Netegrity

Security Assertions Markup Language (SAML)

The standard XML framework for secure information exchange

Netegrity White Paper

PUBLISHED: MAY 20, 2001

Executive Summary

SAML, the Security Assertions Markup Language, defines an eXtensible Markup Language (XML) framework for exchanging security information between business partners over the Internet.

SAML is being standardized at OASIS, the Organization for the Advancement of Structured Information Standards, an international consortium that creates interoperable industry specifications based on XML.

In December 2000, Netegrity originally created an OASIS industry-wide Technical Committee (TC) called Security Services (www.oasis-open.org/committees/security), which is responsible for submitting a draft specification of SAML to the OASIS Board members in the second half of 2001.

SAML is modeled after Netegrity's work in XML-based security for authentication and authorization, defined in the S2ML (Security Services Markup Language) specification, which was submitted to the OASIS Security Services TC as a base document for discussions.

SAML reuses S2ML's principles and architecture, with a few minor differences in scope and purpose to meet as many industry requirements and use cases as possible.

The primary goal of SAML is to enable interoperability between different systems that provide security services. The SAML specification does not define any new technology or approaches for authentication or authorization. Rather, it simply defines a common language for describing the information or outputs generated by these systems in XML.

Traditionally, security has been implemented within a single enterprise. However, companies are now partnering on the Web and dramatically expanding the scope and range of their e-business transactions. Typically, these transactions are initiated by users in business-to-consumer (B2C) scenarios or by XML-based document flows between services in business-to-business (B2B) applications. A transaction initiated at one site can be completed at a different site, requiring security information to be shared among the various Web sites involved in a transaction.

SAML delivers the following benefits:

- **Interoperability** - With SAML, e-marketplaces, service providers, and end-user companies of all sizes can now securely exchange information about users, Web services, and authorization information without requiring partners to change their current security solutions. SAML will become the common language for how different systems communicate data about security.

- **Open Solution** - SAML is designed to work with multiple, industry-standard transport protocols such as HTTP, SMTP, FTP, and others, as well as multiple XML document exchange frameworks such as SOAP, Biztalk, and ebXML.

- **Single Sign-On Across Sites** - SAML will enable Web users to travel across sites with their entitlements so that companies and partners in a trusted relationship can deliver single sign-on across Web sites, together with secure access to shared resources.

With SAML, the industry now has a common framework that can be used to exchange authentication, authorization, and profile information between multiple policy-based security systems, Java application servers, XML messaging frameworks, and multiple operating platforms.

SAML Scope and Purpose

The basic SAML objects are assertions (Authentication, Attribute). SAML assertions are submitted to, and generated by, trusted authorities using a request / response protocol. SAML assertions are embedded in industry-standard transport and messaging frameworks.

SAML defines a data format for:

- authentication assertions, including descriptions for authentication events,
- authorization attributes (i.e., the attributes that a service uses to make authorization decisions, such as an identifier, a group or role, or other user profile information).

SAML will support Web user sessions (message format to end a session due to logout by an end-user or service), although this feature may be available in a later version of the SAML standard due to time constraints.

SAML defines a message format and protocol for distributing SAML data among trusted partners in a business relationship. SAML's message protocol supports "pushing" data assertions from an authoritative source to a receiver. Likewise, SAML is designed to support "pulling" data assertions from an authoritative source to a receiver, thus allowing exchange of event notifications between partners in a trusted relationship.

SAML allows assertions to be shared over standard Internet protocols by binding SAML information to the following industry-standard transport and messaging frameworks:

- Commercial Web browsers: SAML assertions are communicated by a Web browser through cookies or URL strings.
- HTTP: SAML assertions are conveyed from a source Web site to a destination Web site via headers or HTTP POST.
- MIME: SAML assertions are packaged into a single MIME security package (combined with the message payload, e.g., a purchase order, a bank's line-of-credit statement, etc.).
- SOAP: SAML assertions are bound to the SOAP document's envelope header to secure the payload.
- ebXML: Provides a MIME-based envelope structure used to bind SAML assertions to the business payload.

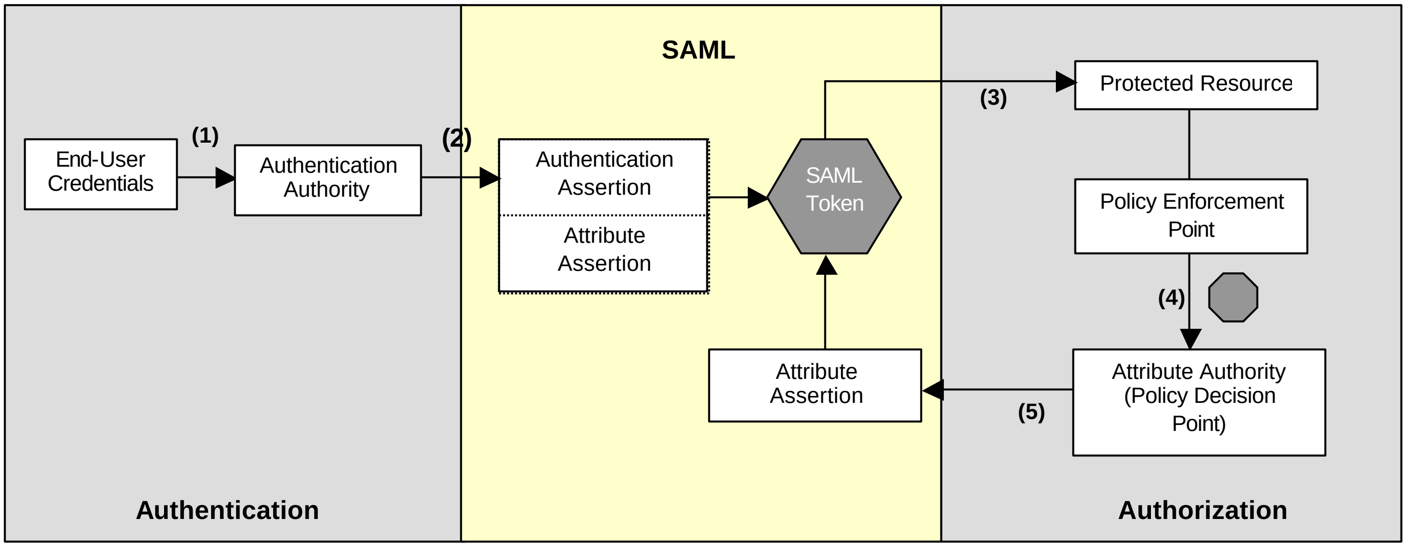
SAML does not define any new cryptographic technology or security models. Instead, the emphasis is on describing industry-standard security technologies using an XML-based syntax in the context of the Internet.

SAML does not provide for negotiation between partnering Web sites. A business agreement must be made as a prerequisite to the use of SAML in a trusted environment.

SAML does not define a data format for expressing authorization policies. This is left to the security system that implements SAML for authentication and authorization services.

SAML Overview

SAML's definition is based on use cases. The most basic use case involves end-user access to protected resources as shown in the figure below.



1. End-user submits credentials to Authentication Authority (any security engine or business application that is SAML-aware).
2. Authentication Authority asserts user's credentials against user directory and generates an Authentication Assertion together with one or more Attribute Assertions (e.g., role and other user profile information). End-user is now authenticated and identified by SAML assertions assembled in a token.
3. End-user attempts to access a protected resource using her SAML token.
4. Policy Enforcement Point (PEP) intercepts end-user request to protected resource and submits the end-user's SAML token (Authentication Assertion) to the Attribute Authority (which can also be any SAML-aware security engine or business application).
5. Attribute Authority or Policy Decision Point (PDP) makes a decision based on its policies. If it authorizes access to resource, it then generates an Attribute Assertion attached to the user's SAML token. The end-user's SAML token can be presented to trusted business partners affiliated in a single sign-on relationship.

SAML Use Case Scenarios

Single Sign-On to Trusted Environment

An end-user authenticates with a source Web site. The end-user then accesses a protected resource at another Web site, without having to re-authenticate herself at that Web site (the destination Web site).

In this model, the destination Web site can "pull" authentication information from the source Web site based on references or tokens provided by the end-user. The source Web site then acts as a credentials collector, authentication authority, and attribute authority. The destination Web site acts as a Policy Decision Point (PDP) and Policy Enforcement Point (PEP).

Likewise, the source Web site can "push" authentication information to the destination Web site, in which case the source Web site acts as a credentials collector, authentication authority, and attribute authority. The destination Web site acts as a PDP and PEP.

In this same scenario, a third-party security service can provide authentication assertions for the end-user. Multiple destination Web sites can then use the same authentication assertions to authenticate the end-user. In this case, the security service acts as a credentials collector, authentication authority, and attribute authority. The destination Web sites act as PDP and PEP.

Authorization Service

In this model, an end-user attempts to access a protected resource or Web service. The security controller for that resource (a PEP) checks the user's authorization to access the resource through a PDP. The PDP provides the authorization service to the PEP.

Typically, the end-user requests a dynamic resource from a Web server. The Web server passes the authentication information to a backend application which checks the user's authorization before processing the request. In this scenario, the security service acts as a credentials collector, authentication authority, attribute authority, and PDP. The backend application acts as a PEP.

Business-To-Business Transaction

This scenario describes partners involved in a transaction based on XML documents. In this model, each partner can be authenticated to their own security system, or partners can use the services of the third-party security engine, in which case partners exchange authentication data provided by their security systems to authenticate the transaction.

Alternatively, partners can enter a business transaction using a B2B exchange as an intermediary. The intermediary adds authentication and authorization data to orders as they go through the transaction process, giving additional points for the decisions made by the partners involved in the transaction. In this model, partners are principals that act as PEP. The partners' respective security systems act as credentials collector, authentication authority, attribute authority, and PDP. The exchange also acts as an authentication authority and attribute authority.

Sessioning

This scenario involves the description of a session which is maintained as the end-user navigates across the Web sites that are part of the single sign-on circle. The source Web site acts as a credentials collector, authentication authority, attribute authority, and session authority. The destination site(s) act as PDP and PEP.

SAML Structure

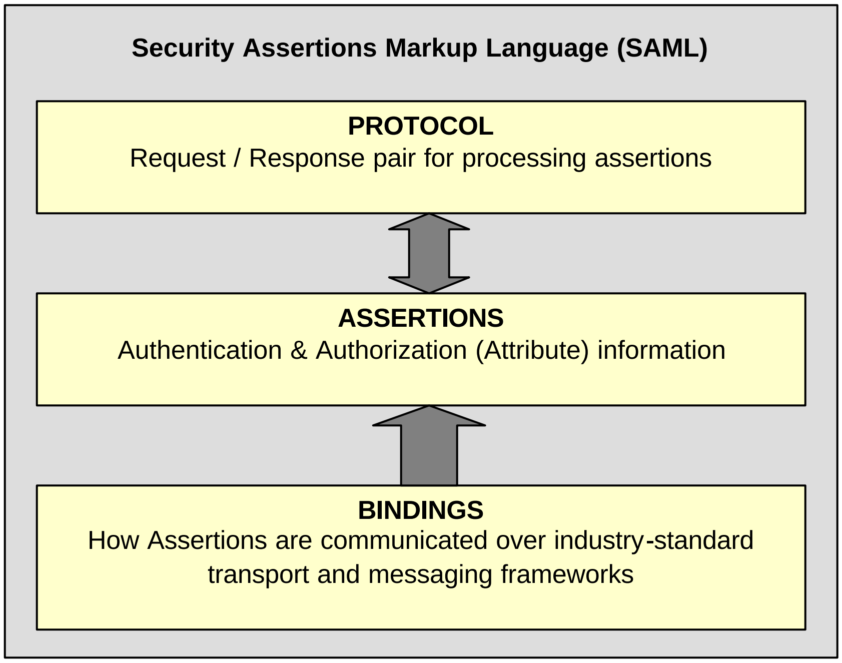
SAML assertions are encoded in a common XML Schema which includes basic information and claims. Basic information specifies a unique identifier used for the assertion name, date and time of issuance, and the time interval for which the assertion is valid. Claims are made by the assertion for authorization and key delegation applications.

An assertion may contain conditions and advice elements. For example, an assertion may be dependent on additional information from a validation service, and an assertion may be dependent on other assertions to be valid. Assertions may contain additional information as advice used to specify the assertions that were used to make a policy decision.

The SAML assertion package is defined to facilitate reuse in other specifications. For this reason, XML elements specific to the management of authentication and authorization data are expressed as claims.

SAML assertions are submitted to authentication and authorization decision points through a request / response protocol exchange (**`SAMLQuery`**, and **`SAMLQueryResponse`**)

SAML assertions are bound to the transport and messaging frameworks used in a particular application. For example, SAML defines how assertions are passed around using HTTP, SMTP, Java Message Services, etc., within a SOAP or ebXML messaging framework.



Security Assertions Markup Language (SAML)

PROTOCOL
Request / Response pair for processing assertions

ASSERTIONS
Authentication & Authorization (Attribute) information

BINDINGS
How Assertions are communicated over industry-standard transport and messaging frameworks

SAML Standardization Process

The OASIS Security Services Technical Committee, of which Negrity is a key member, will produce one or more Committee Specifications that cover core assertions, protocols, bindings, and a conformance suite.

The goal (subject to revision) is to publish a substantially complete set of Committee Specifications by 1 June 2001, and submit a Committee Specification to the OASIS membership for its approval by 1 September 2001.

Conclusion

Thanks to early work on S2ML (www.s2ml.org), Netegrity has been instrumental in creating the industry rally behind the standardization of SAML. The OASIS Technical Committee that Netegrity started includes over 80 members representing more than 30 companies.

Netegrity will provide products supporting the SAML standard as soon as the specification is endorsed by the OASIS Board of Members. SAML-aware components will soon complement SiteMinder, Netegrity's platform for securing e-business over the Internet.