# Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0

## OASIS Standard, 15 March 2005

**Document identifier:**

saml-glossary-2.0-os

**Location:**

http://docs.oasis-open.org/security/saml/v2.0/

**Editors:**

Jeff Hodges, Neustar
Rob Philpott, RSA Security
Eve Maler, Sun Microsystems

**SAML V2.0 Contributors:**

Conor P. Cahill, AOL
John Hughes, Atos Origin
Hal Lockhart, BEA Systems
Michael Beach, Boeing
Rebekah Metz, Booz Allen Hamilton
Rick Randall, Booz Allen Hamilton
Thomas Wisniewski, Entrust
Irving Reid, Hewlett-Packard
Paula Austel, IBM
Maryann Hondo, IBM
Michael McIntosh, IBM
Tony Nadalin, IBM
Nick Ragouzis, Individual
Scott Cantor, Internet2
RL 'Bob' Morgan, Internet2
Peter C Davis, Neustar
Jeff Hodges, Neustar
Frederick Hirsch, Nokia
John Kemp, Nokia
Paul Madsen, NTT
Steve Anderson, OpenNetwork
Prateek Mishra, Principal Identity
John Linn, RSA Security
Rob Philpott, RSA Security
Jahan Moreh, Sigaba
Anne Anderson, Sun Microsystems
Eve Maler, Sun Microsystems
Ron Monzillo, Sun Microsystems
Greg Whitehead, Trustgenix

**Abstract:**

This specification defines terms used throughout the OASIS Security Assertion Markup Language (SAML) specifications and related documents.

**Status:**

This is an **OASIS Standard** document produced by the Security Services Technical Committee. It was approved by the OASIS membership on 1 March 2005.

Committee members should submit comments and potential errata to the security-services@lists.oasis-open.org list. Others should submit them by filling out the web form located at http://www.oasis-open.org/committees/comments/form.php?wg_abbrev=security. The committee will publish on its web page (http://www.oasis-open.org/committees/security) a catalog of any changes made to this document.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights web page for the Security Services TC (http://www.oasis-open.org/committees/security/ipr.php).

# Table of Contents

# 1 Glossary

This normative document defines terms used throughout the OASIS Security Assertion Markup Language (SAML) specifications and related documents.

Some definitions are derived directly from external sources (referenced in an appendix), some definitions based on external sources have been substantively modified to fit the SAML context, and some are newly developed for SAML. Please refer to the external sources for definitions of terms not explicitly defined here.

Some definitions have multiple senses provided. They are denoted by (a), (b), and so on. References to terms defined elsewhere in this glossary are italicized.

Following are the defined terms used in the SAML specifications and related documents.

| Term | Definition |
|---|---|
| Access | To interact with a *system entity* in order to manipulate, use, gain knowledge of, and/or obtain a representation of some or all of a system entity's *resources*. [RFC2828] |
| Access Control | Protection of *resources* against unauthorized *access*; a process by which use of resources is regulated according to a *security policy* and is permitted by only authorized system entities according to that policy. [RFC2828] |
| Access Control Information | Any information used for *access control* purposes, including contextual information [X.812]. Contextual information might include source IP address, encryption strength, the type of operation being requested, time of day, etc. Portions of access control information may be specific to a request itself, some may be associated with the connection via which a request is transmitted, and others (for example, time of day) may be "environmental". [RFC2829] |
| Access Rights | A description of the type of authorized interactions a *subject* can have with a *resource*. Examples include read, write, execute, add, modify, and delete. [Taxonomy] |
| Account | Typically a formal business agreement for providing regular dealings and services between a *principal* and business service providers. |
| Account Linkage | A method of relating *accounts* at two different *providers* that represent the same *principal* so that the providers can communicate about the principal. Account linkage can be established through the sharing of *attributes* or through *identity federation*. |
| Active Role | A role that a *system entity* has donned when performing some operation, for example *accessing* a *resource*. |

| | |
|---|---|
| Administrative Domain | An environment or context that is defined by some combination of one or more administrative policies, Internet Domain Name registrations, civil legal entities (for example, individuals, corporations, or other formally organized entities), plus a collection of hosts, network devices and the interconnecting networks (and possibly other traits), plus (often various) network services and applications running upon them. An administrative domain may contain or define one or more security domains. An administrative domain may encompass a single site or multiple sites. The traits defining an administrative domain may, and in many cases will, evolve over time. Administrative domains may interact and enter into agreements for providing and/or consuming services across administrative domain boundaries. |
| Administrator | A person who installs or maintains a system (for example, a SAML-based security system) or who uses it to manage *system entities*, users, and/or content (as opposed to application purposes; see also *End User*). An administrator is typically affiliated with a particular *administrative domain* and may be affiliated with more than one administrative domain. |
| Affiliation, Affiliation Group | A set of *system entities* that share a single *namespace* (in the federated sense) of *identifiers* for *principals*. |
| Anonymity | The quality or state of being anonymous, which is the condition of having a name or identity that is unknown or concealed. [RFC2828] |
| Artifact | See SAML Artifact. |
| Assertion | A piece of data produced by a *SAML authority* regarding either an act of *authentication* performed on a *subject*, *attribute* information about the subject, or *authorization* data applying to the subject with respect to a specified *resource*. |
| Asserting Party | Formally, the *administrative domain* that hosts one or more *SAML authorities*. Informally, an instance of a *SAML authority*. |
| Attribute | A distinct characteristic of an object (in SAML, of a *subject*). An object's *attributes* are said to describe it. Attributes are often specified in terms of physical traits, such as size, shape, weight, and color, etc., for real-world objects. Objects in cyberspace might have attributes describing size, type of encoding, network address, and so on. Attributes are often represented as pairs of "attribute name" and "attribute value(s)", e.g. "foo" has the value 'bar', "count" has the value 1, "gizmo" has the values "frob" and "2", etc. Often, these are referred to as "attribute value pairs". Note that *Identifiers* are essentially "distinguished attributes". See also Identifier and *XML attribute*. |
| Attribute Authority | A *system entity* that produces *attribute assertions*. [SAMLAgree] |
| Attribute Assertion | An *assertion* that conveys information about *attributes* of a *subject*. |
| Authentication | To confirm a *system entity*'s asserted *principal identity* with a specified, or understood, level of confidence. [CyberTrust] [SAMLAgree] |

| | |
|---|---|
| Authentication Assertion | An *assertion* that conveys information about a successful act of *authentication* that took place for a *subject*. |
| Authentication Authority | A *system entity* that produces *authentication assertions*. [SAMLAgree] |
| Authorization | 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. [Taxonomy] |
| Authorization Decision | The result of an act of *authorization*. The result may be negative, that is, it may indicate that the *subject* is not allowed any *access* to the *resource*. |
| Authorization Decision Assertion | An *assertion* that conveys information about an *authorization decision*. |
| Back Channel | Back channel refers to direct communications between two *system entities* without "redirecting" messages through another system entity such as an HTTP client (e.g. A user agent). See also *front channel*. |
| Binding, Protocol Binding | Generically, a specification of the mapping of some given protocol's messages, and perhaps message exchange patterns, onto another protocol, in a concrete fashion. For example, the mapping of the SAML `<AuthnRequest>` message onto HTTP is one example of a binding. The mapping of that same SAML message onto SOAP is another binding. In the SAML context, each binding is given a name in the pattern "SAML xxx binding". |
| Credentials | Data that is transferred to establish a claimed *principal identity*. [X.800] [SAMLAgree] |
| End User | A natural person who makes use of resources for application purposes (as opposed to system management purposes; see *Administrator*, *User*). |
| Federated Identity | A *principal*'s *identity* is said to be *federated* between a set of *Providers* when there is an agreement between the providers on a set of *identifiers* and/or *attributes* to use to refer to the Principal |
| Federate | To link or bind two or more entities together [Merriam]. |
| Federation | This term is used in two senses in SAML:<br>a) The act of establishing a relationship between two entities [Merriam].<br>b) An association comprising any number of *service providers* and *identity providers*. |
| Front Channel | Front channel refers to the "communications channel" that can be effected between two HTTP-speaking servers by employing "HTTP redirect" messages and thus passing messages to each other via a user agent, e.g. a web browser, or any other HTTP client [RFC2616]. See also *back channel*. |

| | |
|---|---|
| Identifier | This term is used in two senses in SAML:<br>c) One that identifies [Merriam].<br>d) A data object (for example, a string) mapped to a *system entity* that uniquely refers to the system entity. A system entity may have multiple distinct identifiers referring to it. An identifier is essentially a "distinguished attribute" of an entity. See also *Attribute*. |
| Identity | The essence of an entity [Merriam]. One's identity is often described by one's characteristics, among which may be any number of identifiers. See also *Identifier*, *Attribute*. |
| Identity Defederation | The action occurring when *Providers* agree to stop referring to a *Principal* via a certain set of *identifiers* and/or *attributes*. |
| Identity Federation | The act of creating a *federated identity* on behalf of a Principal. . |
| Identity Provider | A kind of *service provider* that creates, maintains, and manages identity information for *principals* and provides principal authentication to other *service providers* within a *federation*, such as with web browser *profiles*. |
| Initial SOAP Sender | The SOAP sender that originates a SOAP message at the starting point of a SOAP message path. [WSGloss] |
| Login, Logon, Sign-On | The process whereby a *user* presents *credentials* to an *authentication authority*, establishes a *simple session*, and optionally establishes a *rich session*. |
| Logout, Logoff, Sign-Off | The process whereby a *user* signifies desire to terminate a *simple session* or *rich session*. |
| Markup Language | A set of *XML elements* and *XML attributes* to be applied to the structure of an XML document for a specific purpose. A markup language is typically defined by means of a set of *XML schemas* and accompanying documentation. For example, the *Security Assertion Markup Language* (SAML) is defined by two schemas and a set of normative SAML specification text. |
| Name Qualifier | A string that disambiguates an *identifier* that may be used in more than one *namespace* (in the federated sense) to represent different *principals*. |
| Namespace | This term is used in several senses in SAML:<br>e) (In discussing federated names) A domain in which an identifier is unique in representing a single principal.<br>f) (With respect to authorization decision actions) A URI that identifies the set of action values from which the supplied action comes.<br>g) (In XML) See *XML namespace*. |
| Party | Informally, one or more *principals* participating in some process or communication, such as receiving an *assertion* or accessing a *resource*. |

| | |
|---|---|
| Persistent Pseudonym | A privacy-preserving name *identifier* assigned by a *provider* to identify a *principal* to a given *relying party* for an extended period of time that spans multiple *sessions*; can be used to represent an *identity federation*. |
| Policy Decision Point (PDP) | A *system entity* that makes *authorization decisions* for itself or for other system entities that request such decisions. [PolicyTerm] For example, a SAML PDP consumes authorization decision requests, and produces *authorization decision assertions* in response. A PDP is an “authorization decision authority”. |
| Policy Enforcement Point (PEP) | A *system entity* that requests and subsequently enforces *authorization decisions*. [PolicyTerm] For example, a SAML PEP sends *authorization decision* requests to a PDP, and consumes the *authorization decision assertions* sent in response. |
| Principal | A *system entity* whose identity can be authenticated. [X.811] |
| Principal Identity | A representation of a principal’s identity, typically an *identifier*. |
| Profile | A set of rules for one of several purposes; each set is given a name in the pattern “xxx profile of SAML” or “xxx SAML profile”.<br>a) Rules for how to embed *assertions* into and extract them from a protocol or other context of use.<br>b) Rules for using SAML protocol messages in a particular context of use.<br>c) Rules for mapping attributes expressed in SAML to another attribute representation system. Such a set of rules is known as an “attribute profile”. |
| Provider | A generic way to refer to both *identity providers* and *service providers*. |
| Proxy | An entity authorized to act for another.<br>a) Authority or power to act for another.<br>b) A document giving such authority. [Merriam] |
| Proxy Server | A computer process that relays a protocol between client and server computer systems, by appearing to the client to be the server and appearing to the server to be the client. [RFC2828] |
| Pull | To actively request information from a *system entity*. |
| Push | To provide information to a *system entity* that did not actively request it. |
| Relying Party | A *system entity* that decides to take an action based on information from another system entity. For example, a SAML relying party depends on receiving *assertions* from an *asserting party* (a *SAML authority*) about a *subject*. |

| | |
|---|---|
| Requester, SAML Requester | A *system entity* that utilizes the SAML protocol to request services from another system entity (a *SAML authority*, a *responder*). The term “client” for this notion is not used because many system entities simultaneously or serially act as both clients and servers. In cases where the SOAP binding for SAML is being used, the SAML requester is architecturally distinct from the *initial SOAP sender*. |
| Resource | Data contained in an information system (for example, in the form of files, information in memory, etc), as well as:<br>a) A service provided by a system.<br>b) An item of system equipment (in other words, a system component such as hardware, firmware, software, or documentation).<br>c) A facility that houses system operations and equipment. [RFC2828]<br><br>SAML uses *resource* in the first two senses, and refers to resources by means of *URI references*. |
| Responder, SAML Responder | A *system entity* (a *SAML authority*) that utilizes the SAML protocol to respond to a request for services from another system entity (a *requester*). The term “server” for this notion is not used because many system entities simultaneously or serially act as both clients and servers. In cases where the SOAP binding for SAML is being used, the SAML responder is architecturally distinct from the *ultimate SOAP receiver*. |
| Role | Dictionaries define a role as “a character or part played by a performer” or “a function or position.” *System entities* don various types of roles serially and/or simultaneously, for example, active roles and passive roles. The notion of an Administrator is often an example of a role. |
| SAML Authority | An abstract *system entity* in the SAML domain model that issues *assertions*. See also *attribute authority*, *authentication authority*, and *policy decision point (PDP)*. |
| Security | A collection of safeguards that ensure the confidentiality of information, protect the systems or networks used to process it, and control access to them. Security typically encompasses the concepts of secrecy, confidentiality, integrity, and availability. It is intended to ensure that a system resists potentially correlated attacks. [CyberTrust] |
| Security Architecture | A plan and set of principles for an *administrative domain* and its security domains that describe the security services that a system is required to provide to meet the needs of its users, the system elements required to implement the services, and the performance levels required in the elements to deal with the threat environment. A complete security architecture for a system addresses administrative security, communication security, computer security, emanations security, personnel security, and physical security, and prescribes security policies for each. A complete security architecture needs to deal with both intentional, intelligent threats and accidental threats. A security architecture should explicitly evolve over time as an integral part of its administrative domain’s evolution. [RFC2828] |

| | |
|---|---|
| Security Assertion | An *assertion* that is scrutinized in the context of a security architecture. |
| Security Assertion Markup Language(SAML) | |
| | The set of specifications describing *security assertions* that are encoded in *XML*, *profiles* for attaching the assertions to various protocols and frameworks, the request/response protocol used to obtain the assertions, and *bindings* of this protocol to various transfer protocols (for example, SOAP and HTTP). |
| SAML Artifact | A small, fixed-size, structured data object pointing to a typically larger,variably-sized SAML protocol message. SAML artifacts are designed to be embedded in URLs and conveyed in HTTP messages, such as HTTP response messages with "3xx Redirection" status codes, and subsequent HTTP GET messages. In this way, a service provider may indirectly, via a user agent, convey a SAML artifact to another provider, who may subsequently dereference the SAML artifact via a direct interaction with the supplying provider, and obtain the SAML protocol message. Various characteristics of the HTTP protocol and user agent implementations provided the impetus for concocting this approach. The HTTP Artifact binding section of [SAMLBind] defines both the SAML Artifact format and the SAML HTTP protocol binding incorporating it. |
| Security Context | With respect to an individual SAML protocol message, the message's security context is the semantic union of the message's security header blocks (if any) along with other security mechanisms that may be employed in the message's delivery to a recipient. With respect to the latter, an examples are security mechanisms employed at lower network stack layers such as HTTP, TLS/SSL, IPSEC, etc.<br><br>With respect to a system entity, "Alice", interacting with another system entity, "Bob", a security context is nominally the semantic union of all employed security mechanisms across all network connections between Alice and Bob. Alice and Bob may each individually be, for example, a provider or a user agent. This notion of security context is similar to the notion of "security contexts" as employed in [RFC2743], and in the Distributed Computing Environment [DCE], for example. |
| Security Domain | An environment or context that is defined by security models and a *security architecture*, including a set of *resources* and set of *system entities* that are authorized to access the resources. One or more security domains may reside in a single *administrative domain*. The traits defining a given security domain typically evolve over time. [Taxonomy] |
| Security Policy | A set of rules and practices that specify or regulate how a system or organization provides *security services* to protect *resources*. Security policies are components of *security architectures*. Significant portions of security policies are implemented via security services, using *security policy expressions*. [RFC2828] [Taxonomy] |
| Security Policy Expression | A mapping of *principal identities* and/or *attributes* thereof with allowable actions. Security policy expressions are often essentially *access control* lists. [Taxonomy] |

| | |
|---|---|
| Security Service | A processing or communication service that is provided by a system to give a specific kind of protection to *resources*, where said resources may reside with said system or reside with other systems, for example, an *authentication* service or a PKI-based document attribution and authentication service. A security service is a superset of AAA services. Security services typically implement portions of *security policies* and are implemented via security mechanisms. [RFC2828] [Taxonomy] |
| Service Provider | A *role* donned by a *system entity* where the system entity provides services to *principals* or other system entities. |
| Session | A lasting interaction between *system entities*, often involving a *Principal*, typified by the maintenance of some state of the interaction for the duration of the interaction. |
| Session Authority | A *role* donned by a *system entity* when it maintains state related to *sessions*. *Identity providers* often fulfill this role. |
| Session Participant | A *role* donned by a *system entity* when it participates in a *session* with at least a *session authority*. |
| Site | An informal term for an *administrative domain* in geographical or DNS name sense. It may refer to a particular geographical or topological portion of an administrative domain, or it may encompass multiple administrative domains, as may be the case at an ASP site. |
| Subject | A *principal* in the context of a *security domain*. SAML assertions make declarations about *subjects*. |
| System Entity, 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. [RFC2828] [SAMLAgree] |
| Time-Out | A period of time after which some condition becomes true if some event has not occurred. For example, a *session* that is terminated because its state has been inactive for a specified period of time is said to “time out”. |
| Transient Pseudonym | A privacy-preserving *identifier* assigned by an *identity provider* to identify a *principal* to a given *relying party* for a relatively short period of time that need not span multiple *sessions*. |
| Ultimate SOAP Receiver | The SOAP receiver that is a final destination of a SOAP message. It is responsible for processing the contents of the SOAP body and any SOAP header blocks targeted at it. In some circumstances, a SOAP message might not reach an ultimate SOAP receiver, for example because of a problem at a SOAP intermediary. An ultimate SOAP receiver cannot also be a SOAP intermediary for the same SOAP message. [WSGloss] |
| User | A natural person who makes use of a system and its resources for any purpose [SAMLAgree] |

| | |
|---|---|
| Uniform Resource Identifier (URI) | A compact string of characters for identifying an abstract or physical *resource*. [RFC2396] URIs are the universal addressing mechanism for resources on the World Wide Web. Uniform Resource Locators (URLs) are a subset of URIs that use an addressing scheme tied to the resource's primary access mechanism, for example, their network "location". |
| URI Reference | A *URI* that is allowed to have an appended number sign (#) and fragment identifier. [RFC2396] Fragment identifiers address particular locations or regions within the identified resource. |
| XML | Extensible Markup Language, abbreviated XML, describes a class of data objects called XML documents and partially describes the behavior of computer programs which process them. [XML] |
| XML Attribute | An XML data structure that is embedded in the start-tag of an XML element and that has a name and a value. For example, the italicized portion below is an instance of an XML attribute:<br>`<Address *AddressID="A12345"*>…</Address>`<br>See also *attribute*. |
| XML Element | An XML data structure that is hierarchically arranged among other such structures in an XML document and is indicated by either a start-tag and end-tag or an empty tag. For example:<br>`<Address AddressID="A12345">`<br>`  <Street>105 Main Street</Street>`<br>`  <City>Springfield</City>`<br>`  <StateOrProvince>`<br>`  <Full>Massachusetts</Full>`<br>`  <Abbrev>MA</Abbrev>`<br>`  </StateOrProvince>`<br>`  <Post Code="56789"/>`<br>`</Address>` |
| XML Namespace | A collection of names, identified by a *URI reference*, which are used in XML documents as element types and attribute names. An XML namespace is often associated with an *XML schema*. For example, SAML defines two schemas, and each has a unique XML namespace. |
| XML Schema | The format developed by the World Wide Web Consortium (W3C) for describing rules for a *markup language* to be used in a set of XML documents. In the lowercase, a "schema" or "XML schema" is an individual instance of this format. For example, SAML defines two schemas, one containing the rules for XML documents that encode security assertions and one containing the rules for XML documents that encode request/response protocol messages. Schemas define not only XML elements and XML attributes, but also datatypes that apply to these constructs. |

# 2 References


**[CyberTrust]** Fred B. Schneider, editor. *Trust in Cyberspace*. Committee on Information Systems Trustworthiness, National Research Council, ISBN 0-309-06558-5, 1999. See http://www.nap.edu/readingroom/books/trust/ and glossary at http://www.nap.edu/readingroom/books/trust/trustapk.htm.

**[DCE]** *DCE 1.2.2 Introduction to OSF DCE.* The Open Group, Catalog number F201, ISBN 1-85912-182-9, Nov 1997. See http://www.opengroup.org/pubs/catalog/f201.htm.

**[Merriam]** *Merriam-Webster Collegiate Dictionary*. CDROM Version 2.5, 2000. An online version is available at http://www.m-w.com.

**[PolicyTerm]** A. Westerinen et al. *Terminology for Policy-Based Management*. IETF RFC 3198, November 2001. See http://www.ietf.org/rfc/rfc3198.txt.

**[RFC2396]** T. Berners-Lee et al. *Uniform Resource Identifiers (URI): Generic Syntax*. IETF RFC 2396, August 1998. See http://www.ietf.org/rfc/rfc2396.txt.

**[RFC2616]** R. Fielding et al. *Hypertext Transfer Protocol – HTTP/1.1*. IETF RFC 2616, June 1999. See http://www.ietf.org/rfc/rfc2616.txt.

**[RFC2743]** J. Linn. *Generic Security Service Application Program Interface Version 2, Update 1*, IETF RFC 2743, January 2000. See http://www.ietf.org/rfc/rfc2743.txt.

**[RFC2828]** R. Shirey. *Internet Security Glossary*. IETF RFC 2828, May 2000. See http://www.ietf.org/rfc/rfc2828.txt.

**[RFC2829]** M. Wahl et al. *Authentication Methods for LDAP*. IETF RFC 2829, May 2000. See http://www.rfc-editor.org/rfc/rfc2829.txt.

**[SAMLAgree]** *OASIS Security Services TC Use Case and Requirements Conference Call Consensus*. Consensus on the wording for this item occurred during one or more conference calls of the SAML Use Cases and Requirements subcommittee. Meeting minutes are available at http://lists.oasis-open.org/archives/security-use/.

**[SAMLBind]** S. Cantor et al. *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS SSTC, March 2005. Document ID saml-bindings-2.0-os. See http://www.oasis-open.org/committees/security/.

**[Taxonomy]** *Security Taxonomy and Glossary*. Lynn Wheeler, ongoing. See http://www.garlic.com/~lynn/secure.htm. See http://www.garlic.com/~lynn/ for the list of sources.

**[X.800]** *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture.* ISO 7498-2:1989, ITU-T Recommendation X.800 (1991). See http://www.itu.int/itudoc/itu-t/rec/x/x500up/x800.html.

**[X.811]** *Security Frameworks for Open Systems: Authentication Framework*. ITU-T Recommendation X.811 (1995 E), ISO/IEC 10181-2:1996(E). See http://www.itu.int/itudoc/itu-t/rec/x/x500up/x811.html.

**[X.812]** *Security frameworks for open systems: Access control framework*. ITU-T Recommendation X.812 (1995 E), ISO/IEC 10181-3:1996(E). See http://www.itu.int/itudoc/itu-t/rec/x/x500up/x812.html.

**[XML]** T. Bray et al. *Extensible Markup Language (XML) 1.0 (Third Edition)*. World Wide Web Consortium Recommendation, February 2004. See http://www.w3.org/TR/2004/REC-xml-20040204.

**[WSGloss]** H. Haas et al. *Web Services Glossary*, World Wide Web Consortium Note, February 2004. See http://www.w3.org/TR/ws-gloss/.

# Appendix A. Acknowledgments

The editors would like to acknowledge the contributions of the OASIS Security Services Technical Committee, whose voting members at the time of publication were:

- Conor Cahill, AOL
- John Hughes, Atos Origin
- Hal Lockhart, BEA Systems
- Mike Beach, Boeing
- Rebekah Metz, Booz Allen Hamilton
- Rick Randall, Booz Allen Hamilton
- Ronald Jacobson, Computer Associates
- Gavenraj Sodhi, Computer Associates
- Thomas Wisniewski, Entrust
- Carolina Canales-Valenzuela, Ericsson
- Dana Kaufman, Forum Systems
- Irving Reid, Hewlett-Packard
- Guy Denton, IBM
- Heather Hinton, IBM
- Maryann Hondo, IBM
- Michael McIntosh, IBM
- Anthony Nadalin, IBM
- Nick Ragouzis, Individual
- Scott Cantor, Internet2
- Bob Morgan, Internet2
- Peter Davis, Neustar
- Jeff Hodges, Neustar
- Frederick Hirsch, Nokia
- Senthil Sengodan, Nokia
- Abbie Barbir, Nortel Networks
- Scott Kiester, Novell
- Cameron Morris, Novell
- Paul Madsen, NTT
- Steve Anderson, OpenNetwork
- Ari Kermaier, Oracle
- Vamsi Motukuru, Oracle
- Darren Platt, Ping Identity
- Prateek Mishra, Principal Identity
- Jim Lien, RSA Security
- John Linn, RSA Security
- Rob Philpott, RSA Security
- Dipak Chopra, SAP
- Jahan Moreh, Sigaba
- Bhavna Bhatnagar, Sun Microsystems
- Eve Maler, Sun Microsystems
- Ronald Monzillo, Sun Microsystems

- Emily Xu, Sun Microsystems
- Greg Whitehead, Trustgenix

The editors also would like to acknowledge the following former SSTC members for their contributions to this or previous versions of the OASIS Security Assertions Markup Language Standard:

- Stephen Farrell, Baltimore Technologies
- David Orchard, BEA Systems
- Krishna Sankar, Cisco Systems
- Zahid Ahmed, CommerceOne
- Tim Alsop, CyberSafe Limited
- Carlisle Adams, Entrust
- Tim Moses, Entrust
- Nigel Edwards, Hewlett-Packard
- Joe Pato, Hewlett-Packard
- Bob Blakley, IBM
- Marlena Erdos, IBM
- Marc Chanliau, Netegrity
- Chris McLaren, Netegrity
- Lynne Rosenthal, NIST
- Mark Skall, NIST
- Charles Knouse, Oblix
- Simon Godik, Overxeer
- Charles Norwood, SAIC
- Evan Prodromou, Securant
- Robert Griffin, RSA Security (former editor)
- Sai Allarvarpu, Sun Microsystems
- Gary Ellison, Sun Microsystems
- Chris Ferris, Sun Microsystems
- Mike Myers, Traceroute Security
- Phillip Hallam-Baker, VeriSign (former editor)
- James Vanderbeek, Vodafone
- Mark O'Neill, Vordel
- Tony Palmer, Vordel

Finally, the editors wish to acknowledge the following people for their contributions of material used as input to the OASIS Security Assertions Markup Language specifications:

- Thomas Gross, IBM
- Birgit Pfitzmann, IBM

# Appendix B. Notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.