator)

URLs identify resources via a representation of their primary access mechanism (e.g., their network location) rather than identifying the resource by name or by some other attributes of that resource [RFC2396]. URLs are a proper subset of URIs.

URN (Uniform Resource Name)

Persistent, location-independent, resource names with delegatable sub-namespaces, termed *Uniform Resource Name (URN) Namespaces* [RFC2141]. Liberty's URN Namespace is defined in [RFC3622]. URNs are a proper subset of URIs.

user agent

Software that a "natural person" interacts with directly. A user agent typically implements a user interface.

user interface

The controls (such as menus, buttons, prompts, etc.) and mechanisms (such as selection and focus) provided by, e.g., a user agent.

web service

(1) Generically, a *service* defined in terms of an *XML*-based protocol, often transported over *SOAP*, and/or a service whose instances, and possibly data objects managed therein, are concisely addressable via *URIs*. Such a *generic web service* (gWS) may be defined in various proprietary and/or standardized terms, e.g. security paradigms.

(2) As specifically used in Liberty specifications, usually in terms of *WSCs* and *WSPs*, it means a web service that's defined in terms of the *ID-** "stack", and thus utilizes [LibertySOAPBinding], [LibertySecMech], and is "discoverable" [LibertyDisco]. See also *identity web service*.

Web Service Consumer

A *role* donned by a *system entity* when it makes a request to a *web service*.

Web Services Description Language

A means to describe the interface of a Web service. See [WSDLv1.1].

Web Service Provider

A *role* donned by a *system entity* when it provides a *web service*.

WML (Wireless Markup Language)

A markup language based on XML and intended for use in specifying content and user interface for narrowband devices, including cellular phones and pagers.

WSC

See *Web Service Consumer*.

WSDL

See *Web Services Description Language*.

WSP

See *Web Service Provider*.

X.509 token

A *X.509 token* is a type of *security token* containing an X.509 public key certificate.

XML (eXtensible Markup Language)

A W3C technology for encoding information and documents for exchange over the Web. See [XML], [XMLCanon], [XMLDsig], [xmlenc-core], [Schema1] and [Schema2]

References

Normative

[LibertyAuthn] Hodges, Jeff, Aarts, Robert, eds. " Liberty ID-WSF Authentication Service Specification ," Version 1.1, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs/* *[http://www.projectliberty.org/specs/]*

[LibertyAuthnContext] Madsen, Paul, eds. "Liberty ID-FF Authentication Context Specification," Version 1.3, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs*

[LibertyBindProf] Cantor, Scott, Kemp, John, Champagne, Darryl, eds. "Liberty ID-FF Bindings and Profiles Specification," Version 1.2-errata-v2.0, Liberty Alliance Project (12 September 2004). *http://www.projectliberty.org/specs*

[LibertyDisco] Sergent, Jonathan, eds. "Liberty ID-WSF Discovery Service Specification," Version 1.2, Liberty Alliance Project (12 December 2004). *http://www.projectliberty.org/specs*

[LibertyDST] "Liberty ID-WSF Data Services Template Specification," Version 1.1, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs* Kainulainen, Jukka, Ranganathan, Aravindan, eds.

[LibertyIDEP] Kellomaki, Sampo, eds. (12 November 2003). "Liberty ID-SIS Employee Profile Service Specification," Version 1.0, Liberty Alliance Project *http://www.projectliberty.org/specs*

[LibertyIDPP] Kellomaki, Sampo, eds. "Liberty Identity Personal Profile Service Specification," Version 1.0, Liberty Alliance Project (12 November 2003). *http://www.projectliberty.org/specs*

[LibertyIDWSFSecurityPrivacyGuidelines] Landau, Susan, eds. "Liberty ID-WSF Security and Privacy Overview," Version 1.0, Liberty Alliance Project (8 October 2003). *http://www.projectliberty.org/specs*

[LibertyMetadata] Davis, Peter, eds. "Liberty Metadata Description and Discovery Specification," Version 1.1, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs*

[LibertySecMech] Ellison, Gary, eds. "Liberty ID-WSF Security Mechanisms," Version 1.2, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs*

[LibertySOAPBinding] Hodges, Jeff, Kemp, John, Aarts, Robert, eds. " Liberty ID-WSF SOAP Binding Specification ," Version 1.2, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs*

[LibertyProtSchema] Cantor, Scott, Kemp, John, eds. "Liberty ID-FF Protocols and Schema Specification," Version 1.2-errata-v3.0, Liberty Alliance Project (14 December 2004). *http://www.projectliberty.org/specs*

[RFC1510] Kohl, J., Neuman, , C., eds. (September 1993). "The Kerberos Network Authentication Service (V5)," RFC 1510, Internet Engineering Task Force *http://www.ietf.org/rfc/rfc1510.txt [September 1993].*

[RFC2119] Bradner, S., eds. "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, The Internet Engineering Task Force (March 1997). *http://www.ietf.org/rfc/rfc2119.txt [March 1997].*

[RFC2141] Moats, R., eds. (May 1997). "URN Syntax," RFC 2141, Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2141.txt [May 1997].*

[RFC2222] "Simple Authentication and Security Layer (SASL)," John G. Myers (October 1997). RFC 2222, Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2222.txt [October 1997].*

[RFC2246] Dierks, T., Allen, C., eds. (January 1999). "The TLS Protocol," Version 1.0 RFC 2246, Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2246.txt [January 1999].*

[RFC2396] Berners-Lee, T., Fielding, R., Masinter, L., eds. (August 1998). "Uniform Resource Identifiers (URI): Generic Syntax," RFC 2396, The Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2396.txt [August 1998].*

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T., eds. (June 1999). "Hypertext Transfer Protocol – HTTP/1.1," RFC 2616, The Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2616.txt [June 1999].*

[RFC2828] Shirey, R., eds. (May 2000). "Internet Security Glossary," RFC 2828., Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2828.txt [May 2000].*

[RFC3280] Housley, R., eds. (April 2002). "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," RFC 3280, The Internet Engineering Task Force *http://www.ietf.org/rfc/rfc3280.txt [April 2002].*

[RFC3622] Mealling, M., eds. (February 2004). " A Uniform Resource Name (URN) Namespace for the Liberty Alliance Project ," RFC 3622, The Internet Engineering Task Force *http://www.ietf.org/rfc/rfc3622.txt [Feb 2004].*

[SAMLBind11] Maler, Eve, Mishra, Prateek, Philpott, Rob, eds. (27 May 2003). "Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML) V1.1," OASIS Committee Specification, version 1.1, Organization for the Advancement of Structured Information Standards *http://www.oasis-open.org/committees/documents.php?wg_abbrev=security*

[SAMLCore11] Maler, Eve, Mishra, Prateek, Philpott, Rob, eds. (27 May 2003). "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1," OASIS Committee Specification, version 1.1, Organization for the Advancement of Structured Information Standards *http://www.oasis-open.org/committees/documents.php?wg_abbrev=security*

[SAMLGloss] Hodges, Jeff, Maler, Eve, eds. (05 November 2002). "Glossary for the OASIS Security Assertion Markup Language (SAML)," Version 1.0, OASIS Standard, Organization for the Advancement of Structured Information Standards *http://www.oasis-open.org/committees/security/#documents*

[Schema1] Thompson, Henry S., Beech, David, Maloney, Murray, Mendelsohn, Noah, eds. (May 2002). "XML Schema Part 1: Structures," Recommendation, World Wide Web Consortium *http://www.w3.org/TR/xmlschema-1/*

[Schema2] Biron, Paul V., Malhotra, Ashok, eds. (May 2002). "XML Schema Part 2: Datatypes," Recommendation, World Wide Web Consortium *http://www.w3.org/TR/xmlschema-2/*

[SOAPv1.1] "Simple Object Access Protocol (SOAP) 1.1," Box, Don, Ehnebuske, David , Kakivaya, Gopal, Layman, Andrew, Mendelsohn, Noah, Nielsen, Henrik Frystyk, Winer, Dave, eds. World Wide Web Consortium W3C Note (08 May 2000). *http://www.w3.org/TR/2000/NOTE-SOAP-20000508/*

[SOAPv1.2] "SOAP Version 1.2 Part 1: Messaging Framework," Gudgin, Martin, Hadley, Marc, Mendelsohn, Noah, Moreau, Jean-Jacques, Nielsen, Henrik Frystyk, eds. World Wide Web Consortium W3C Proposed Recommendation (07 May 2003). *http://www.w3.org/TR/2003/PR-soap12-part1-20030507/*

[WSDLv1.1] "Web Services Description Language (WSDL) 1.1," Christensen, Erik, Curbera, Francisco, Meredith, Greg, Weerawarana, Sanjiva, eds. World Wide Web Consortium W3C Note (15 March 2001). *http://www.w3.org/TR/2001/NOTE-wsdl-20010315*

[wss-saml] Hallam-Baker, Phillip, Kaler, Chris, Monzillo, Ronald, Nadalin, Anthony, eds. (December 1, 2004). Organization for the Advancement of Structured Information Standards *http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.0.pdf* "Web Services Security: SAML Token Profile," OASIS Standard V1.0 [OASIS 200412], [TrustInCyberspace] Schneider, Fred B., eds. " Trust in Cyberspace ," National Research Council (1999). *http://www.nap.edu/readingroom/books/trust/*

[XML] Bray, Tim, Paoli, Jean, Sperberg-McQueen, C.M., Maler, Eve, eds. (Oct 2000). "Extensible Markup Language (XML) 1.0 (Second Edition)," Recommendation, World Wide Web Consortium *http://www.w3.org/TR/2000/REC-xml-20001006*

[XMLDsig] Eastlake, Donald, Reagle, Joseph, Solo, David, eds. (12 Feb 2002). "XML-Signature Syntax and Processing," Recommendation, World Wide Web Consortium *http://www.w3.org/TR/xmldsig-core*

[XMLCanon] Boyer, John, Eastlake, Donald, Reagle, Joseph, eds. (18 July 2002). "Exclusive XML Canonicalization," Recommendation, World Wide Web Consortium *http://www.w3.org/TR/xml-exc-c14n*

[xmlenc-core] Eastlake, Donald, Reagle, Joseph, eds. (December 2002). "XML Encryption Syntax and Processing," W3C Recommendation, World Wide Web Consortium *http://www.w3.org/TR/xmlenc-core/*

Informative

[IANA] "The Internet Assigned Numbers Authority," *http://www.iana.org/*

[Merriam-Webster] "Merriam-Webster Dictionary," *http://www.merriam-webster.com/*

[RFC2289] "A One-Time Password System," N. Haller C. Metz P. Nessner M. Straw (February 1998). RFC 2289, Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2289.txt [February 1998]*.

[RFC2444] Newman, C., eds. (October 1998). "The One-Time-Password SASL Mechanism," RFC 2444, The Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2444.txt [October 1998]*.

[RFC2965] Kristol, D., Montulli, L., eds. (October 2000). "HTTP State Management Mechanism," RFC 2965., Internet Engineering Task Force *http://www.ietf.org/rfc/rfc2965.txt [October 2000]*.