OASIS Security Services Technical Committee

Security Assertions Markup Language

Issues List

Version 7

January 16, 2002

Hal Lockhart, Editor

Purpose

This document catalogs issues for the Security Assertions Markup Language (SAML) developed the Oasis Security Services Technical Committee.

Introduction

The issues list presented here documents issues brought up in response to draft documents as well as other issues mentioned on the security-use and security mailing lists, in conference calls, and in other venues.

Each issue is formatted according to the proposal of David Orchard to the general committee:

ISSUE:[Document/Section Abbreviation-Issue Number: Short name] Issue long description. Possible resolutions, with optional editor resolution Decision

The issues are informally grouped according to general areas of concern. For this document, the "Issue Number" is given as "#-##", where the first number is the number of the issue group.

Issues on this list were initially captured from meetings of the Use Cases subcommittee or from the security-use mailing list. They were refined to a voteable form by issue champions within the subcommittee, reviewed for clarity, and then voted on by the subcommittee. To achieve a higher level of consensus, each issue required a 75% super-majority of votes to be resolved. Here, the 75% number is of votes counted; abstentions or failure to vote by a subcommittee member did not affect the percentage.

At the second face-to-face meeting it was agreed to close all open issues relating to Use Cases and requirements accepting the findings of the sub committee, with the exception of issues that were specifically selected to remain open. This has been interpreted to mean that:

- Issues that received a consensus vote by the committee were settled as indicated.
- Issues that did not achieve consensus were settled by selecting the “do not add” option.

To make reading this document easier, the following convention has been adopted for shading sections in various colors.

Gray is used to indicate issues that were previously closed.

Blue is used to indicate issues that have just been closed in the most recent revision

Yellow is used to indicated issues which have recently been created or modified or are actively being debated.

Other open issues are not marked, i.e. left white.

Beginning with version 5 of this document, issues with lengthy write-ups, that have been closed “for some time” will be removed from this document, in order to reduce its overall size. The headings, a short description and resolution will be retained. All vote summaries from closed issues have also been removed.

Use Case Issues

Group 0: Document Format & Strategy

CLOSED ISSUE:[UC-0-01:MergeUseCases]

There are several use case scenarios in the Straw Man 1 that overlap in purpose. For example, there are several single sign-on scenarios. Should these be merged into a single use case, or should the multiplicity of scenarios be preserved?

Possible Resolutions:

1. Merge similar use case scenarios into a few high-level use cases, illustrated with UML use case diagrams. Preserve the detailed use case scenarios, illustrated with UML interaction diagrams. This allows casual readers to grasp quickly the scope of SAML, while keeping details of expected use of SAML in the document for other subcommittees to use.
2. Merge similar use case scenarios, leave out detailed scenarios.

Status: Closed, resolution 2 carries.

CLOSED ISSUE:[UC-0-02:Terminology]

Several subcommittee members have found the current document, and particularly the use case scenario diagrams, confusing in that they use either domain-specific terminology (e.g., "Web User", "Buyer") or vague, undefined terms (e.g., "Security Service.").

One proposal is to replace all such terms with a standard actor naming scheme, suggested by Hal Lockhart and adapted by Bob Morgan, as follows:

1. User
2. Authn Authority
3. Authz Authority
4. Policy Decision Point (PDP)
5. Policy Enforcement Point (PEP)

A counter-argument is that abstraction at this level is the point of design and not of requirements analysis. In particular, the real-world naming of actors in use cases makes for a more concrete goal for other subcommittees to measure against.

Another proposal is, for each use case scenario, to add a section that maps the players in the scenario to one or more of the actors called out above.

Possible Resolutions:

1. Replace domain-specific or vague terms with standard vocabulary above.
2. Map domain-specific or vague terms to standard vocabulary above for each use-case and scenario.
3. Don't make global changes based on this issue.

Status: Closed, resolution 3 carries

CLOSED ISSUE:[UC-0-03:Arrows]

Another problem brought up is that the use case scenarios have messages (arrow) between actors, but not much detail about the actual payload of the arrows. Although this document is intended for a high level of analysis, it has been suggested that more definite data flow in the interaction diagrams would make them clearer.

UC-1-08:AuthZAttrs, UC-1-09:AuthZDecisions, and UC-1-11:AuthNEvents all address this question to some degree, but this issue is added to state for a general editorial principle for the document.

Possible Resolutions:

1. Edit interaction diagrams to give more fine-grained detail and exact payloads of each message between players.
2. Don't make global changes based on this issue.

Status: Closed, resolution 2 carries.

Group 1: Single Sign-on Push and Pull Variations

CLOSED ISSUE:[UC-1-01:Shibboleth]

The Shibboleth security system for Internet 2 (http://middleware.internet2.edu/shibboleth/index.shtml) is closely related to the SAML effort.

[Text Removed to Archive]

If these issues, along with the straw man 2 document, have addressed the requirements of Shibboleth, then the subcommittee can address each issue on its own, rather than Shibboleth as a monolithic problem.

Possible Resolutions:

1. The above list of issues, combined with the straw man 2 document, address the requirements of Shibboleth, and no further investigation of Shibboleth is necessary.
2. Additional investigation of Shibboleth requirements are needed.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-02:ThirdParty]

Use case scenario 3 (single sign-on, third party) describes a scenario in which a Web user logs in to a particular 3rd-party security provider which returns an authentication reference that can be used to access multiple destination Web sites. Is this different than Use case scenario 1 (single sign-on, pull model)? If not, should it be removed from the use case and requirements document?

[Text Removed to Archive]

Possible Resolutions:

1. Edit the current third-party use case scenario to feature passing a third-party authentication assertion from one destination site to another.
2. Remove the third-party use case scenario entirely.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-03:ThirdPartyDoable]

Questions have arisen whether use case scenario 3 is doable with current Web browser technology. An alternative is using a Microsoft Passport-like architecture or scenario.

[Text Removed to Archive]

Possible Resolutions:

1. The use case scenario should be removed because it is unimplementable.
2. The use case scenario is implementable, and whether it should stay in the document or not should be decided based on other factors.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-1-04:ARundgrenPush]

Anders Rundgren has proposed on security-use an alternative to use case scenario 2 (single sign-on, push model). The particular variation is that the source Web site requests an authorization profile for a resource (e.g., the credentials necessary to access the resource) before requesting access.

[Text Removed to Archive]

Possible Resolutions:

1. Use this variation to replace scenario 2 in the use case document.
2. Add this variation as an additional scenario in the use case document.
3. Do not add this use case scenario to the use case document.

Status: Closed per F2F #2 3 carries

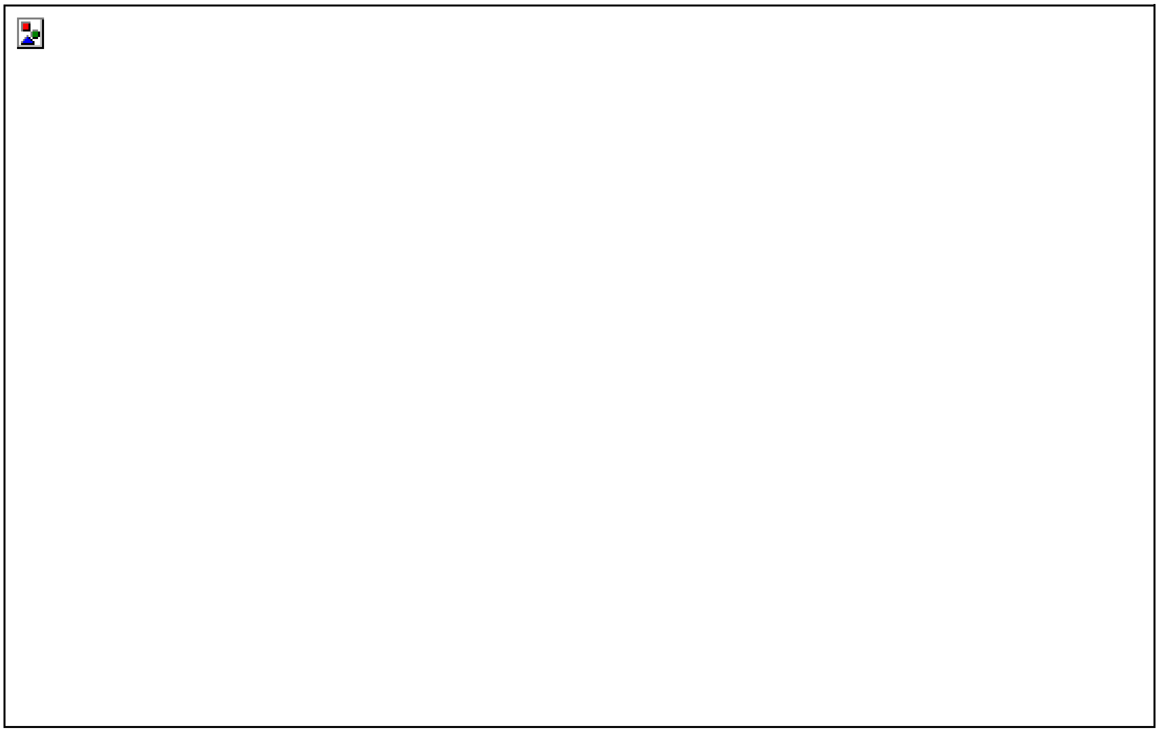
ISSUE:[UC-1-05:FirstContact]

A variation on the single sign on use case that has been proposed is one where the Web user goes directly to the destination Web site without authenticating with a definitive authority first.

A single sign-on use case scenario would be added as follows:

In this single sign-on scenario, the user does not first authenticate with their home security domain. Instead, they go directly to the destination Web site, first. The destination site must then redirect the user to a site they can authenticate at. The situation then continues as if in a single sign-on, push model scenario.

{PRIVATE "TYPE=PICT;ALT=Single Sign-on, Alternative Push
Model"}



Single Sign-on, Alternative Push Model

Steps:

1. Web user requests resource from destination Web site.
2. Destination Web site determines that the Web user is unauthenticated. It chooses the appropriate home domain for that user (deployment dependent), and redirects the Web user to that source Web site.
3. Web user authenticates with source Web site.
4. Source Web site provides user with authentication reference (AKA "name assertion reference"), and redirects user to destination Web site.
5. Web user requests destination Web site resource, providing authentication reference.
6. Destination Web site requests authentication document ("name assertion") from source Web site, passing authentication reference.
7. Source Web site returns authentication document.
8. Destination Web site provides resource to Web user.

Possible Resolutions:

1. Add this use case scenario to the use case document.
2. Do not add this use case scenario to the use case document.

Status: Voted, No conclusion

Voting Results

{PRIVATE}Date	23 Feb 2001
Eligible	18
Resolution 1	6
Resolution 2	3
Abstain	0

Bob Blakley said, " I agree that servers will have to do this, but it can easily be done by writing HTML with no requirement for us to provide anything in our specification."

CLOSED ISSUE:[UC-1-06:Anonymity]

What part does anonymity play in SAML conversations? Can assertions be for anonymous parties? Here, "anonymous" means that an assertion about a principal does not include an attribute uniquely identifying the principal (ex: user name, distinguished name, etc.).

A requirement for anonymity would state:

> [CR-1-06-Anonymity] SAML will allow assertions to be made about anonymous principals, where "anonymous" means that an assertion about a principal does not include an attribute uniquely identifying the principal (ex: user name, distinguished name, etc.).

Possible Resolutions:

1. Add this requirement to the use case and requirement document.
2. Do not add this requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-07:Pseudonymity]

What part do pseudonyms play in SAML conversations? Can assertions be made about principals using pseudonyms? Here, a pseudonym is an attribute in an assertion that identifies the principal, but is not the identifier used in the principal's home domain.

A requirement for pseudonymity would state:

> [CR-1-07-Pseudonymity] SAML will allow assertions to be made about principals using pseudonyms for identifiers.

Possible Resolutions:

1. Add this requirement to the use case and requirement document.
2. Do not add this requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-08:AuthZAttrs]

It's been pointed out that the concept of an "authentication document" used in the use case and requirements document does not clearly specify the inclusion of authz attributes. Here, authz attributes are attributes of a principal that are used to make authz decisions, e.g. an identifier, or group or role membership.

Since authz attributes are important and are required by [R-AuthZ], it has been suggested that the single sign-on use case scenarios specify when authz assertions are passed between actors.

Possible Resolutions:

1. Edit the use case scenarios to specify passing authz attributes with authentication documents.
2. Do not specify the passing of authz attributes in the use case scenarios.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-09:AuthZDecisions]

The current use case and requirements document mentions "Access Authorization" and "Access Authorization References." In particular, this data is a record of a authorization decision made about a particular principal performing a particular action on a particular resource.

It would be more clear to label this data as "AuthZ Decision Documents" to differentiate from other AuthZ data, such as AuthZ attributes or AuthZ policy. To this point, the mentions of "access authorization" would be changed, and a new requirement would be added as follows:

> [CR-1-09-AuthZDecision] SAML should define a data format for recording authorization decisions.

Possible Resolutions:

1. Edit the use case scenarios to use the term "authz decision" and add the [CR-1-09-

AuthZDecision] requirement.

2. Do not make these changes.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-10:UnknownParty]

The current straw man 2 document does not have a use case scenario for exchanging data between security services that are previously unknown to each other. For example, a relying party may choose to trust assertions made by an asserting party based on the signatures on the AP's digital certificate, or through other means.

[Text Removed to Archive]

Possible Resolutions:

1. Add this use case scenario to the use case document.
2. Do not add this use case scenario to the use case document.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-1-11:AuthNEvents]

It is not specified in straw man 2 what authentication information is passed between parties. In particular, specific information about authn events, such as time of authn and authn protocol are alluded to but not specifically called out.

The use case scenarios would be edited to show when information about authn events would be transferred, and the requirement for authn data would be edited to say:

> [CR-1-11-AuthN] SAML should define a data format for authentication assertions, including descriptions of authentication events.

Possible Resolutions:

1. Edit the use case scenarios to specifically define when authn event descriptions are transferred, and edit the R-AuthN requirement.
2. Do not change the use case scenarios or R-AuthN requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-1-12:SignOnService]

Bob Morgan suggests changing the title of use case 1, "Single Sign-on," to "Sign-on Service."

Possible Resolutions:

1. Make this change to the document.
2. Don't make this change.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-1-13:ProxyModel]

Irving Reid suggests an additional use case scenario for single sign-on, based on proxies.

[Text Removed to Archive]

Possible Resolutions:

1. Add this use case scenario to the document.
2. Don't make this change.

Status: Closed by explicit vote at F2F #2, 2 carries, however see UC-1-14

CLOSED ISSUE:[UC-1-14: NoPassThruAuthnImpactsPEP2PDP]

Stephen Farrell has argued that dropping PassThruAuthN prevents standardization of important functionality in a commonly used configuration.

The counter argument is the technical difficulty of implementing this capability, especially when both username/password and PKI AuthN must be supported.

Possible Resolutions:

1. Add this requirement to SAML 1.0
2. authorize a subgroup/task force to evaluate a suitable pass-through authN solution for eventual inclusion in V.next of SAML. If the TC likes the design once it is presented, it may choose to open up its scope to once again include pass-through authN in V1.0. Stephen is willing to champion this."
3. Do not add this requirement.

Status: Closed on May 15 telcon, 2 carries

Group 2: B2B Scenario Variations

CLOSED ISSUE:[UC-2-01:AddPolicyAssertions]

Some use cases proposed on the security-use list (but not in the straw man 1 document) use a concept of a "policy document." In concept a policy document is a statement of policy about a particular resource, such as that user "evanp" is granted "execute" privileges on file "/usr/bin/emacs." Another example may be that all users in domain "Acme.com" with role "backup administrator" may perform the "shutdown" method on resource "mail server," during non-business hours.

Use cases where policy documents are exchanged, and especially activities like security discovery as in UC-4-04:SecurityDiscovery, would require this type of assertion. If these use cases and/or services were adapted, the term "policy document" should be used. In addition, the following requirement would be added:

> [CR-2-01-Policy] SAML should define a data format for security policy about resources.

In addition, the explicit non-goal for authorization policy would be removed.

Another thing to consider is that the intended XACML group within Oasis is planning on working on defining a policy markup language in XML, and any work we do here could very well be redundant.

Possible Resolutions:

1. Remove the non-goal, add this requirement, and refer to data in this format as "policy documents."
2. Maintain the non-goal, leave out the requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-2-02:OutsourcedManagement]

A use case scenario provided by Hewlett Packard illustrates using SAML enveloped in a CIM/XML request. Should this scenario be included in the use case document?

[Text Removed to Archive]

Potential Resolutions:

1. Add this use-case scenario to the document.
2. Do not add this use-case scenario.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-2-03:ASP]

A use case scenario provided by Hewlett Packard illustrates using SAML for a secure interaction between an application service provider (ASP) and a client. Should this scenario be included in the use case document?

[Text Removed to Archive]

Potential Resolutions:

1. Add this use-case scenario to the document.

2. Do not add this use-case scenario.

Status: Closed per F2F #2, 2 carries

ISSUE:[UC-2-05:EMarketplace]

Zahid Ahmed proposes the following additional use case scenario for inclusion in the use case and requirements document.

Scenario X: E-Marketplace

{PRIVATE
"TYPE=PICT;ALT=EMarketplace"}

Fig X. EMarketplace.

Figure X: E-Marketplace Transaction.

A B2B Transaction involving buyers and suppliers that conduct trade via an e-marketplace that provides trading party authentication and authorization services, and other business services, in support of secure transaction and routing of business document exchanges between trading parties.

Steps:

1. A trading party (TP, e.g., buyer) creates a business document for subsequent transaction with another trading party (e.g., supplier) accessible via its e-marketplace.

2. The sending, i.e., transaction-initiating trading party (TP) application creates credential data to be authenticated by the authentication and security service operated by an e-marketplace.

3. The trading party application transaction client packages the XML-based credential data along with the other XML-based business document over a specific transport, messaging, and application protocol. Note: Credential data for login is not in SAML scope at the present time.

   Some examples of such (layered) protocols are following (but not limited to):

   - Secure transports: SSL and/or HTTPS
   - Messaging protocol: S/MIME and JMS.
   - Message Enveloping Formats: SOAP, etc.
   - B2B Application Protocol: ebXML, BizTalk, etc.

4. E-marketplace Authentication Service validates the TP Credential and creates a SAML authn assertion along with attribute assertions for the transaction-initiating TP.

   NOTE: The authentication protocol and service and message processing service that process SAML document instances are beyond the scope of the OASIS SAML Specification. However, it is included here mainly to highlight the transaction flow and is not defined as part of any SAML spec.

5. The E-marketplace Messaging Service then packages the AuthN Assertion and attribute assertions along with the original message payload into a tamper-proof envelope (i.e., S/MIME multi-part signed)

6. The resulting message envelope is transmitted to the target trading party (service provider).

7. The receiving trading party application extracts and processes the TP identity and authorization information available in the received envelope.

8. Receiving TP application then processes the business document of the sending TP.

9. Receiving TP sends back a response to sending TP via its e-marketplace by repeating Steps 1 through 5.

Possible Resolutions:

1. The above scenario should be added to the use cases document.

2. The above scenario should not be added to the document.

Status: Voted, No conclusion

Voting Results

{PRIVATE}Date	6 Apr 2001
Eligible	12
Resolution 1	7
Resolution 2	4

CLOSED ISSUE:[UC-2-06:EMarketplaceDifferentProtocol]

Zahid Ahmed has proposed that the following use case scenario be added to the use case and requirements document.

[Text Removed to Archive]

Possible Resolutions:

1. Add this scenario to the document.
2. This use case scenario should not be added to the document.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-2-07:MultipleEMarketplace]

Zahid Ahmed proposes the following use case scenario for inclusion in the document. This use case/issue is a variant of ISSUE# [UC-2-05].

[Text Removed to Archive]

Possible Resolutions:

1. Add this scenario to the document.
2. The above scenario should not be added to the document.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-2-08:ebXML]

Maryann Hondo proposed this use case scenario for inclusion in the use case document

[Text Removed to Archive].

Potential Resolutions:

1. Add this use case scenario to the use case and requirements document.

2. Do not add this scenario.

Status: Closed per F2F #2, 2 carries

Group 3: Sessions

[At F2F #2, it was agreed to charter a sub group to “do the prep work to ensure that logout, timein, and timeout will not be precluded from working with SAML later; commit to doing these other pieces "next" after 1.0.” Therefore all the items in this section have been closed with the notation “referred to sub group.”]

The purpose of the issues/resolutions in this group is to provide guidance to the rest of the TC as to the functionality required related to sessions. Some of the scenarios contain some detail about the messages which are transferred between parties, but the intention is not to require a particular protocol. Instead, these details are offered as a way of describing the functionality required. It would be perfectly acceptable if the resulting specification used different messages to accomplish the same functionality.

CLOSED ISSUE:[UC-3-01:UserSession]

Should the use cases of log-off and timeout be supported

[Text Removed to Archive].

Possible Resolutions:

1. Add this requirement and/or use cases to SAML.
2. Do not add this requirement and/or use cases.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-02:ConversationSession]

Is the concept of a session between security authorities separate from the concept of a user session? If so, should use case scenarios or requirements supporting security system sessions be supported? [DavidO: I don't understand this issue, but I have left in for backwards compatibility]. [DarrenP: I think this issue arose out of a misunderstanding/miscommunication on the mailing list and has been resolved. This is more of a formality to vote this one to a closed status.]

Possible Resolutions:

1. Do not pursue this requirement as it is not in scope.
2. Do further analysis on this requirement to determine what it is specifically.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-03:Logout]

Should SAML support transfer of information about application-level logouts (e.g., a principal intentionally ending a session) from the application to the Session Authority ?

Candidate Requirement:

> [CR-3-3-Logout] SAML shall support a message format to indicate the end of an application-level session due to logout by the principal.

Note that this requirement is implied by Scenario 1-3 (the second scenario 1-3 in straw man 3 - oops). This issue seeks to clarify the document by making the requirement explicit.

Possible Resolutions:

1. Add this requirement to SAML.
2. Do not add this requirement to SAML.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-05:SessionTermination]

For managing a SAML User Sessions, it may be useful to have a way to indicate that the SAML-level session is no longer valid. The logout requirement would invalidate a session based on user input. This requirement, for termination, would invalidate the SAML-level session based on other factors, such as when the user has not used any of the SAML-level sessions constituent application- level sessions for more than a set amount of time. Timeout would be an example of a session termination.

Candidate requirement:

> [CR-3-5-SessionTermination] SAML shall support a message format for timeout of a SAML-level session. Here, "termination" is defined as the ending of a SAML-level session by a security system not based on user input. For example, if the user has not used any of the application-level sub-sessions for a set amount of time, the session may be considered "timed out."

Note that this requirement is implied by Scenario 1-3, figure 6, specifically the last message labeled 'optionally delete/revoke session'. This issue seeks to clarify the document by making the requirement explicit.

Possible Resolutions:

1. Add this requirement to SAML.
2. Do not add this requirement and/or use cases.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-06:DestinationLogout]

Should logging out of an individual application-level session be supported? Advantage: allows application Web sites control over their local domain consistent with the model most widely implemented on the web. Disadvantage: potentially more interactions between the application and the Session Authority.

[Text Removed to Archive]

Possible Resolutions:

1. Add this scenario and requirement to SAML.
2. Do not add this scenario or requirement.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-07:Logout Extent]

What is the impact of logging out at a destination web site?

Possible Resolution:

1. Logout from destination web site is local to destination [DavidO recommendation]
2. Logout from destination web site is global, that is destination + source web sites.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-08:DestinationSessionTermination]

Having the Session Authority determine the timeout of a session is covered under [UC-3-5]. This issue covers the manner and extent to which systems participating in that session can initiate and control the timeout of their own sessions.

[Text Removed to Archive].

Possible Resolutions:

1. Add this scenario and requirement to SAML.
2. Do not add this scenario or requirement.

Status: Closed, referred to sub group

CLOSED ISSUE:[UC-3-09:Destination-Time-In]

In this scenario, a user has traveled from the source site (site of initial login) to some destination site. The source site has set a maximum idle-time limit for the user session, based on user activity at the source or destination site. The user stays at the destination site for a period longer than the source site idle-time limit; and at that point the user returns to the source site. We do not wish to have the user time-out at the source site and be re-challenged for authentication; instead, the user should continue to enjoy the original session which would somehow be cognizant of user activity at the destination site.

Candidate Requirement:

> [CR-3-9:Destination-TimeIn] SAML shall support destination system time-in.

Possible Resolutions:

1. Add this scenario and requirement to SAML.
2. Do not add this scenario or requirement to SAML.

Status: Closed, referred to sub group

Group 4: Security Services

CLOSED ISSUE:[UC-4-01:SecurityService]

Should part of the use case document be a definition of a security service? What is a security service and how is it defined?

Potential Resolutions:

1. This issue is now obsolete and can be closed as several securityservices (shared sessioning, PDP--PEP relationship) have been identified within SAML.
2. This issue should be kept open.

Status: Closed per F2F #2, 1 carries

CLOSED ISSUE:[UC-4-02:AttributeAuthority]

Should a concept of an attribute authority be introduced into the [SAML] use case document? What part does it play? Should it be added in to an existing use case scenario, or be developed into its own scenario?

The "attribute authority" terminology has already been introduced in the Hal/David diagrams and discussed by the use-case group. So this issue can be viewed as requiring more detail concerning the flows derived from the diagram to be introduced into the use-case document.

The following use-case scenario is offered as an instance:

(a) User authenticates and obtains an AuthN assertion. (b) User or server submits the AuthN assertion to an attribute authority and in response obtains an AuthZ assertion containing authorization attributes.

Potential Resolutions:

1. A use-case or use-case scenario similar to that described above should be added to SAML.
2. This issue is adequately addressed by existing use cases and does not require further elaboration within SAML.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-4-03:PrivateKeyHost]

A concept taken from S2ML. A user may allow a server to host a private key. A credentials field within an AuthN assertion identifies the server that holds the key. Should this concept be introduced into the [SAML] use case document? As a requirement? As part of an existing use case scenario, or as its own scenario?

The S2ML use-case scenario had the following steps:

1. User Jane (without public/private key pair) authenticates utilizing a trusted server X and receives an AuthN assertion. The trusted server holds a private/public key pair.The AuthN assertion received by Jane includes a field for the server X's public key.
2. User submits a business payload and said AuthN assertion to trusted server X. The trusted server "binds" the assertion to the payload using some form of digital signing and sends the composite package onto the next stage in the business flow.

Potential Resolutions:

1. A use-case or use-case scenario comprising steps 1 and 2 above should be added to the use-case document.
2. A requirement for supporting "binding" between AuthN assertions and business payloads thru digital signature be added to the use-case document.
3. This issue has been adequately addressed elsewhere; there is no need for any additions to the use-case document.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-4-04:SecurityDiscover]

UC-1-04:ARundgrenPush describes a single sign-on scenario that would require transfer of authorization data about a resource between security zones.Should a service for security discovery be part of the [SAML] standard?

Possible Resolutions:

1. Yes, a service could be provided to send authorization dataabout a service between security zones. This would require some sort of policy assertions (UC-2-01:AddPolicyAssertions).
2. No, this extends the scope of [SAML] too far. AuthZ in [SAML]should be concerned with AuthZ attributes of a principal, not of resources.

Status: Closed per F2F #2, Resolution 2 Carries

Group 5: AuthN Protocols

CLOSED ISSUE:[UC-5-01:AuthNProtocol]

Straw Man 1 explicitly makes challenge-response authentication a non-goal. Is specifying which types of authn are allowed and what protocols they can use necessary for this document? If so, what types and which protocols?

[Text Removed to Archive]

Possible Resolutions (not mutually exclusive):

1. The Non-Goal

   > "Challenge-response authentication protocols are outside the scope of the SAML"

   should be removed from the Strawman 3 document.

2. The following requirements should be added to the Strawman 3 document:

   > [CR-5-01-1-StandardCreds] SAML should provide a data format for credentials including those based on name-password, X509v3 certificates, public keys, X509 Distinguished name, and empty credentials.
   >
   > [CR-5-01-2-ExtensibleCreds] SAML The credentials data format must support extensibility in a structured fashion.

Status: Closed per F2F #2, 1 is not removed, 2 is not added, but see UC-1-14

CLOSED ISSUE:[UC-5-02:SASL]

Is there a need to develop materials within SAML that explore its relationship to SASL [SASL]?

Possible Resolutions:

1. Yes
2. No

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-5-03:AuthNThrough]

All the scenarios in Straw Man 1 presume that the user provides authentication credentials (password, certificate, biometric, etc) to the authentication system out-of-band.

Possible Resolutions (not mutually exclusive):

1. Should SAML be used directly for authentication? In other words should the SAML model or express one or more authentication methods or a framework for authentication?
2. Should this be explicitly stated as a non-goal?
3. Should the following statement be added to the non-goals section?

> [NO-Authn] Authentication methods or frameworks are outside the scope of SAML.

Status: Closed per F2F #2, Resolution 1 Fails, Resolution 2 Passes, Resolution 3 Fails

Group 6: Protocol Bindings

CLOSED ISSUE:[UC-6-01:XMLProtocol]

Should mention of a SOAP binding in the use case and requirements document be changed to a say "an XML protocol" (lower case, implying generic XML-based protocols)? Or "XML Protocol", the specific W3 RPC-like protocol using XML (http://www.w3.org/2000/xp/)?

Although SOAP is being reworked in favor of XP, the current state of XML Protocol is unknown. Requiring a binding to that protocol by June may not be feasible.

Per David Orchard, "There is no such deliverable as XML Protocol specification. We don't know when an XMLP 1.0 spec will ship. We can NEVER have forward references in specifications. When XMLP ships, we can easily change the requirements. [...] I definitely think we should mandate a SOAP 1.1 binding."

Possible Resolutions:

1. Change requirement for binding to SOAP to binding to XML Protocol.
2. Leave current binding to SOAP.
3. Remove mention of binding to either of these protocols.

Status: Closed per F2F #2, Resolution 2 Carries

Group 7: Enveloping vs. Enveloped

ISSUE:[UC-7-01:Enveloping]

SAML data will be transferred with other types of XML data not specific to authn and authz, such as financial transaction data. What should the relationship of the documents be?

One possibility is requiring that SAML allow for enveloping business-specific data within SAML. Such a requirement might state:

> [CR-7-01:Enveloping] SAML messages and assertions should be able to envelop conversation-specific XML data.

Note that this requirement is not in conflict with [CR-7-02:Enveloped]. They are mutually compatible.

Possible Resolutions:

1. Add this proposed requirement.
2. Do not add this proposed requirement.

Status: Voted, No Conclusion

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	9
Resolution 2	4
Abstain	1

ISSUE:[UC-7-02:Enveloped]

SAML data will be transferred with other types of XML data not specific to authn and authz, such as financial transaction data. What should the relationship of the documents be?

One possibility is requiring that SAML should be fit for being enveloped in other XML documents.

> [CR-7-02:Enveloped] SAML messages and assertions should be fit to be enveloped in conversation-specific XML documents.

Note that this requirement is not in conflict with [CR-7-01:Enveloping]. They are mutually compatible.

Possible Resolutions:

1. Add this proposed requirement.
2. Do not add this proposed requirement.

Status: Voted, Resolution 1 Carries

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	12
Resolution 2	2

Group 8: Intermediaries

CLOSED ISSUE:[UC-8-01:Intermediaries]

The use case scenarios in the S2ML 0.8a specification include one where an intermediary passes an S2ML message from a source party to a destination party. What is the part of intermediaries in an SAML conversation?

A requirement to enable passing SAML data through intermediaries could be phrased as follows:

> [CR-8-01:Intermediaries] SAML data structures (assertions and messages) will be structured in a way that they can be passed from an asserting party through one or more intermediaries to a relying party. The validity of a message or assertion can be established without requiring a direct connection between asserting and relying party.

Possible Resolutions:

1. Add this requirement to the document.
2. Do not add this requirement to the document.

Status: Closed per F2F #2, Resolution 1 Carries

ISSUE:[UC-8-02:IntermediaryAdd]

One question that has been raised is whether intermediaries can make additions to SAML documents. It is possible that intermediaries could add data to assertions, or add new assertions that are bound to the original assertions.

If we wanted to support allowing intermediaries to add data to SAML documents, the following use-case scenario could be added to the use case and requirements document:

In this use case scenario, two parties -- a buyer and a seller -- perform a transaction using a B2B exchange as an intermediary. The intermediary adds AuthN and AuthZ data to orders as they go through the system, giving additional points for decisions made by the parties.

{PRIVATE "TYPE=PICT;ALT=Intermediary

Add"}

Fig. X. Intermediary Add

Steps:

1. Buyer authenticates to Buyer Security System.

2. Buyer Security System provides a SAML AuthN assertion to Buyer, containing data about the authentication event and authorization attributes about the Buyer.

3. Seller authenticates to Seller Security System.
4. Seller Security System provides a SAML AuthN assertion to Seller, containing data about the authentication event and authorization attributes about the Seller.
5. Buyer requests authorization from Buyer Security System to submit a given order.
6. Buyer Security System provides a SAML AuthZ Decision assertion to Buyer, stating that Buyer is allowed to submit the order.
7. Buyer submits order to B2B Exchange, providing AuthN assertion and AuthZ decision assertion.
8. B2B exchange adds AuthN assertion data, specifying that the exchange authenticated the buyer (using the assertion).
9. B2B exchange adds AuthZ decision assertion data, stating that the Buyer is permitted to use the exchange to make this order.
10. B2B exchange submits order to Seller.
11. Seller validates the order, using the assertions.
12. Seller requests authorization from Seller Security System to fulfill a given order.
13. Seller Security System provides a SAML AuthZ Decision assertion to Seller, stating that Seller is allowed to fulfill the order.
14. Seller submits intention to fulfill the order to the B2B exchange, including AuthN assertions and AuthZ decision assertions.
15. B2B exchange adds AuthN data, specifying that it used the original SAML AuthN assertion to authenticate the Seller.
16. B2B exchange add AuthZ decision data, specifying that the seller is authorized to fulfill this order through the exchange.
17. B2B exchange sends the order fulfillment to the Buyer.
18. Buyer validates the order fulfillment based on AuthN assertion(s) and AuthZ decision assertion(s).

Possible Resolutions:

1. Add this use-case scenario to the document.
2. Don't add this use-case scenario.

Status: Voted, Resolution 1 Carries

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	11
Resolution 2	3

ISSUE:[UC-8-03:IntermediaryDelete]

Another issue with intermediaries is whether SAML must support allowing intermediaries to delete data from SAML documents.

If so, the following use-case scenario could be added to the use case document to illustrate.

Use Case Scenario X: Intermediary Delete

In this scenario, a buyer and a seller are using a B2B exchange to perform a transaction. The B2B exchange acts as an intermediary between the two parties. The exchange has an interest in not being disintermediated by the parties, so it modifies submitted SAML data to anonymize the buyer. This would prevent the seller from directly contacting the buyer without using the exchange.

{PRIVATE "TYPE=PICT;ALT=Intermediary Delete"}

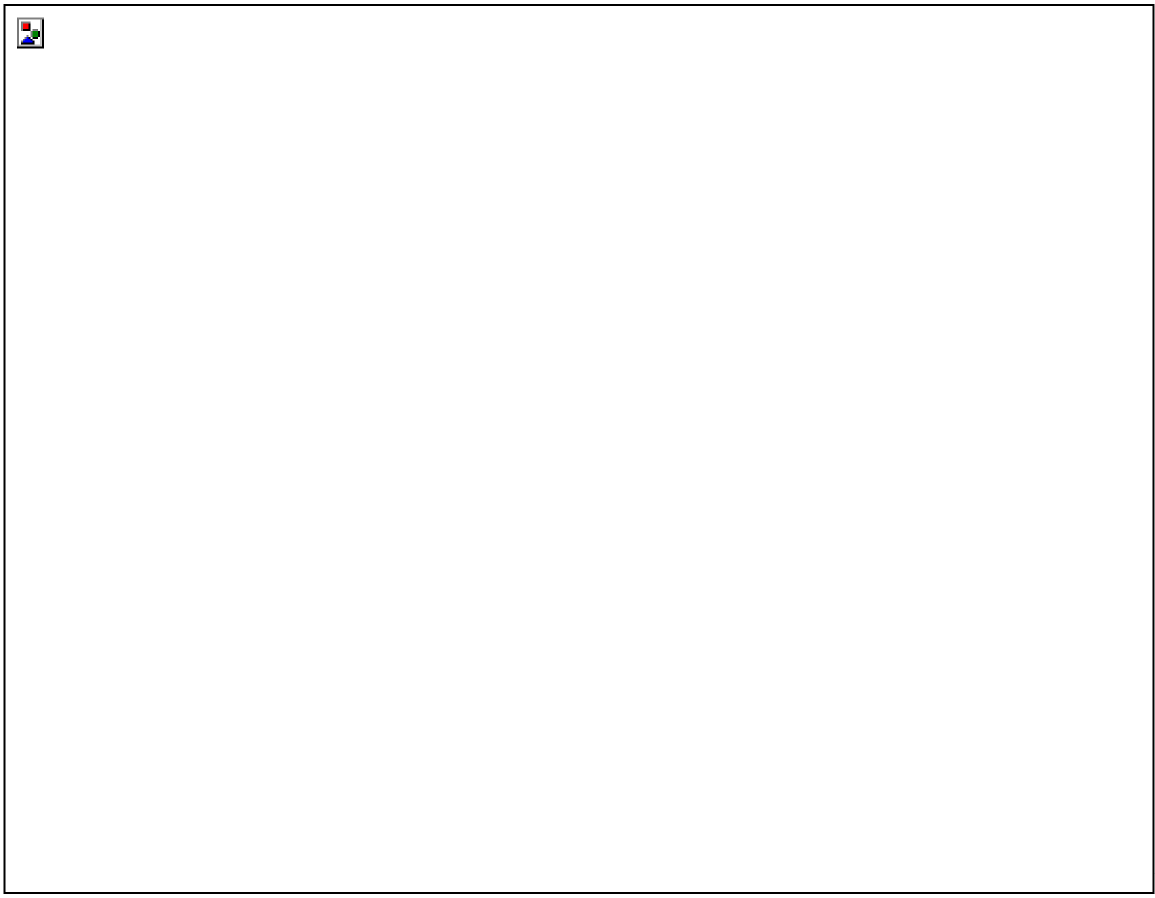


Fig. X. Intermediary Delete

Steps:

1. Buyer authenticates to Buyer Security System.
2. Buyer Security System provides a SAML AuthN assertion to Buyer, containing data about the authentication event and authorization attributes about the Buyer.
3. Buyer requests authorization from Buyer Security System to submit a given order.
4. Buyer Security System provides a SAML AuthZ Decision assertion to Buyer, stating that Buyer is allowed to submit the order.
5. Buyer submits order to B2B Exchange, providing AuthN assertion and AuthZ decision assertion.
6. B2B exchange anonymizes the order by removing identifying attributes from the SAML submitted by Buyer.
7. B2B exchange submits order to Seller.

Possible Resolutions:

1. Add this use-case scenario to the document.
2. Don't add this use-case scenario.

Status: Voted, No Conclusion

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	6
Resolution 2	8

ISSUE:[UC-8-04:IntermediaryEdit]

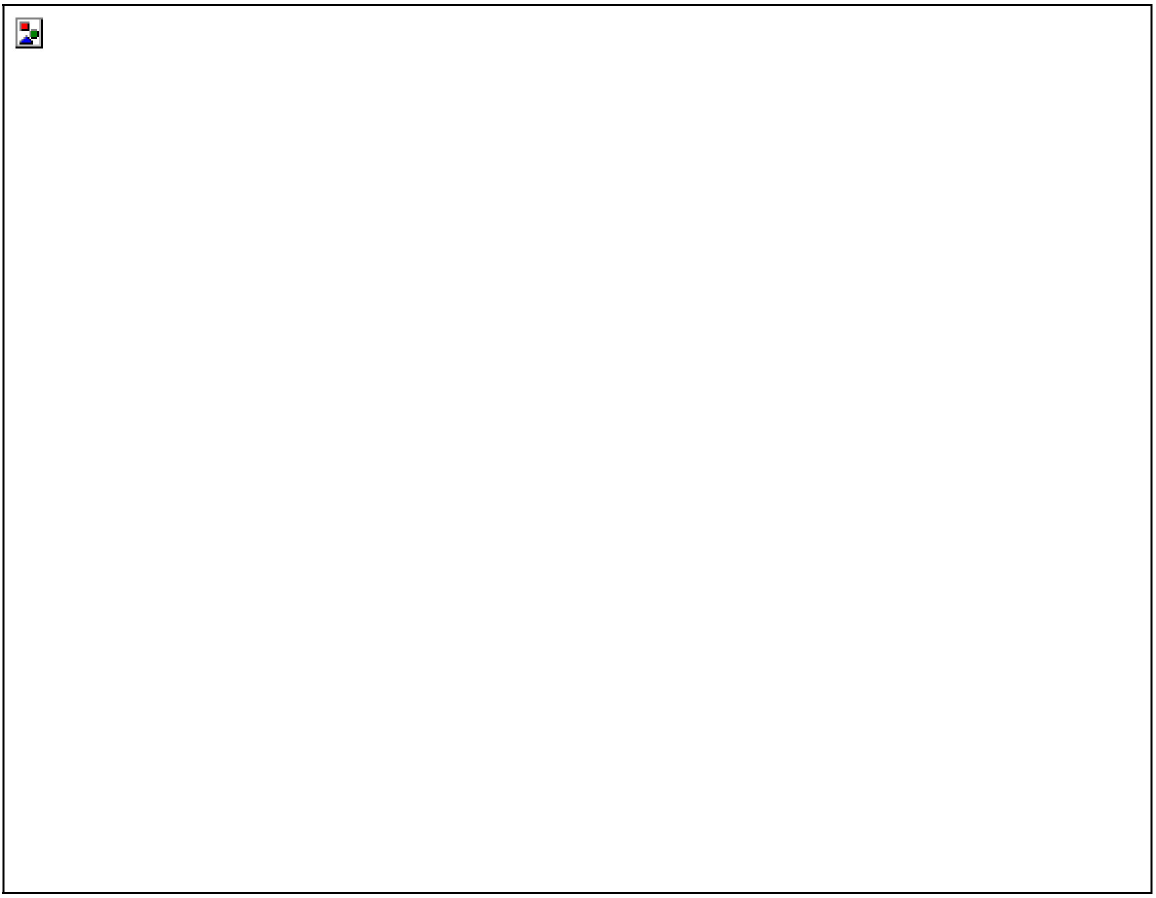
Similar to [UC-8-03:IntermediaryDelete] is the issue of whether SAML must support allowing intermediaries to edit or change SAML data as they pass it between parties.

If so, the following use-case scenario could be added to the use case document to illustrate.

Use Case Scenario X: Intermediary Edit

In this scenario, a buyer and a seller are using a B2B exchange to perform a transaction. The B2B exchange acts as an intermediary between the two parties. In this case, the buyer and seller use different vocabularies for expressing security concepts and also different vocabularies for domain concepts. The B2B exchange provides a translation before passing on SAML documents.

{PRIVATE "TYPE=PICT;ALT=Intermediary



Edit"}

Fig. X. Intermediary Edit

Steps:

1. Buyer authenticates to Buyer Security System.

2. Buyer Security System provides a SAML AuthN assertion to Buyer, containing data about the authentication event and authorization attributes about the Buyer. One AuthZ attribute is that the Buyer has a "role" of "purchase agent".

3. Buyer requests authorization from Buyer Security System to submit a given order.

4. Buyer Security System provides a SAML AuthZ Decision assertion to Buyer, stating that Buyer is allowed to submit the order. Specifically, it states that Buyer has the "purchase" privilege for the given order.

5. Buyer submits order to B2B Exchange, providing AuthN assertion and AuthZ decision assertion.

6. Based on registered settings of the Seller, the B2B exchange knows that Seller uses a different vocabulary than Buyer. For example, Seller has only group-based AuthZ, not

role-based. So it changes the "role" attribute to "group". Additionally, it knows that the Seller uses the term "buy" and not "purchase" for the privilege of making an order, so it translates that AuthZ information, too.

7. B2B exchange submits order to Seller.

Possible Resolutions:

1. Add this use-case scenario to the document.
2. Don't add this use-case scenario.

Status: Voted, No Conclusion

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	4
Resolution 2	10

ISSUE:[UC-8-05:AtomicAssertion]

One implicit assumption about SAML is that assertions will be represented as XML elements with associated digital signatures. Any additions, deletions or changes would make the signature on the assertion invalid. This would make it difficult for relying parties to determine the validity of the assertion itself, especially if it is received through an intermediary.

Thus, the implementation of assertions as element + signature would make [UC-8-02:IntermediaryAdd], [UC-8-03:IntermediaryDelete], and [UC-8-04:IntermediaryEdit] difficult to specify, if the idea is to actually modify the original assertions themselves. One possible solution is that some kind of diff or change structure could be added. Another possibility is that signatures on each individual sub-element of the assertion could be required, so that if the intermediary changes one sub-element the others remain valid. Neither of these is a clean solution.

However, if there's no goal of changing the sub-elements of the assertion, then it's possible to implement modifications. For example, [UC-8-02:IntermediaryAdd] can be implemented without breaking apart assertions. The B2B exchange could simply add its own assertions to the order, as well as the assertions provided by the buyer.

Deletion and edition could be implemented by simply replacing the assertions made by the buyer -- passing new AuthZ and AuthC assertions made and signed by the B2B exchange. These would incorporate elements from the assertions made by the Buyer Security System, but be signed by the B2B exchange.

There is semantic value to who makes an assertion, though. If the B2B exchange makes the assertion rather than the Buyer Security System, there is a different level of validity for the Seller.

Since assertion as element + signature is a very natural implementation, it may be good to express the indivisibility of the assertion as part of a non-goal. One such non-goal could be:

> [CR-8-05:AtomicAssertion] SAML does not need to specify a mechanism for additions, deletions or modifications to be made to assertions.

In addition, the use case scenarios should be edited to specifically point out that additions, deletions or modifications make changes to whole assertions, and not to parts of assertions.

Possible Resolutions:

1. Add this non-goal to the document, and change use case scenarios to specify that intermediaries must treat assertions as atomic.

2. Don't add this non-goal.

Status: Voted, Resolution 1 Carries

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	12
Resolution 2	2

Group 9: Privacy

ISSUE:[UC-9-01:RuntimePrivacy]

Should protecting the privacy of the user be part of the SAML conversation? In other words, should user consent to exchange of data be given at run time, or at the time the user establishes a relationship with a security system?

An example of runtime privacy configuration would be use case scenario described in [UC-1-04:ARundgrenPush]. Because this scenario has been rejected by the use cases and requirement group, it makes sense to phrase this as a non-goal of SAML, rather than as a requirement.

> [CR-9-01:RuntimePrivacy] SAML does not provide for subject control of data flow (privacy) at run-time. The determination of privacy policy is between the subject and security authorities and should be determined out-of-band, for example, in a privacy agreement.

Possible Resolutions

1. Add this proposed non-goal.
2. Do not add this proposed non-goal.

Status: Voted, No Conclusion

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	9
Resolution 2	4

ISSUE:[UC-9-02:PrivacyStatement]

Important private data of end users should be shared as needed between peers in an SAML conversation. In addition, the user should have control over what data is exchanged. How should the requirement be expressed in the use case and requirements document?

One difficulty is that, if run-time privacy is out of scope per UC-9-01:RuntimePrivacy, it's difficult to impose a privacy requirement on eventual implementers. Especially considering that our requirements doc is for the specification itself, and not for implementers. In addition, specifications rarely proscribe guiding principles that cannot be expressed in the specified technology itself.

One statement suggested by Bob Morgan is as follows:

> [CR-9-02-3-DisclosureMorgan] SAML should support policy-based disclosure of subject security attributes, based on the identities of parties involved in an authentication or authorization exchange.

Another, by Bob Blakley:

> [CR-9-02-2-DisclosureBlakley] SAM should support *restriction of* disclosure of subject security attributes, *based on a policy stated by the subject*. *This policy might be* based on the identities of parties involved in an authentication or authorization exchange.

A final one, by Prateek Mishra:

> [CR-9-02-4-DisclosureMishra] An AP should only release credentials for a subject to an RP if the subject has been informed about this possibility and has assented. The exact mechanism and format for interaction between an AP and a subject concerning such privacy issues is outside the scope of the specification.

Comment by David Orchard:

"My concerns about all of the disclosure requirements, is that I cannot see how any piece of software could be tested for conformance. In the case of Blakely style, "SAM should support *restriction of* disclosure of subject security attributes, *based on a policy stated by the subject*", how do I write a conformance test that verifes:

- what are allowable and non-allowable restrictions?
- How do I test that an non-allowable restriction hasn't been made?
- How do I verify that a subject has stated a policy?
- How can a subject state a policy?"

Possible Resolutions

1. Add [CR-9-02-3-DisclosureMorgan] as a requirement.
2. Add [CR-9-02-2-DisclosureBlakley] as a requirement.
3. Add [CR-9-02-4-DisclosureMishra] as a requirement.
4. Add none of these as requirements.

Status: Voted, No Conclusion

Voting Results

{PRIVATE}Date	27 Mar 2001
Eligible	15
Resolution 1	4
Resolution 2	0
Resolution 3	4
Resolution 4	7

Group 10: Framework

CLOSED ISSUE:[UC-10-01:Framework]

Should SAML provide a framework that allows delivery of security content negotiated out-of-band? A typical use case is authorization extensions to the core SAML constructs. The contra-position is to rigidly define the constructs without allowing extension.

A requirement already exists in the SAML document for extensibility: [R-Extensible] SAML should be easily extensible. Therefore, the change that voting on this issue would make would be to remove rather than add a requirement.

Possible Resolutions:

1. Remove the extensibility requirement.

2. Leave the extensibility requirement.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-10-02:ExtendAssertionData]

Assertions are the "nouns" of SAML. One way to extend SAML is to allow additional elements in an assertion besides the ones specified by SAML. This could be used to add additional attributes about a subject, or data structured under another namespace.

A requirement that captures this functionality would be:

> [CR-10-02:ExtendAssertionData] The format of SAML assertions should allow the addition of arbitrary XML data as extensions.

Possible Resolutions:

1. Add requirement [CR-10-02:ExtendAssertionData].

2. Do not add this requirement.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-10-03:ExtendMessageData]

Similarly to [UC-10-02], it would be useful to allow additional data to SAML messages. Either defined SAML assertions, or arbitrary XML, could be attached.

A potential requirement to add this functionality would be:

> [CR-10-03:ExtendMessageData] The format of SAML messages should allow the

addition of arbitrary XML data, or SAML assertions not specified for that message type, as extensions.

Possible Resolutions:

1. Add requirement [CR-10-03:ExtendMessageData].
2. Do not add this requirement.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-10-04:ExtendMessageTypes]

It's common in protocol definitions that real-world implementations require additional message types. For example, a system handling a request for authorization that is taking a long time might send a <KeepWaiting> or <AskAgainLater> message to the requester.

Many protocols explicitly allow for a mechanism for adding extended message types in their specification. We may want to require that SAML also allow for extended message types in the specification. One requirement may be:

> [CR-10-04:ExtendMessageTypes] The SAML protocol will explicitly allow for additional message types to be defined by implementers.

Note that this is different from [UC-10-03:ExtendMessageData]. That issue is about adding extended data to existing message types in the protocol. This issue is about adding new message types entirely.

Also note that adding this requirement would strongly favor [CR-10-07-1], to allow interoperability.

Possible Resolutions:

1. Add requirement [CR-10-04:ExtendMessageTypes].
2. Do not add this requirement.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-10-05:ExtendAssertionTypes]

As with [UC-10-04], it may be useful to add extended assertions to a SAML conversation. As an admittedly stretched example, an implementer may choose to add auditing to the SAML specification, and therefore define one or more <AuditAssertion> types.

[Text Removed to Archive]

Possible Resolutions:

1. Add requirement [CR-10-05:ExtendAssertionTypes].
2. Do not add this requirement.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-10-06:BackwardCompatibleExtensions]

Because SAML is an interoperability standard, it's important that custom extensions for SAML messages and/or assertions be compatible with standard SAML implementations. For this reasons, extensions should be clearly recognizable as such, marked with flags to indicate whether processing should continue if the receiving party does not support the extension.

One possible requirement for this functionality is the following:

> [CR-10-06-BackwardCompatibleExtensions] Extension data in SAML will be clearly identified for all SAML processors, and will indicate whether the processor should continue if it does not support the extension.

Possible Resolutions:

1. Add requirement [CR-10-06-BackwardCompatibleExtensions].
2. Do not add this requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-10-07:ExtensionNegotiation]

Many protocols allow a negotiation phase between parties in a message exchange to determine which extensions and options the other party supports. For example, HTTP 1.1 has the OPTIONS method, and ESMTP has the EHLO command.

Since this is a fairly common design model, it may be useful to add such a feature to SAML. One option is to add a requirement for extension negotiation:

> [CR-10-07-1:ExtensionNegotiation] SAML protocol will define a message format for negotiation of supported extensions.

However, this may unnecessarily complicate the SAML protocol. Because negotiation is a common design, it may be a good idea to have a clarifying non-goal in the requirements document:

> [CR-10-07-2:NoExtensionNegotiation] SAML protocol does not define a message format for negotiation of supported extensions.

Possible Resolutions:

1. Add requirement [CR-10-07-1:ExtensionNegotiation].
2. Add non-goal [CR-10-07-2:NoExtensionNegotiation].
3. Add neither the requirement nor the non-goal.

Status: Closed per F2F #2, 3 carries

Group 11: AuthZ Use Case

CLOSED ISSUE:[UC-11-01:AuthzUseCase]

Use Case 2 in Strawman 3 (http://www.oasis-open.org/committees/security/docs/draft-sstc-use-strawman- 03.html) describes the use of SAML for the conversation between a Policy Enforcement Point (PEP) and a Policy Decision Point (PDP), in which the PEP sends a request describing a particular action (such as 'A client presenting the attached SAML data wishes to read http://foo.bar/index.html'), and the PDP replies with an Authorization Decision Assertion instructing the PEP to allow or deny that request.

Possible Resolutions:

1. Continue to include this use case.
2. Remove this use case.

Status: Closed per F2F #2, Resolution 1 Carries

Group 12: Encryption

[Text Removed to Archive]

CLOSED ISSUE:[UC-12-01:Confidentiality]

Add the following requirement:

> [R-Confidentiality] SAML data should be protected from observation by third parties or untrusted intermediaries.

Possible Resolutions:

1. Add [R-Confidentiality]
2. Do not add [R-Confidentiality]

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-12-02:AssertionConfidentiality]

1. Add the requirement: [R-AssertionConfidentiality] SAML should define a format so that individual SAML assertions may be encrypted, independent of protocol bindings.
2. Add the requirement: [R-AssertionConfidentiality] SAML assertions must be encrypted, independent of protocol bindings.
3. Add a non-goal: SAML will not define a format for protecting confidentiality of individual assertions; confidentiality protection will be left to the protocol bindings.
4. Do not add either requirement or the non-goal.

Status: Closed per F2F #2, No Conclusion

CLOSED ISSUE:[UC-12-03:BindingConfidentiality]

The first option is intended to make the protection optional (both in the binding definition, and by the user at runtime).

1. [R-BindingConfidentiality] Bindings SHOULD (in the RFC sense) provide a means to protect SAML data from observation by third parties. Each protocol binding must include a description of how applications can make use of this protection. Examples: S/MIME for MIME, HTTP/S for HTTP.
2. [R-BindingConfidentiality] Each protocol binding must always protect SAML data from observation by third parties.

3. Do not add either requirement.

Status: Closed per F2F #2, Resolution 1 Carries

CLOSED ISSUE:[UC-12-04:EncryptionMethod]

If confidentiality protection is included in the SAML assertion format (that is, you chose option 1 or 2 for [UC-12-02:AssertionConfidentiality]), how should the protection be provided?

Note that if option 2 (assertion confidentiality is required) was chosen for UC-12-02, resolution 1 of this issue implies that SAML will not be published until after XML Encryption is published.

Proposed resolutions; choose one of:

1. Add the requirement: [R-EncryptionMethod] SAML should use XML Encryption.
2. Add the requirement: [R-EncryptionMethod] Because there is no currently published standard for encrypting XML, SAML should define its own encryption format. Edit the existing non-goal of not creating new cryptographic techniques to allow this.
3. Add no requirement now, but include a note that this issue must be revisited in a future version of the SAML spec after XML Encryption is published.
4. Do not add any of these requirements or notes.

Status: Closed per F2F #2, Resolution 3 Carries

Group 13: Business Requirements

CLOSED ISSUE:[UC-13-01:Scalability]

Bob Morgan brought up several "business requirements" on security-use. One was scalability. This issue is a placeholder for further elaboration on the subject.

A candidate requirement might be:

> [CR-13-01-Scalability] SAML should be appropriate for high volume of messages, and for messages between parties made up of several physical machines.

Potential Resolutions:

1. Add requirement [CR-13-01-Scalability].
2. Do not add this requirement.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-13-02:EfficientMessages]

Philip Hallam-Baker's core assertions requirement document included several requirements that were efficiency-oriented. When that requirement document was merged into Straw Man 2, the efficiency requirements were excluded.

One such requirement was:

> [CR-13-02-EfficientMessages] SAML should support efficient message exchange.

Potential Resolutions:

1. Add this requirement to the use case and requirements document.
2. Leave this requirement out of use case and requirements document.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-13-03:OptionalAuthentication]

Philip Hallam-Baker's core assertions requirement document included several requirements that were efficiency-oriented. When that requirement document was merged into Straw Man 2, the efficiency requirements were excluded.

One such requirement was:

> [CR-13-03-OptionalAuthentication] Authentication between asserting party and relying

> party should be optional. Messages may omit authentication altogether.

In this case, "authentication" means authentication between the parties in the conversation (for example, by means of a digital signature) and not authentication by the subject.

Potential Resolutions:

1. Add this requirement to the use case and requirements document.
2. Leave this requirement out of use case and requirements document.

Status: Closed per F2F #2, 2 carries

CLOSED ISSUE:[UC-13-04:OptionalSignatures]

Philip Hallam-Baker's core assertions requirement document included several requirements that were efficiency-oriented. When that requirement document was merged into Straw Man 2, the efficiency requirements were excluded.

One such requirement was:

> [CR-13-04-OptionalSignatures] Signatures should be optional.

Potential Resolutions:

1. Add this requirement to the use case and requirements document.
2. Leave this requirement out of use case and requirements document.

Status: Closed, Voted on May 15 telcon for resolution 1

CLOSED ISSUE:[UC-13-05:SecurityPolicy]

Bob Morgan proposed a business-level requirement as follows:

> [CR-13-05-SecurityPolicy] Security measures in SAML should support common institutional security policies regarding assurance of identity, confidentiality, and integrity.

Potential Resolutions:

1. Add this requirement to the use case and requirements document.
2. Leave this requirement out of use case and requirements document.

Status: Closed per F2F #2, Resolution 2 Carries

CLOSED ISSUE:[UC-13-06:ReferenceReqt]

Bob Morgan has questioned requirement [R-Reference] in that it is not specific enough. In particular, he said: "Goal [R-Reference] either needs more elaboration or (likely) needs to be dropped. What is a 'reference'? It doesn't have a standard well-understood security meaning nor is it defined in the glossary. This Goal seems to me to be making an assumption about a low-level mechanism for optimizing some of the transfers."

One possible, more specific elaboration might be:

> [CR-13-06-1-Reference] SAML should define a data format for providing references to authentication and authorization assertions. Here, a "reference" means a token that may not be a full assertion, but can be presented to an asserting party to request a particular assertion.
>
> [CR-13-06-2-Reference-Message] SAML should define a message format for requesting authentication and authorization assertions using references.
>
> [CR-13-06-2-Reference-Size] SAML references should be small. In particular, they should be small enough to be transferred by Web browsers, either as cookies or as CGI parameters.

Potential Resolutions:

1. Replace [R-Reference] with these requirements.
2. Leave [R-Reference] as it is.
3. Remove mention of references entirely.

Status: Closed per F2F #2, Resolution 2 Carries

ISSUE [UC-13-07: Hailstorm Interoperability]

Should SAML provide interoperability with the Microsoft Hailstorm architecture, including the Passport login system?

Status: Open

Group 14: Domain Model

ISSUE:[UC-14-01:UMLCardinalities]

The cardinalities in the UML diagrams in the Domain Model are backwards.

Frank Seliger comments: The Domain model claims to use the UML notation, but has the multiplicities according to the Coad method. If it were UML, the diagram would state that one Credential could belong to many Principals. I assume that we would rather want to state that one Principal can have many Credentials, similarly for System Entity, the generalization of User. One Principal would belong to several System Entities or Users according to the diagram. I would rather think we want one System Entity or User to have several Principals.

My theory how these wrong multiplicities happened is the following: As I can see from the change history, the tool Together has been used to create the initial version of this diagram. Together in its first version used only the Peter Coad notation. Later versions still offered the Coad notation as default. Peter Coad had the cardinalities (UML calls this multiplicities) just swapped compared to the rest of the world. This always caused grief, and it did again here.

Dave Orchard agrees this should be fixed.

Status: Open

Design Issues

Group 1: Naming Subjects

CLOSED ISSUE:[DS-1-01: Referring to Subject]

By what means should Assertions identify the subject they refer to?

Bob Blakely points out that references can be:

1. Nominative (by name, i.e. some identifier)
2. Descriptive (by attributes)
3. Indexical (by “pointing”)

SAML may need to use all types, but Indexical ones in particular can be dangerous from a security perspective.

Status: Closed by vote on Sept 4, superceded by more specific issues.

ISSUE:[DS-1-02: Anonymity Technique]

How should the requirement of Anonymity of SAML assertions be met?

Potential Resolutions:

1. Generate a new, random identified to refer to an individual for the lifetime of a session.

2. ???

Status: Open

ISSUE:[DS-1-03: SubjectComposition]

What is the composition of a subject or "subject specifier" within:

- An AuthnAssn?

- An AuthnAssnReq?

Note that we have consensus on the overall composition as noted in [sec. 2, 3, & 4 of WhiteboardTranscription-01.pdf].

This was identified as F2F#3-9.

This is a more specific variant of DS-1-01.

Status: Open

ISSUE:[DS-1-04: AssnSpecifiesSubject]

Should it be possible to specify a subject in an Assertion or Assertion Request by reference to another Assertion containing the subject in question? The referenced Assertion might be indicated by its AssertionID or including it in its entirety.

For example, a PDP might request an Attribute Assertion from an Attribute Authority by providing an Authentication Assertion (or its ID) as the way of identifying the subject.

There are two cases: AssertionID and complete Assertion.

AssertionID

When requesting an Assertion, it will be useful to specify an AssertionID in a situation where the requestor does not have a copy of the Assertion, but was had received the AssertionID from some source, for example in a Web cookie. Of course, it would be necessary that the Asserting Party be able to obtain the Assertion in question. This scenario would be particularly convenient if the Asserting Party already possessed the referenced Assertion, either because it had used it previously for some other purpose or because it was co-located with the Authority that created it originally.

Using an AssertionID to specify the subject of an Assertion seems less useful, because it would make it impossible to interpret the Assertion by itself. If at some later time, the referenced Assertion was no longer available; it would not be possible to determine the subject of the Assertion in question. Even it the Assertion was available, having two assertions rather than one would be much less convenient.

Complete Assertion

Whether requesting an Assertion or creating a new assertion, it would never be strictly necessary to include another Assertion in its entirety to specify the subject of the first Assertion, because the subject field could be copied instead. Hypothetically, the complete contents of the Assertion might have some value, as the basis of a policy decision, however the same need could be served as well by attaching the second Assertion, rather than including it within the subject field of the first.

This was identified as F2F#3-19 and F2F#3-27, although the scope of the latter is limited to the specific case of an Authentication Assertion being referenced within an Attribute Assertion.

Potential Resolutions:

1. Allow a subject to be specified by an AssertionID or complete Assertion.
2. Allow a subject to be specified by an AssertionID, but not a complete Assertion.
3. Allow a subject to be specified only in an Assertion Request by an AssertionID.

4. Do not allow a subject to be specified by either an AssertionID or complete Assertion.

Status: Open

CLOSED ISSUE:[DS-1-05: SubjectofAttrAssn]

This statement's exact meaning needs to be clarified: "the only Subjects of Attribute Assertions are Subjects as described by Authentication Assertions."

This was identified as F2F#3-26.

Status: Closed by vote on Sept, 4. The statement "the only Subjects of Attribute Assertions are Subjects as described by Authentication Assertions" has not been clarified, however the Subject element of both types of Assertion have identical schemas and there is no suggestion in the core spec that they differ in any way.

ISSUE:[DS-1-06: MultipleSubjects]

Can an Assertion contain multiple subjects? The multiple subjects might represent different identities, which all refer to the same system entity. Allowing multiple subjects seems more general and allows for unanticipated future uses.

On the other hand, having multiple subjects creates a number of messy issues, particularly if they don't refer to the same entity.

Champion: Irving Reid

Status: Open

ISSUE:[DS-1-07: MultpleSubjectConfirmations]

Should multiple Confirmation methods be allowed for a single NameIdentifier within the Subject? Basically, this a tradeoff between flexibility and complexity of (possibly undefined) semantics.

Champion: Gil Pilz

Status: Open

ISSUE:[DS-1-08: HolderofKey]

If a HolderOfKey SubjectConfirmation is used, does that imply that the subject is the sender of the associated application message (request)? In general, the semantics of SubjectConfirmation need to be made very explicit in the core specification.

Champion: Irving Reid

Status: Open

ISSUE:[DS-1-09: SenderVouches]

What are the semantics of SenderVouches? How does an Assertion containing this element differ from one that does not? When should it be used?

Champion: Prateek Mishra

Status: Open

Group 2: Naming Objects

CLOSED ISSUE:[DS-2-01: Wildcard Resources]

Nigel Edwards has proposed that Authorization Decision Assertions be allowed to refer to multiple resources by means of some kind of wildcards.

Potential Resolutions:

1. Allow resources to be specified with fully general regular expressions.
2. Allow resources to be specified with simple * wildcard in the final path element: e.g. /foo/*, but not /foo/*/x or /foo/y*
3. Don’t allow wildcarded resources

Status: Closed by vote during May 29 telecon

CLOSED ISSUE:[DS-2-02: Permissions]

Should the qualifiers of objects be called permissions, actions or operations? Authorization decision assertions contain an object that identifies the target of the request. This is qualified with a field called permissions, containing values like “Read” and “Write”. Normal English language usage suggests that this field represents an Action or Operation on the object.

Possible Resolutions:

1. Retain Permissions
2. Change to Actions
3. Change to Operations

Status: Closed by vote on Sept 4. Resolution 2 (Actions)

Group 3: Assertion Validity

ISSUE:[DS-3-01: DoNotCache]

It has been suggested that there should be a way in SAML to specify that an assertion is currently valid, but should not be cached for later use. This should not depend on the particular amount of variation between clocks in the network.

For example, a PDP may wish to indicate to a PEP that it should make a new request for every authorization decision. For example, its policy may be subject to change at frequent and unpredictable intervals. It would be desirable to have a SAML specified convention for doing this. This may interact with the position taken on clock skew. For example, if SAML takes no position on clock skew the PDP may have to set the NotAfter value to some time in the future to insure that it is not considered expired by the PEP.

Potential Resolutions:

1. SAML will specify some combination of settings of the IssueInstant and ValidityInterval to mean that the assertion should not be cached. For example, setting all three datetime fields to the same value could be deemed indicate this.

2. SAML will add an additional element to either Assertions or Responses to indicate the assertion should not be cached.

3. SAML will provide no way to indicate that an Assertion should not be cached.

Status: Open

ISSUE:[DS-3-02: ClockSkew]

SAML should consider the potential effects of clock skew in environments it is used.

It is impossible for local system clocks in a distributed system to be exactly the same, the only question is: how much do they differ by? This becomes an issue in security systems when information is marked with a validity period. Different systems will interpret the validity period according to their local time. This implies:

1. Relying parties may not make the same interpretation as asserting parties.

2. Distinct relying parties may make different interpretations.

Generally what matters is not the absolute difference, but the difference as compared to the total validity interval of the information. For example, the PKI world has tended to (rightly) ignore this issue because CA and EE certificates tend to have validity intervals of years. Even Attribute Certificates and SAML Attribute Assertions are likely to have validity intervals of days or hours. However, it seems likely that Authorization Decision Assertions may sometimes have validity intervals of minutes or seconds. Therefore, the issue must be raised.

One common problem is what to set the NotBefore element to. If it is set to the AP's current time, it may not yet be valid for the RP. If set in the past, (a common practice) the questions arise 1) how far in the past? and 2) should the NotAfter time also be adjusted? If NotBefore is omitted, this may not be satisfactory for nonrepudiation purposes.

The NotAfter value can also be an issue if the assumed clock skew is large compared to the Validity Interval.

[These paragraphs contain personal observations by Hal Lockhart, others may disagree.

In the early 1990's some popular computer systems had highly erratic system clocks which could drift from the correct time by as much as five minutes per day. Kerberos's requirement for rough time synchronization (usually 5 minutes) was criticized at that time because of this reality.

Today most popular computer systems have clocks which keep time accurately to seconds per month. Therefore the most common current source of time differences is the manual process of setting time. Therefore, most systems tend to be accurate within a few minutes, generally less than 10.

By means of NTP or other time synchronization system, it is not hard to keep systems synchronized to less than a minute, typically within 10 seconds. It is common for production server systems to be maintained this way. The price of GPS hardware has fallen to the point where it is not unreasonably expensive to keep systems synchronized to the true time with sub-second accuracy. However, few organizations bother to do this.]

Potential Resolutions:

1. SAML will leave it up to every deployment how to deal with clock skew.

2. SAML will explicitly state that deployments must insure that clocks differ by no more that X amount of time (X to be specified in the specification)

3. SAML will provide a parameter to be set during deployment that defines the maximum clock skew in that environment. This will be used by AP's to adjust datetime fields according to some algorithm.

4. SAML will provide a parameter in assertions that indicates the maximum skew in the environment. RPs should use this value in interpreting all datetime fields.

Status: Open

ISSUE:[DS-3-03: ValidityDependsUpon]

In a previous version of the draft spec, assertions contained a `ValidityDependsUpon` element, which allowed the asserting party to indicate that this assertion was valid only if another, specified assertion was valid. This was dropped because it was felt that the lack of a SAML mechanism to revoke previously issued assertions made it moot.

A number of people feel that this element is useful nevertheless and should be restored.

It is worth noting that even in the absence of this element (from the a particular assertion or SAML as a whole) a particular relying party can still have a policy that requires multiple assertions to be valid.

Status: Open

Group 4: Assertion Style

CLOSED ISSUE:[DS-4-01: Top or Bottom Typing]

Should assertions be identified as Authentication, Attribute and Authorization Decision, each containing specified elements? (Top Typing) Or should only the elements be defined allowing them to be freely mixed? (Bottom Typing)

Two comprehensive proposals to address this issue have been made in draft-orchard-maler-assertion-00 and draft-sstc-core-08.

Status: Closed by vote on Sept 4. Made moot by current schemas, which draw on both sets of ideas.

ISSUE:[DS-4-02: XML Terminology]

Which XML terms should we be using in SAML? Possibilities include: message, document, package.

Status: Open

CLOSED ISSUE:[DS-4-03: Assertion Request Template]

What is the best way to provide a template of values in an assertion request?

Two comprehensive proposals to address this issue have been made in draft-orchard-maler-assertion-00 and draft-sstc-core-08.

Potential Resolutions:

1. The requestor sends an assertion with the required field types, but missing values

2. The requestor sends fields and values, in the form of a list, not an assertion

3. XPATH expressions

4. XML query statements

Status: Closed by vote on Sept 4. Agreed upon approach does not use a template.

ISSUE:[DS-4-04: URIs for Assertion IDs]

Should URIs be used as identifiers in assertions?

This issue was identified as F2F#3-8: “We need to decide the syntax of AssertionID.” Although this is a broader formulation, the discussion below is actually directed towards it rather than the original form (above).

This was identified as CONS-02. Does the specification (core-12) need additional specification for the types of assertion, request, and response IDs? If so, what are these requirements?

Background...

From the focus group minutes [1]:

> >- URIsForAssertionIDs: What are the pros and cons? What other

> > methods are there?

>

> DS-4-04: URIs for Assertion IDs: (still open after today)

>

> Eve, with help from Dave, gave a short tutorial on the problems with

> URI identity in XML namespace names.

There followed a brief discussion in which we touched upon various aspects of this problem space. We terminated the discussion upon issuing the above "new action". (the discussion as-documented in the aforementioned minutes is attached below for reference [1])

Further background, in the form of the specs for AssertionID and Issuer from draft-sstc-core-07 are excerpted at [2].

Relevant, recent discussion on security-services@lists.oasis-open.org...

Hal said in

http://lists.oasis-open.org/archives/security-services/200105/msg00146.html

> 5. In 1.3.1 I don't understand the intended purpose of AssertionID.

PHB replied in

http://lists.oasis-open.org/archives/security-services/200105/msg00159.html

> The AssertionID provides a unique reference for the assertion. ...

> Within SAML 1.0 the principle use of an AssertionID would be to allow

> one assertion to reference another (see previous Tim discussion) thus

> allowing statements of the form `this assertion was constructed from

> that assertion'.

> The principle use of the AssertionID however would be in systems built
> around SAML, they provide the basis for audit and accountability for
> example. If a system is built that allows for second order logic
> (assertions may be true or false and other assertions may make
> statements about validity (c.f. TASS meta-assertions)), then an
> assertionID is essential.

Analysis...

The stated purpose of the AssertionID element is as an "assertion unique identifier" [2]. The stated syntax of this identifier is a URI [3]. Implicit in this line of thinking is a notion that URIs may be created (aka "minted") in a globally decentralized, non-colliding fashion due to the properties of the URI "space" [4].

The following is stated in [2] about AssertionID..

> The URI is used as a name for the assertion and not as a locator. It
> is only necessary to ensure that no two assertions share the same
> identifier. Provision of a service to resolve an identifier into an
> assertion is not a requirement.

Also, as far as I can tell, [2] postulates (in section 1.3) that a requester need supply only an assertionID in a SAMLQuery in order to obtain an assertion. It does not make clear any distinction between newly minting an assertion and retrieving an already-existing one.

Thus it seems that there is a tacit assumption in [2] that an assertion may be uniquely identified and minted/retrieved using only an assertionID, regardless of the quote above.

So it seems that an assertionID is being asked to both..

A. identify, globally and uniquely, assertions;

B. provide at least a hint about where to direct requests for minting
   or retrieving assertions.

..but again, this is to a fair degree inferred from a rough, incomplete, draft spec ([2]).

Additionally, there are many subtleties to using URIs as identifiers rather than straight-ahead resoure locators. See the minutes of the "Future of URIs" Birds of the Feather session held at the 50th IETF meeting [11],

Thoughts...

It is an arguably good design principle to separate functions between various data items such that their roles in life are unambiguous.

[2] already has an "Issuer" assertion element. If identifying assertions is predicated on using the tuple "assertionID, Issuer", and some method for guaranteeing non-colliding Issuer names is used (e.g. DNS domain names, and things built upon them), then the assertionID can be quite simple, e.g. an integer (as is done in PKIX [10]).

In using the "assertionID, Issuer" tuple to identify assertions, and also provide guidance about where to go to make requests about or for them, the role of the Issuer element may arguably be (too) overloaded. E.g. if the overall SAML design calls for assertions to (perhaps optionally) specify within their structure where a receiver of an assertion may go to make queries about the assertion, then the requirements for persistence and location-independence for that particular identifier may conflict with the requirements of simply globally and uniquely (and perhaps persistently) identifying the Issuer security domain.

So it may be the case that to..

case 1) globally uniquely identify an assertion one needs the combination of "assertionID, Issuer",

case 2) uniquely identify assertions in the context of a given security domain, one needs only "assertionID" (it doesn't need to be disambiguated from assertions from other security domains; in this case the assertionID starts to look a lot like a serial number),

case 3) one needs to cover either of the prior cases, and also needs to specify where to go (and "how" to "go") to make requests to the security domain in question. I.e...

```
 <assertionID>123123123123</assertionID>

 <Issuer>some-issuer-identifier</Issuer>  -- perhaps optional

 <Source>saml://example.org/send-yer-SAML-based-requests-here   -- optional

 </Source>
```

Tho there are good arguments for not making Issuer optional (case 2), thus the overall set of identifying information might be structured something like this..

```
 <assertionID>

   <serialNumber>123123123123</serialNumber>
```

```
    <Issuer>some-issuer-identifier</Issuer>

  </assertionID>

  <Source>saml://example.org/send-yer-SAML-based-requests-here   -- optional

  </Source>
```

Further thoughts...

There's tons of subtle-but-important details in all of this that need to be considered in nailing down a design. Some of them are..

D1. if one uses a URL or URL-like flavor of URI as an identifier, we need to specify how comparisons between said identifier and other blobs of data are made. [3] details some of these subtleties in sections 1.5 and 2.1. The lowest-common-denominator option of specifying that such comparisons are made by performing a byte-by-byte octet string comparison will only technically work if certain restrictions are specified for the URI-based values. The SAML specs may need to consider/specify/incorporate one or more or all of..

* charset restrictions for all or some SAML elements,
* charset specifications, and bounds on said specifications, for SAML elements whose value syntaxes are URI [3],
* charset(s) specified/allowed by underlying protocols and interaction thereof with the prior items in this list,
* [perhaps others/more]

Of note is "Character Model for the World Wide Web 1.0" [14] which defines an algorithm called "String Identity matching" (in section 6), which has implications for the above. (it also has implications for SAML in general, see D6).

D1.1. See also [16] [17] for further musing about internationalization for URI and other identifiers.

D1.2. See also "Considerations for URI and FQDN Protocol Parameters" [18] for further musings about using DNS domain names and/or URI as identifiers in protocol elements.

D1.3. If URI are used as identifiers in protocol elements, software modules that handle them (this includes people as a boundary condition ;) may wonder just what the heck their semantics are, because their semantics can be so varied. "URI Relationship Discovery via RESCAP" [19] touches upon and enumerates these questions, as well as sketch a protocol-based approach that specifies a service providing such info. Additionally, the more recent I-D, "URI Resolution using the Dynamic Delegation Discovery System" [20], also provides some relevant background info.

D1.4. Registration issues -- URI (nee URL) schemes should be registered, same with URN namespaces. See [9] for pointers to relevant RFCs on how to accomplish such registrations.

D2. some-issuer-identifier -- should this simply be a DNS fully-qualified-domain-name? Should it be a URN [6]? Should it be something else?

D3. use of URNs -- URNs have semantics of persistence and location-independence. Their use may or may not be appropriate in the context of SAML assertions depending upon the semantics of the thing they're being called upon to identify [6] [7]. E.g. it is questionable to use a URN to identity a given non-persistent, indeed likely ephemeral, artifact such as an instantiation of a SAML assertion. However, it is

D4. if URNs are used, what namespace identifiers are appropriate? Any? Only a selected one(s)? Formal or informal? [7] [12]

D5. the DOI work [13] is likely not appropriate for SAML's purposes due to that effort's Intellectual Property emphasis and also because of the implied (required?) dependency upon the Handle System. The latter is an nascent, intended-to-be-scalable-to-the-Internet, naming and name resolution system [13] (I haven't yet read the internet-drafts in detail).

D6. The emergent "Character Model for the World Wide Web 1.0" MAY have various implications for SAML's specification, beyond that noted in D1.

D7. IMHO, "tag:" URIs [15] are not appropriate for our problem space, given their present specification, but reading about them and the discussion thereof on the uri@w3.org list is educational.

D9. If an artifact is not persistent, then it's identifier may be reused under certain conditions. Something to keep in mind and think about.

Notes and References...

[1] URIsForAssertionIDs discussion, from Focus subgroup concall, 22-May-2001:

http://lists.oasis-open.org/archives/security-services/200105/msg00139.html

>- URIsForAssertionIDs: What are the pros and cons? What other methods

>   are there?

DS-4-04: URIs for Assertion IDs: (still open after today)

Eve, with help from Dave, gave a short tutorial on the problems with URI identity in XML namespace names.

Thomas: The DOI people are working on this general problem. (http://www.doi.org, http://www.handle.net/)

Eve: It would be acceptable to use URIs if we apply constraints. E.g., they should be absolute (or even should be absolute URNs) and we should define what equality means. Dave: Solving the "whole URI problem" is way bigger than SAML's scope.

Jeff: There was recently an IETF BOF on the future of URIs, and W3C was investigating these issues, but nothing has really happened.

Eve: See W3C's Character Model spec for recommendations on normalization and internationalized URIs. (http://www.w3.org/TR/charmod/)

Dave: Cautioned that we have to be concerned with real-world websites and their behavior, which is not precisely the same as the standards. For example, http://www.jamcracker.com and http://www.jamcracker.com/index.html point to the same resource, but how can people know that?

BobB: Aliases, symbolic links, etc. are a problem if you have policies on different aliases that conflict.

Hal: We can take a hard line on URIs for assertion IDs, but for resources, we may have to deal with the vagaries of real-world URIs.

Evan: URIs are opaque strings, and XML makes data's structure more transparent.

Hal: There will probably be more cases than just AssertionID where identifiers will have properties of uniqueness (RequestID?) and are just "internal to SAML." We should pull out the description of these properties into a separate section and have it referred to from the various sections.

Hal: We should register a new URI scheme, e.g. "saml:" Thomas: We could

just use URNs and have the same effect. Jeff: It's pretty easy to register

a new scheme with IANA. (http://www.ietf.org/rfc/rfc2717.txt)

Eve: It's surprisingly hard to register a new URN namespace (http://www.ietf.org/rfc/rfc2611.txt)

NEW ACTION: Jeff to send out email about possible URI constraints and identity definitions we should consider imposing in the case of SAML's unique identifiers.

[2] from draft-sstc-core-07: http://www.oasis-open.org/committees/security/docs/draft-sstc-core-07.pdf

> 1.4.2 Element <AssertionID>
>
> Each assertion MUST specify exactly one unique assertion identifier.

> All identifiers are encoded as a Uniform Resource Identifier (URI)
> and are specified in full (use of relative identifiers is not
> permitted).
>
> The URI is used as a name for the assertion and not as a locator. It
> is only necessary to ensure that no two assertions share the same
> identifier. Provision of a service to resolve an identifier into an
> assertion is not a requirement.
> The following schema defines the <AssertionID> element:
> <element name="AssertionID" type="string"/>
> 1.4.3 Element <Issuer>
> The Issuer element specifies the issuer of the assertion by means of a
> URI. It is defined by the following XML schema:
> The following schema defines the <Issuer> element:
> <element name="Issuer" type="string"/>

[3] Uniform Resource Identifiers (URI): Generic Syntax http://www.ietf.org/rfc/rfc2396.txt

[4] URIs encompass both URLs and URNs. The former [5] often (but not always) depend upon the Domain Name System (DNS) namespace, which enables the capability to mint globally unique URLs in a decentalized fashion. The latter [6] define a hierarchical namespace that is DNS-independent but centrally mediated [7] in order to provide "location independent identification of a resource, as well as longevity of reference".

```
This picture is from [8]...
         ___________________________________________________________
        |                                                           |
        |         __________________                                |
        |        |  ftp:             |                              |
        |        |  gopher:          |                              |
        |        |  http:          __|_____________                 |
        |        |  etc           |  |  urn:       |                |
        |        |________________|__|             |                |
        |               URLs      |                |                |
        |                         |________________|                |
        |                              URNs                         |
        |___________________________________________________________|
```

URIs

URIs, URLs, and URNs are described by a plethora of documents. An attempt to tie them all together is given in [9].

[5] Uniform Resource Locators (URL) http://www.ietf.org/rfc/rfc1738.txt

[6] URN Syntax http://www.ietf.org/rfc/rfc2141.txt

[7] URN Namespace Definition Mechanisms http://www.ietf.org/rfc/rfc2611.txt

[8] Naming and Addressing: URIs, URLs, ...http://www.w3.org/Addressing/

[9] Uniform Resource Identifiers: Comprehensive Standard http://www.ietf.org/internet-drafts/draft-daigle-uri-std-01.txt

[10] PKIX Certificate and CRL Profile http://www.ietf.org/rfc/rfc2459.txt

[11] Future of Uniform Resource Identifiers BOF (furi) [50th IETF, Minneapolis MN, Mar-2001] http://www.ietf.org/proceedings/01mar/ietf50-39.htm#TopOfPage

[12] URI.NET -- a clearing house for information on URIs in general and on specific URI schemes and software http://www.uri.net/

[13] Digital Object Identifiers, The Handle System http://www.doi.org, http://www.handle.net/

[14] Character Model for the World Wide Web 1.0 http://www.w3.org/TR/charmod/

[15] "Tag" URI Scheme http://www.taguri.org/ see also the thread on uri list "Proposal: 'tag' URIs", from Tim Kindberg <timothy@hpl.hp.com>...http://lists.w3.org/Archives/Public/uri/2001Apr/0013.html

http://www.taguri.org/2001-04-26/draft-kindberg-tag-uri-00.txt

[16] Internationalization: URIs and other identifiers http://www.w3.org/International/O-URL-and-ident.html

[17] Internationalized Resource Identifiers (IRI) http://www.ietf.org/internet-drafts/draft-masinter-url-i18n-07.txt

[18] Considerations for URI and FQDN Protocol Parameters http://www.ietf.org/internet-drafts/draft-eastlake-uri-fqdn-param-00.txt

[19] URI Relationship Discovery via RESCAP http://www.ietf.org/internet-drafts/draft-mealling-uri-rdf-00.txt

[20] URI Resolution using the Dynamic Delegation Discovery System http://www.ietf.org/internet-drafts/draft-ietf-urn-uri-res-ddds-03.txt

Status: Open

ISSUE:[DS-4-05: SingleSchema]

Should we design the schema for Assertions and their respective request/response messages in different XML namespaces?

Request/response messages could reference the core assertions schema. There could be many applications that reference the core assertions without referencing the request/response stuff. Making them pull in the request/response namespace is just extra overhead.

This has been identified as F2F#3-36.

Potential Resolutions:

1. Use a single schema for Assertions and Request/Response messages.
2. Have a schema for Assertions that is distinct from the schema for Request/Response messages.

Status: Open

CLOSED ISSUE:[DS-4-06: Final Types]

Does the TC plan to restrict certain types in the SAML schema to be final? If so, which types are to be so restricted?

This was identified as CONS-03.

Status: Closed by vote on Sept 4. The Schema recommendations proposed by Eve and Phill at F2F#4 have been accepted.

CLOSED ISSUE:[DS-4-07: ExtensionSchema]

One of the goals of the F2F #3 “whiteboard draft” was to use strong typing to differentiate between the three assertion types and between the three different query forms. This has been achieved (in core-12) through the use of ``abstract’’ schema and schema inheritance. One implication is that any concrete assertion instance MUST utilize the xsi:type attribute to specifically describe its type even as all assertions will continue to use a single <Assertion> element as their container. XML processors can key off this attribute during assertion processing.

Is this an acceptable approach? Other approaches, such as the use of substitution groups, are also available. Using substitution groups, each concrete assertion type would receive its own distinguished top-level element (e.g., <AuthenticationAssertion>) and there would be no need for the use of xsi:type attribute in any assertion instance. At the same time the SAML schema would be made somewhat more complex through the use of substitution groups.

Should the TC investigate these other approaches? Most important: what is the problem with the current approach?

This was identified as CONS-04.

Status: Closed by vote on Sept 4. The Schema recommendations proposed by Eve and Phill at F2F#4 have been accepted

ISSUE:[DS-4-08: anyAtttribute]

Summary: In order to make it possible to extend SAML to add attributes to native elements, we would need to add <xsd:anyAttribute> all over the place. Should we do this?

Explanation:

We have expended a lot of effort trying to get SAML's customizability "right". We allow the extension of our native types to get new elements, and in selected places we allow for the addition of foreign elements by design. Given our prohibition against changing SAML semantics with foreign markup, we wouldn't have to worry if foreign attributes were tacked onto native elements, and this is a relatively cheap and easy way to "extend" a vocabulary.

For example, if a SAML assertion producer finds it convenient to add ID attributes to various elements for internal management purposes, or if they want to state what natural language an attribute value is in, currently they can't do that and still validate the results:

```
<saml:AttributeValue xml:lang="EN-US" AttValID="12345">...
```

Now, xml:lang is somewhat of a special case, since its semantics are baked into core XML, but you still need to account for it in the schema if you want to validate. We may want to account for xml:lang and xml:space specially in the schema just because XML always allows them, but that doesn't answer the ID attribute case, or any other similar case.

The anyAttribute approach is used in some other schemas I know of, but in general they also use ##any and ##other a lot more too.

Do we want to allow this kind of flexibility in SAML?

Champion: Eve Maler

Status: Open

ISSUE:[DS-4-09: Eliminate SingleAssertion]

Proposal:

- Eliminate the <SingleAssertion> Element and SingleAssertionType.

- Rename the <Assertion> element to <AbstractAssertion>.
- Rename <MultipleAssertion> to <Assertion> and MultipleAssertionType to AssertionType.

Rationale:

In the current core the <Assertion> element is of type AssertionAbstractType and contains assertion header data and no statements. <SingleAssertion> is of type SingleAssertionType and contains assertion header data and exactly one statement. <MultipleAssertion> is of type MultipleAssertionType and contains assertion header data and ZERO or more statements.

There are a number of problems with this.

First of all it is entirely possible to construct a SAML assertion containing one statement in two valid ways: as either a <SingleAssertion>, or as a <MultipleAssertion> that contains exactly one element. In general we want to avoid creating languages that allow you to say the same thing different ways--primarily to avoid the possibility of implementers drawing a distinction between the two cases.

I would suggest doing away with the <SingleAssertion> element and type altogether, since it's functionality is entirely incorporated into the <MultipleAssertion> element and type.

Theoretically we lose the benefit of being able to make slightly more efficient systems for cases where it is KNOWN that only single statements will be contained in the assertions passed. I would assert that this benefit is illusory, but that even if it were real in some cases it's loss is certainly outweighed by the fact that general SAML systems would not have to handle both <SingleAssertion> and <MultipleAssertion> elements--without even considering the general gain of avoiding the "two ways to say one thing" problem.

Secondly there is the problem of the <Assertion> element. I assume that it is declared to allow people to specify that other elements will contain an "assertion", and that the intention is that in practice this will be populated with an descendant type that is identified via the xsi:type notation. In other words, I think the intention is that no one will even create an <Assertion> element that actually has the "AssertionAbstractType" type--they will only ever use it as a placeholder to indicate that a descendant of the "AssertionAbstractType" should be inserted. If this is the case then I suggest that we make this explicit by renaming the <Assertion> element to <AbstractAssertion>.

Thirdly, we can now rename <MultipleAssertion> to <Assertion> and "MultipleAssertionType" to "AssertionType".

The result:

A core where the <AbstractAssertion> element is of type "AssertionAbstractType", and contains only assertion header data, and the <Assertion> element--which is of "AssertionType" contains assertion header data and zero or more statements.

Champion: Chis McLaren

Status: Open

Group 5: Reference Other Assertions

A number of requirements have been identified to reference an assertion with in another assertion or within a request.

Phillip Hallam-Baker observes: “there is more than one way to support this requirement,

“[A] The first is to simply cut and paste the assertion into the <Subject> field so we have <Subject><Assertion><Claims><Subject>[XYZ]. This approach is simple and direct but does not seem to achieve much since it essentially comes down to ‘you can unwrap this structure to find the information you want’. Why not just cut to the chase and specify <Subject>[XYZ] ?

“[B] The problem with cutting to the chase is that it means that the application is simply told the <subject> without any information to specify where that data came from. In many audit situations one would need this type of information so that if something bad happens it is possible to work out exactly where the bogus information was first introduced and how many inferences were derived from it. So we might have <Subject><AssertionRef>[XYZ]

“[C] The above is my preferred representation since the assertion can be used immediately by the simplest SAML application without the need to dereferrence the assertion reference to discover the subject of the assertion. However one could argue that an application might want to specify simply <Subject><AssertionRef> and then specify the referenced assertion in the advice container.

“I think that the choice is really between [B] and [C] since the first suggestion in [A] is unwieldy and the second is simply the status quo.

“Of these [B] is more verbose, [C] requires applications to perform some pointer chasing and could be seen as onerous.”

The following four scenarios have been identified where this is required:

ISSUE:[DS-5-01: Dependency Audit]

One issue with draft-sstc-core-07.doc is a lack of support for audit of assertion dependency between co-operating authorities. As one explicit goal of SAML was to support inter-domain security (i.e., each authority may be administered by a separate business entity) this seems to be a serious "gap" in reaching that goal.

Consider the following example:

(1) User Ravi authenticates in his native security domain and receives

Assertion A:

```
        <Assertion>
     <AssertionID>http://www.small-company.com/A</AssertionID>
     <Issuer>URN:small-company:DivisionB</Issuer>
     <ValidityInterval> . . . </ValidityInterval>
     <Claims>
       <subject>"cn=ravi, ou=finance, id=325619"</subject>
       <attribute>manager</attribute>
     </Claims>
   </Assertion>
```

(2) User Ravi authenticates to the Widget Marketplace using assertion A and based on the policy:

All entities with "ou=finance" authenticated thru small-company.com with attribute manager have purchase limit $100,000 receives Assertion B from the Widget Marketplace:

```
        <Assertion>
    <AssertionID>http://www.WidgetMarket.com/B<AssertionID>
    <Issuer>URN:WidgetMarket:PartsExchange</Issuer>
    <ValidityInterval>. . . </ValidityInterval>
    <Claims>
      <subject>"cn=ravi, ou=finance, id=325619"</subject>
      <attribute>max-purchase-limit-$100,000</attribute>
    </Claims>
  <Assertion>
```

(3) User Ravi purchases farm machinery from a parts provider hosted at the Widget Marketplace. The parts provider authorizes the transaction based on Assertion B.

Even though Assertion B has been issued by the Widget Marketplace in response to assertion A (I guess another way to look at this to view assertion A as the subject of B as in [1]) there is no way to represent this information within SAML.

If there is a problem with Ravi's purchases at the Widget Marketplace (Ravi wont pay his bills) there is nothing in the SAML flow that ties Assertion B to Assertion A. This appears to be a significant missing piece to me.

Status: Open

CLOSED ISSUE:[DS-5-02: Authenticator Reference]

The authenticator element of an assertion should be able to reference another assertion, used solely for authentication.

Status: Closed by vote on Sept 4. This approach was not used.

CLOSED ISSUE:[DS-5-03: Role Reference]

The role element should be able to reference another assertion that asserts the attributes of the role.

Status: Closed by vote on Sept 4. Role is no longer part of the core schema.

ISSUE:[DS-5-04: Request Reference]

There should be a way to reference an assertion as the subject of a request. For example, a request might reference a Attribute Assertion and ask if the subject of that assertion could access a specified object.

Status: Open

Group 6: Attributes

ISSUE:[DS-6-01: Nested Attributes]

Should SAML support nested attributes? This means that for example, a role could be a member of another role. This is one standard way of distinguishing the semantics of roles from groups.

There are many issues of semantics and pragmatics related to this. These include:

1. Limit of levels if any
2. Circular references
3. Distributed definition
4. Mixed attribute types.

Status: Open

CLOSED ISSUE:[DS-6-02: Roles vs. Attributes]

Should Attributes and Roles be identified as separate objects?

Status: Closed by vote on Sept 4. Core no longer contains roles.

CLOSED ISSUE:[DS-6-03: Attribute Values]

Should Attributes have some ‘attribute-value’ type structure to them?

Status: Closed by vote on Sept 4. Current core defines element Attribute to have three sub-elements, optional namespace, required name and one or more values. Values in turn may be defined in another namespace.

ISSUE:[DS-6-04: Negative Roles]

Should there be a way to state that someone does not have a role?

Status: Open

ISSUE:[DS-6-05: AttributeScope]

Should the core schema specify a way to express an attributes scope, or should this be left as a part of the structure of the attribute? Scope has essentially the same meaning as security domain. See DS-8-01 and DS-8-03.

Champion: Scott Cantor

Status: Open

Group 7: Authentication Assertions

CLOSED ISSUE:[DS-7-01: AuthN Datetime]

An Authentication Assertion should contain the date and time that the Authentication occurred. This could be done by explicitly assigning this meaning to the IssueInstant or NotBefore elements or create a new element containing a datetime.

Possible Resolutions:

1. Use IssueInstant in a AuthN Assertion to indicate datetime of AuthN.
2. Use NotBefore in a AuthN Assertion to indicate datetime of AuthN.
3. Create a new element to indicate datetime of AuthN.

Status: Closed by vote on Sept 4. Current core contains AuthenticationInstant, satisfying this issue.

CLOSED ISSUE:[DS-7-02: AuthN Method]

An element is required in AuthN Assertions to indicate the method of AuthN that was used. This could be a simple text field, but the values should be registered with some central authority. Otherwise different identifiers will be created for the same methods, harming interoperability.

Core-12 addresses this issue with AuthenticationCode. CONS-12 asks: what restrictions, if any, should be placed on the format of the contents of the AuthenticationCode element? Should this be a closed list of possible values? Should the list be open, but with some “well-known” values? Should we refer to another list already in existence?

Are the set of values supported for the <Protocol> element (DS-8-03) essentially the same as those required for the <AuthenticationCode> element?

Status: Closed by vote on Sept 4. Current core contains AuthenticationMethod, satisfying this issue.

ISSUE:[DS-7-03: AuthN Method Strength]

SAML has identified a requirement to indicate that a negative AuthZ decision might be changed if a “stronger” means of AuthN was used. In support of this it is useful to introduce the concept of AuthN strength. AuthN strength is an element containing an integer representing strength of AuthN, where a larger number is considered stronger. Individual deployments could assign numbers to particular AuthN methods according to their policies. This would allow an AuthZ policy to state that the required AuthN must exceed some value.

Possible Resolutions:

1. Add an AuthN strength element.
2. Do not add an AuthN strength element.

Status: Open

ISSUE:[DS-7-04: AuthN IP Address]

Should an AuthN Assertion contain the (optional) IP Address from which the Authentication was done? This information might be used to require that other requests in the same session originate from the same source. Alternatively it might be used as an input to an AuthZ decision or simply recorded in an Audit Trail.

One reason not to include this information is that it is not authenticated and can be spoofed. Also requiring that the IP address match future requests may cause spurious errors when firewalls or proxies are used. On the other hand, many systems today use this information.

This was identified as F2F#3-12.

Possible Resolutions:

1. Add IP Address to the AuthN Assertion schema.
2. Do not add IP Address to the AuthN Assertion schema.

Status: Open

ISSUE:[DS-7-05: AuthN DNS Name]

Should the AuthN Assertion contain an (optional) DNS name, distinct from the DNS name indicating the security domain of the Subject? If so, what are the semantics of this field?

An obvious answer is that the DNS name is the result of doing a reverse lookup on the IP Address from which the Authentication was done. This suggests that there is a relationship between this issue and DS-7-04. Presumably if the IP Address is not included in the specification, this field will not be either. However if IP Address is included, DNS name might still not be.

The DNS name in the subject represents the security domain that knows how to authenticate this subject. The DNS name of authentication would reflect the location from which the Authentication was done. These will often be different from each other.

This value might be used for AuthZ decisions or Audit. Of course, a reverse lookup could be done on the IP Address at a later time, but the result might be different. Like the IP Address, the DNS name is not authenticated and could be spoofed, either by spoofing the IP Address or impersonating a legitimate DNS server.

This was identified as F2F#3-13.

Possible Resolutions:

3. Add DNS Name to the AuthN Assertion schema.
4. Do not add DNS Nameto the AuthN Assertion schema.

Status: Open

ISSUE:[DS-7-06: DiscoverAuthNProtocols]

Should SAML provide a means to discover supported types of AuthN protocols?

Simon Godik has suggested: One way to do it is to use AuthenticationQuery with empty Authenticator subject. Then SAMLRequest will carry AuthenticationAssertion with Authenticator subject listing acceptable protocols.

The problem is that Authenticator element does not allow for 0 occurances of Protocol. Should we specify minOccurs=0 on Protocol element for that purpose?

Possible Resolutions:

1. Declare AuthN Protocol discovery out of scope for SAML V1.0.
2. Support it in the way suggested.
3. Support it some other way.

Status: Open

Group 8: Authorities and Domains

The following points are generally agreed.

- An Assertion is issued by an Authority.
- Assertions may be signed.
- The name of a subject must be qualified to some security domain.
- Attributes must be qualified by a security domain as well.
- Nigel Edwards has suggested that resources also need to be qualified by domain.

ISSUE:[DS-8-01: Domain Separate]

Stephen Farrell has pointed out that there may be a requirement to encrypt, for example, the user name but not the domain. Therefore they should be in separate elements. If domains are going to appear all over the place, maybe we need a general way of having element pairs or domain and "thing in domain."

Possible Resolutions:

1. Domains will always appear in a distinct element from the item in the domain
2. The domain and item may be combined in a single element.

Status: Open

CLOSED ISSUE:[DS-8-02: AuthorityDomain]

Should SAML take any position on the relationship between the 1) Authority, 2) the entity that signed the assertion, and 3) the various domains scattered throughout the assertion? For example, the Authority and Domain could be defined to be the same thing. Alternatively, Authorities could assert for several domains, but each domain would have only one authority. Another possibility would be to require that the domain asserted for be the same as that found in the Subject field of the PKI certificate used to sign the assertion.

The contrary view is that is a matter for private arrangement among asserting and relying parties.

At F2F #3 this issue was raised in the form of:

- F2F#3-15: Can an Authentication Authority issue assertions "for" ("from") multiple domains?
- F2F#3-16: Can multiple Authentication Authorities issue assertions "for" a given single

domain?

The general consensus from F2F #3 was that an Authority (Asserting Party) of any type can issue Assertions about multiple domains and multiple Authorities can issue Assertions about the same domain. However, this issue has not been officially closed.

Status: Closed by vote on Sept 4. There is nothing in the current core to prevent Authorities from issuing Assertions about Subjects in multiple domains or to prevent multiple Authorities from issuing Assertions about Subjects in the same domain.

ISSUE:[DS-8-03: DomainSyntax]

What is the composition of a “security domain” specifier? What is their syntax? What do they designate? Are they arbitrary or are they structured? JeffH has suggested that they are essentially the same as Issuer identifiers.

This was identified as F2F#3-11.

Core-12 addresses this issue with SecurityDomain. CONS-08 asks: Should the type of the <SecurityDomain> element of a <NameIdentifier> have additional or different structure?

Status: Open

ISSUE:[DS-8-04: Issuer]

Does the specification (core-12) need to further specify the Issuer element? Is a string type adequate for its use in SAML? See also DS-4-04.

This was identified as CONS-05.

Status: Open

Group 9: Request Handling

ISSUE:[DS-9-01: AssertionID Specified]

SAML should define the responses to requests that specify a particular AssertionID. For example,

- What if the assertion doesn't exist or has expired?
- What if the assertion contents do not match the request?
- Is it ever legal to send a different assertion?

Status: Open

ISSUE:[DS-9-02: MultipleRequest]

Should SAML provide a means of requesting multiple assertion types in a single request? This has been referred to as "boxcaring." In simplest form this could consist of concatenating several defined requests one message. However there are usecases in which it would convenient to have the second request use data from the results of the first.

For example, it would be useful to ask for an AuthN Assertion by ID and for and Attribute Assertion referring to the same subject.

Potential Resolutions:

1. Do not specify a way to make requests for multiple assertions types in SAML V1.0.
2. Allow simple concatenation of requests in one message.
3. Provide a more general scheme for multiple requests.

Status: Open

ISSUE:[DS-9-03: IDandAttribQuery]

Should SAML allow queries containing both an Assertion ID and Attributes?

Tim Moses comments: The need to convey an assertion id and attributes in the same query arises in the following circumstances.

A browser contacts a content site and is redirected to an authentication site. The content site has specific requirements for:

1. The authentication scheme between the browser and the authentication site (I'll call this

"primary" authentication);

2. The authentication scheme between the browser and the content site upon its return to the content site (I'll call this "secondary" authentication, normally this would be a bearer token, but who knows?);

3. The space in which the subject's name should appear; and

4. User attributes.

So, the content site needs to communicate its requirements in these four areas to the authentication site, preferably, before primary authentication takes place.

There is currently no fully-specified way for the content site to communicate its needs to the authentication site. What are the possible solutions?

1. The authentication site "just knows" what authentication schemes, namespaces and attributes the content site needs.

2. Each authentication site URL corresponds to a single authentication scheme. Then the content site specifies the authentication scheme by redirecting the browser to the appropriate URL.

3. The authentication site returns assertions containing every authentication scheme, namespace and additional attribute, and the content site searches through them for the ones that suit its needs.

4. The authentication site returns its own choice of authentication assertion and the content site submits a further query for any additional, or alternative, assertions that it needs.

Solution 1 works because we don't.

Solution 2 addresses requirement 1, but not requirements 2, 3 and 4.

Solution 3 is unsatisfactory from an identity-theft/privacy point of view.

Solution 4 introduces more delay than is absolutely necessary.

We have, in both the "fat object" and "artifact" browser profiles, opportunities to solve these questions in a more satisfactory manner.

In the "fat object" profile, the "form" can contain the Assertion Queries. In the "artifact" profile, the initial redirection by the content site to the authentication site can contain an artifact, in the redirection URL, corresponding to the Assertion Queries, using either of the push or pull communication models. The thing that is new and surprising about this approach is that the artifact does not correspond to an "assertion", but to a "query". There would then have to be a communication directly between the content and authentication sites in which the content site would request assertions that directly meet its needs.

This is what it looks like in both the "push" and "pull" models.

Push model

```
Browser                 Content site          Authentication site

1 <---- redirect(artifact1) ----
2 --------------------- redirect(artifact1)--------------->
3                        ---- query(artifact1) ---->
4 <----------------------- authenticate ------------------>
5                        <- assertions(artifact2) --
6 <-----------------------------------redirect(artifact2)--
7 -------redirect(artifact2)--->
```

Pull model

```
Browser                 Content site          Authentication site

1 <---- redirect(artifact1) ----
2 --------------------- redirect(artifact1) --------------->
3 <------------------------ authenticate ------------------>
4                       <- request query(artifact1) -
5                       ----- query(artifact2) ----->
7                       <-------- assertions --------
6 <------------------- redirect(artifact2) ----------------
7 -----redirect(artifact2)---->
```

Line 3 of the push model and line 5 of the pull model involve a query with both an artifact (or assertion id) and the set of requested attributes.

Possible Resolutions:

1. Allow queries to specify both an Assertion ID and Attributes
2. Only allow queries to specify one or the other.

Status: Open

ISSUE:[DS-9-04: AssNType in QuerybyArtifact]

When an Assertion is requested by providing an Artifact, there should be a way to refer to which type of Assertion is being requested. Originally, an Artifact referred to a specific Assertion, so this was not required. However, under current design, an Artifact may refer to both an Authentication Assertion and an Attribute Assertion.

Champion: Simon Godik

Status: Open

ISSUE:[DS-9-05: RequestAttributes]

We should be able to pass request attributes to the issuing party.

I would like to propose addition to the RequestType:

```
<complexType name="RequestType">
        <complexContent>
                <extension base="samlp:RequestAbstractType">
                        <sequence>
                                <element ref="saml:Attribute" minOccurs="0" maxOccurs="unbounded"/>
                                <choice>
                                        -- same as before --
                                </choice>
                        <sequence>
                </extension>
        </complexContent>
</complexType>
```

Champion: Simon Godik

Status: Open

ISSUE:[DS-9-06: Locate AttributeAuthorities]

Should an Authentication Assertion provide the means to locate Attribute Authorities with information about the same subject?

Context here is that Authentication Authority can front several Attribute Authorities
as in the case of Shibboleth. Authentication Authority should be able to point
to the correct Attribute Authority for authenticated subject by including information
about Attribute Authority in AuthenticationAssertion.

Proposed text:

SAML assumes that given authentication assertion relying party can find
attribute authority for the authenticated subject.

In a more dynamic situation Authentication Authority can be placed in front
of a number of Attribute Authorities. In this case Authentication Authority
may want to direct relying parties to the specific Attribute Authorities at the
time when authentication assertion is issued.

AuthorityBinding element specifies the type of authority (authentication, attribute, authorization) and points to it via URI. AuthenticationStatementType contains optional list of AuthorityBinding's. All AuthorityBinding's in the list must be of the 'attribute' type. Any authority pointed to by the AuthorityBinding list may be queried by the relying party.

```
<element name="AuthorityBinding" type="saml:AuthorityBindingType"/>
<complexType name="AuthorityBindingType">
        <attribute name="AuthorityKind">
                <simpleType>
                        <restriction base="string">
                                <enumeration value="authentication"/>
                                <enumeration value="attribute"/>
                                <enumeration value="authorization"/>
                        </restriction>
                </simpleType>
        </attribute>
        <attribute name="Binding" type="anyURI"/>
</complexType>
```

```
        <element name="AuthenticationStatement" type="saml:AuthenticationStatementType"/>
        <complexType name="AuthenticationStatementType">
                <complexContent>
                        <extension base="saml:SubjectStatementAbstractType">
                                <sequence>
                                        <element ref="saml:AuthenticationLocality" minOccurs="0"/>
                                        <element ref="saml:AuthorityBinding" minOccurs="0"
maxOccurs="unbounded"/>

                                </sequence>
                                <attribute name="AuthenticationMethod" type="anyURI"/>
                                <attribute name="AuthenticationInstant" type="dateTime"/>
                        </extension>
                </complexContent>
        </complexType>
```

Champion: Simon Godik

Status: Open

ISSUE:[DS-9-07: Request Extra AuthzDec Info]

Should the Authorization Decision Request be able to request additional information relating to the Actions specified?

Champion: Simon Godik

Status: Open

ISSUE:[DS-9-08: No Attribute Values in Request]

Is it intended that when AttributeDesignator from the saml: namespace is reused in the protocol schema (for an AttributeQuery), you're supposed to supply the AttributeValue? I would think that in an assertion you do want to spell out an attribute value, but in a query you just want to ask for the attribute of the specified name, without parameterizing it by the value.

E.g., if I want to know the PaidStatus of a subscriber to a service, I would just say "Please give me the value of the PaidStatus attribute" -- I wouldn't say "Please give me the PaidStatus=PaidUp attribute". Right??

If we want to change this, we would need to have something like a base AttributeDesignatorType (and an AttributeDesignator element) in saml: that just has AttributeName and AttributeNamespace (currently XML attributes). Then we should extend it in samlp: to get an AttributeValueType (and an AttributeValue element) that adds an element called AttributeValue.

Champion: Eve Maler

Status: Open

ISSUE:[DS-9-09: Drop CompletenessSpecifier]

CompletenessSpecifier was intended to control the behavior of requests for Attribute Assertions, when an Authority could only partly fulfill requests for enumerated attributes. However, much confusion was generated over the proper behavior, error responses and general motivation for this feature. It is proposed that the CompletenessSpecified be dropped entirely.

Champion: Eve Maler

Status: Open

ISSUE:[DS-9-10: IssueInstant in Req&Response]

Should IssueInstant be added to Request and Response messages? This would allow implementations to prevent replay attacks in environments where these are not prevented by other means.

Champion: Scott Cantor

Status: Open

Group 10: Assertion Binding

ISSUE:[DS-10-01: AttachPayload]

There is a requirement for assertions to support some structure to support their "secure attachment" to payloads. This is a blocking factor to creating a SOAP profile or a MIME profile. If needed, the bindings group can make a design proposal in this space but we would like input from the broader group.

Status: Open

Group 11: Authorization Decision Assertions

ISSUE:[DS-11-01: MultipleSubjectAssertions]

It has been proposed (WhiteboardTranscription-01.pdf section 4.0) that an Authorization Decision Assertion Request (and presumably the Assertion sent in response) may contain multiple subject Assertions (or their Ids). Must these assertions all refer to the same subject or may they refer to multiple subjects.

One view is that the assertions all provide evidence about a single subject who has requested access to a resource. For example, the request might include a Authentication Assertion and one or more Attribute Assertions about the same person.

Another view is that for efficiency or other reasons it is desirable to ask about access to a resource by multiple individuals in a single request. This raises the question of how the PDP should respond if some subjects are allowed and others are not.

The PDP might have the freedom to return a single, all encompassing Assertion in response or reduce the request in order to give a positive response or return multiple Assertions with positive and negative indications.

Identified as F2F#3-30 and F2F#3-31.

Possible Resolutions:

1. Require that all the assertions and assertion ids in a request refer to the same subject.
2. Treat assertions with different subjects as requesting a decision for each of the subjects mentioned.
3. Treat assertions with different subjects and a question about the collective group, i.e. true only if access is allowed for all.
4. Allow multiple subjects, but assign some other semantic to such a request.

Status: Open

ISSUE:[DS-11-02: ActionNamespacesRegistry]

Authorization Decision Assertions contain an object and an action to be performed on the object. Different types of actions will be appropriate in different situations, so an action will be qualified by an XML namespace. Should a public registry of namespaces be established somewhere? This would allow groups applying SAML to different fields of interest to define appropriate syntaxes.

This was identified as F2F#3-32. It relates to MS-2-01 and DS-7-02.

Identified as CONS-14.

Possible Resolutions:

1. Establish an action namespace registry.
2. Do not establish an action namespace registry.

Status: Open

CLOSED ISSUE:[DS-11-03: AuthzNDecAssnAdvice]

Should Authorization Decision Assertions contain an Advice field? If so, what are the semantics of Advice? It has been proposed that Conditions and Advice be fields that allow additional information relative to the Assertion to be included. The distinction being that a relying party could safely ignore items in Advice that it does not understand, but should discard an Assertion if it does not understand all the Conditions.

Such as scheme would allow for backward compatibility between SAML versions and/or the possibility of proprietary usages.

This was identified as F2F#3-33 and F2F#3-34.

Note this is closely related to DS-14-01.

Possible Resolutions:

1. Include Advice in AuthZDecAssns.
2. Do not include Advice in AuthZDecAssns.

Status: Closed by vote on Sept 4. Current core specifies an Advice element in all Assertion types.

ISSUE:[DS-11-04: DecisionTypeValues]

CONS-13 asks: does {Permit, Deny, Indeterminate} (as proposed in core12) cover the range of decision answers we need? See also discussion in [ISSUE:F2f#3-33]. (This is DS-11-03, not clear how this relates. ed.)

Status: Open

CLOSED ISSUE:[DS-11-05: MultipleActions]

The F2F #3 left it somewhat unclear if multiple actions are supported within an <Object>. There is clear advantage to this type of extension (as defined in core-12) as it provides a simple way to aggregate actions. Given that actions are strings (as opposed to pieces of XML) this does seem to provide additional flexibility within the SAML framework.

Does the TC support this type of flexibility?

This was identified as CONS-15.

Status: Closed by vote on Sept 4. Current schema allows multiple Actions to be specified.

ISSUE:[DS-11-06: Authz Decision]

Change the names of AuthorizationStatement and AuthorizationQuery to AuthorizationDecisionStatement and AuthorizationDecisionQuery to eliminate ambiguity.

Early in the process of this committee we decided, after much contention and explanation and careful thought about concepts and terminology, that one of our three assertions (now statements, of course) is an "Authorization Decision Assertion", where that name precisely captures the intent of the structure. In particular we observed as part of that discussion that the single word "authorization" by itself can mean so many different things that it has to be qualified to be useful. The text of core-20, in section 1, uses the term "Authorization Decision Assertion", and section 1.5 has this phrase as its title.

However, the actual name of the element, as specified in section 1.5 and elsewhere, is "AuthorizationStatement". And, the name of the corresponding query element, as specified in section 2.5, is "AuthorizationQuery". It seems to me that these names are misleading and should be changed. This is especially true since a likely user of our statement structures is the XACML work, which (though I haven't followed it) is supposedly about managing and expressing authorization information.

So, I strongly suggest that these elements be renamed "AuthorizationDecisionStatement" and "AuthorizationDecisionQuery" and that the corresponding types be similarly renamed.

Champion: Bob Morgan

Status: Open

Group 12: Attribute Assertions

CLOSED ISSUE:[DS-12-01: AnyAllAttrReq]

Should an Attribute Assertion Request be allowed to specify “ANY” and/or “ALL”? If so, what attributes should be returned and should an error be returned in for ANY and for ALL in each of the following case:

- Subject possesses all requested attributes
- Subject possesses some of requested attributes, but the others exist
- Subject possesses some of requested attributes, but others do not exist
- Subject possesses some requested attributes which are not permitted to be returned to this relying party because of privacy policy
- Subject possesses none of requested attributes, but does possess others
- All of attributes possessed by this subject are not permitted to be returned to this relying party because of privacy policy
- Attribute Authority has no information about this subject

An arguably common attribute authority implementation will be one layered over an LDAP-based directory service. The LDAP-based directory semantics presented to such an attribute authority are noted in [F3], below. Multiple attrs, of an entry, may be requested in a given LDAP search/read request. Note that there are no errors returned about whether or not specific attributes were found in the entry or not; LDAP does return errors about whether the entry itself was found, or not. If SAML mandates that the Attr Authority MUST return errors about each individually requested attribute, then that will make layering an Attr Authority over an LDAP-based directory arguably harder. One approach would be to store each individual attribute of a subject in an individual directory entry subordinate to an entry representing the subject. Whether forcing such a design on Attr Authority designers/implementors/deployers is reasonable or not is debatable.

```
[F3] nuances of LDAPv3 responses wrt attributes
-----------------------------------------------
>From http://www.ietf.org/rfc/rfc2251.txt, section 4.5.1, pages 25 & 26...

        SearchRequest ::= [APPLICATION 3] SEQUENCE {
                baseObject      LDAPDN,
                scope           ENUMERATED {
                        baseObject              (0),
                        singleLevel             (1),
                        wholeSubtree            (2) },
                derefAliases    ENUMERATED {
                        neverDerefAliases       (0),
                        derefInSearching        (1),
                        derefFindingBaseObj     (2),
                        derefAlways             (3) },
                sizeLimit       INTEGER (0 .. maxInt),
                timeLimit       INTEGER (0 .. maxInt),
                typesOnly       BOOLEAN,
                filter          Filter,
                attributes      AttributeDescriptionList }
                    ^                       ^
                    +-----------------------+
            This is where the client specifies the list of attrs to return
            from each directory entry that matches the baseobject and/or
            filter.

>From rfc2251, section 4.5.1, pages 29...

   - attributes: A list of the attributes to be returned from each entry
     which matches the search filter. There are two special values which
     may be used: an empty list with no attributes, and the attribute
     description string "*".  Both of these signify that all user
     attributes are to be returned.  (The "*" allows the client to
     request all user attributes in addition to specific operational
     attributes).

     Attributes MUST be named at most once in the list, and are returned
     at most once in an entry.   If there are attribute descriptions in
     the list which are not recognized, they are ignored by the server.

     If the client does not want any attributes returned, it can specify
     a list containing only the attribute with OID "1.1".  This OID was
     chosen arbitrarily and does not correspond to any attribute in use.

     Client implementors should note that even if all user attributes are
     requested, some attributes of the entry may not be included in
     search results due to access control or other restrictions.
     Furthermore, servers will not return operational attributes, such
     as objectClasses or attributeTypes, unless they are listed by name,
     since there may be extremely large number of values for certain
     operational attributes. (A list of operational attributes for use
     in LDAP is given in [5].)

-----------------------------------------------
[end of F3]
```

This was identified as F2F#3-20, F2F#3-24 and F2F#3-25.

PRO-03 asks if core-12 satisfies this issue.

PRO-05 asks: Is the all or “error” semantics (in core-12) for the ALL qualifier appropriate?

Should we just follow LDAP semantics for this type of query?

Status: Closed by vote on Sept 4. At that time the core schema proposed a choice of “Partial” of “AllOrNone” in the CompletnessSpecifier. (The CompletenessSpecifier was subsequently dropped entirely.)

CLOSED ISSUE:[DS-12-02: CombineAttrAssnReqs]

It has been proposed (WhiteboardTranscription-01.pdf section 4.0) that it be possible 1) to request all of the attributes of a subject and also 2) to request ANY and/or ALL attributes (with specific error semantics. Can requests of type 1 and 2 be accommodated in a single request structure? If not, the reasons for having distinct types should be documented.

This was identified as F2F#3-21.

PRO-03 asks if core-12 satisfies this issue.

Possible Resolutions:

1. Combine the requests.
2. Leave them as distinct types and document the reason.

Status: Closed by vote on Sept 4. Both all and specified attributes can be requested.

ISSUE:[DS-12-03: AttrSchemaReqs]

Should it be possible to request only the Attribute schema?

This was identified as F2F#3-22.

Possible Resolutions:

1. Allow Attribute Schema Requests.
2. Do not allow Attribute Schema Requests.

Status: Open

ISSUE:[DS-12-04: AttrNameReqs]

Should it be possible to request only attribute names and not values? It is not clear whether these would be all the attributes the Attribute Authority knows about or just the ones pertaining to a particular subject. It is not clear what this would be used for. No usecase seems to require it.

This was identified as F2F#3-23.

This was identified as PRO-04.

Possible Resolutions:

3. Allow Attribute Name Requests.
4. Do not allow Attribute Name Requests.

Status: Open

CLOSED ISSUE:[DS-12-05: AttrNameValueSyntax]

What is the syntax of attribute names and values? Should attribute names be qualified by an xml namespace? Should an attribute value be a monolithic opaque thing, with any internal syntax agreed to out-of-band, or something with perceivable-in-protocol-context internal structure? Does the use of XPath [http://www.w3.org/TR/xpath] in AttrAssnReqs mitigate the restrictiveness of having attr values being monolithic opaque things, presumably where the value is actually XML encoded and having arbitrarily complexity?

- One possible approach is to use XPath in AttrAssnReqs.
- Another approach is to define a very simple name/value pairs. A problem with this is that, if the users/developers want to formulate any kind of structured values, they have to flatten them into the SAML-defined thing. Thus the concern is how do we allow for flexible (i.e. complex) value structures without unduly complicating AttrAssnReqs & AttrAssnResps?

This was identified as F2F#3-28, F2F#3-29 and F2F#3-37.

PRO-06 asks if the simple queries proposed in core-12 are sufficient.

Status: Closed by vote on Sept 4. Schema allows both names and values to have namespaces.

ISSUE:[DS-12-06: RequestALLAttrbs]

How should a request for all available attributes be made? Some have objected to the idea that if no attributes are specified it means “all”.

This should not be confused with the Completeness Specifier AllOrNothing (formerly ALL) which controls what should be returned when a request cannot be fully satisfied.

Potential Resolutions:

1. Declare an empty list of attributes to mean “all attributes.”
2. Define a reserved keyword, such as “AllAttributes” for this purpose.

Status: Open

Group 13: Dynamic Sessions

ISSUE:[DS-13-01: SessionsinEffect]

How can a relying party determine if dynamic sessions are in effect? If dynamic sessions are in effect it will be necessary to determine if the session has ended, even if the relevant Assertions have not yet expired. However, if dynamic sessions are not in use, attempting to check session state is likely to increase response times unnecessarily.

This was identified as F2F#3-3.

Proposed Resolutions:

1. Define a field in Assertion Headers to indicate dynamic sessions.
2. Configure the implementation based on some out of band information.

Status: Open

Group 14:General – Multiple Message Types

CLOSED ISSUE:[DS-14-01: Conditions]

Should Assertions contain Conditions and if so, what items should be included under conditions and what should the semantics of conditions be?

It has been proposed that Conditions and Advice be fields that allow additional information relative to the Assertion to be included. The distinction being that a relying party could safely ignore items in Advice that it does not understand, but should discard an Assertion if it does not understand all the Conditions.

In addition to general design and rationale, the following questions have been posed. Should Audience be under Conditions? Should Validity Interval be under Conditions? What sort of extensibility should be allowed: upward compatibility between SAML versions? Proprietary extensions? Other types?

At F2F #3, the following straw poll results were obtained:

- Yes, we want something with the semantic of "conditions" to appear in Assertions.
- Yes, we need to re-work the design of conditions.
- Yes, we want to place the validity interval into the conditions (However, it was noted that doesn't this make validity interval optional? Do we want that?)
- "Maybe" to providing a general conditions framework
- "Maybe" to putting audiences into conditions

This was identified as F2F#3-17 and F2F#3-18.

Note this is closely related to DS-11-03.

Core-12 addresses this issue with ConditionsType. CONS-07 asks: Does the ConditionsType meet the TC's requirements? If not, why not?

Status: Closed by vote on Sept 4. Schema contains a Conditions element.

ISSUE:[DS-14-02: AuthenticatorRequired]

It has been proposed that an Assertion may contain an Authenticator element which can be used in any of a number of ways to associate the Assertion with a request, either directly or indirectly via some cryptographic primitive. Should this element be a part of SAML?

Basically the question is whether the complexity associated with supporting this mechanism is absolutely required or simply “nice to have.”

This has been identified as F2F#3-14.

Potential Resolutions:

1. Include the Authenticator element.
2. Do not include the Authenticator element.

Status: Open

CLOSED ISSUE:[DS-14-03: AuthenticatorName]

Assuming DS-14-02 is resolved affirmatively, should the Authenticator be called something else? Suggestions include: HolderofKey and Subject Authenticator.

This has been identified as F2F#3-10.

Also identified as CONS-09.

Status: Closed by vote on Sept 4. Schema now contains SubjectConfirmation element for this purpose.

ISSUE:[DS-14-04: Aggregation]

Do we need an explicit element for aggregating multiple assertions into a single object as part of the SAML specification? If so, what is the type of this element?

This was identified as CONS-01.

Status: Open

ISSUE:[DS-14-05: Version]

Does the specification (core-12) need to further specify the version element? If so, what are these requirements? Should this be a string? Or is an `unsignedint` enough?

This was identified as CONS-06

Status: Open

ISSUE:[DS-14-06: ProtocolIDs]

Core-12 proposes a <Protocol> element with the AuthenticatorType. CONS-10 suggests that the TC will develop a namespace identifier (e.g., protocol) and set of standard namespace specific strings for the <Protocol> element above. If not, what approach should be taken here?

Status: Open

ISSUE:[DS-14-07: BearerIndication]

Core-12 proposes the following for identifying a ``bearer'' assertion: A distinguished URI urn:protocol:bearer be used as the value of the <Protocol> element in <Authenticator> with no other sub-elements. CONS-11 asks: Is this an acceptable design?

Status: Open

ISSUE:[DS-14-08: ReturnExpired]

Should the specification make any normative statements about the expiry state of assertions returned in response to SAMLRequests? Is it a requirement that only unexpired assertions are returned, or is the client responsible for checking? (*Seems pretty clear that the client will have to check anyway at time-of-use, so forcing the responder to check before replying seems like extra processing.*)

Note that regardless of how this issue is settled, Asserting Parties will be free to discard expired Assertions at any time.

Identified as PRO-01.

Possible Resolutions:

1. The specification will state that Asserting Parties MUST return only Assertions that have not expired.
2. The specification will state that Asserting Parties MAY return expired Assertions.
3. The specification will make no statement about returning expired Assertions.

Status: Open

ISSUE:[DS-14-09: OtherID]

PRO-01 states: in some instances (such as the web browser profile) it is necessary to lookup an assertion using an identifier other than the <AssertionID>. Typically, such an identifier is opaque and may have been created in some proprietary way by an asserting party. Do we need an additional element in SAMLRequestType to model this type of lookup?

Status: Open

ISSUE:[DS-14-10: StatusCodes]

PRO-07 asks: are the status codes listed for StatusCodeType (in core-12) sufficient? If not how do we want to define a bigger list: keep it open with well-known values, use someone else's list, define an extension system, etc.

See also ISSUE:[F2F#3-33, 34].(Not clear the relationship. These issues are about Advice. ed.)

Status: Open

ISSUE:[DS-14-11: CompareElements]

Should SAML specify the rules for comparing various identifiers, such as Assertion IDs, Issuer, Security Domain, Subject Name? Currently these are all specified as strings. Issues include:

- Upper and lower case equivalence
- Leading and trailing whitespace
- Imbedded whitespace

Possible Resolutions:

1. Declare only exact binary matching.
2. Define a set of matching rules.

Status: Open

ISSUE:[DS-14-12: TargetRestriction]

Add a new condition type to the schema called TargetRestriction.

The "Form POST" web browser profile of SAML (bindings-06, section 4.1.6) identifies a particular security threat (4.1.6.1.1, bullet 3), which is that a malicious site, receiving an asserted authentication statement via POST, might replay the assertion to some other site, in an attempt to pose as the subject of the statement (ie, the authenticated user). The identified countermeasure for this threat is to include information in the assertion that restricts its use to the site to which the POST is done. In that case, if the malicious site attempts to replay the assertion somewhere else, the receiver will see the mismatch and reject the assertion.

Up to now the profile has called for the use of the AudienceRestrictionCondition element to carry this information. However, we have argued that this condition, though similar, is actually different in use, so a new condition is needed. There was discussion of this point at the recent F2F in San Francisco, and the group agreed to add a new condition for this purpose.

The justifications are as follows. First, the existing text on AudienceRestrictionCondition (core-20, section 1.7.2) describes a more policy-based use, to limit the use of the assertion to receivers conforming to some policy statement. Shibboleth, for example, would use this condition to indicate that an assertion conforms to conditions including non-traceability of subject name, user agreement with attribute release, etc. This description would have to be rewritten to also support the more specific restriction required by the POST profile (which could be done).

A more telling issue is matching. While the current description of Audience doesn't say how matching is done (should it?), it seems likely that in practice these policy URIs would be complete and opaque; that is, the receiver would simply do a string match on its available set of policy URIs. A URI "http://example.com/policy1" has no necessary relation to "http://example.com/policy2". On the other hand, for the POST profile, the most likely approach would be for the assertion issuer to include the entire target URL in the assertion. The assertion receiver would then have to match on some substring of the URL to determine whether to accept the assertion. If the same condition were to be used for both purposes the receiver would have to do matching based on the value of the URI, which seems suboptimal.

Cardinality is another issue. It's reasonable for multiple AudienceRestriction elements to be included to indicate that the recipient should be bound by all the indicated policies. But it doesn't really make sense to say the recipient has to be named by multiple names.

Champion: Bob Morgan

Status: Open

ISSUE:[DS-14-13: StatusCodes]

How should SAML Requests report errors? Many suggestions have been made, ranging from a simple list of error codes to adopting SOAP error codes. Scott proposes:

SAML needs an extensible, more flexible status code mechanism. This proposal is a hierarchical Status structure to be placed inside Response as a required element. The Status element contains a nested Code tree in which the top level Value attribute is from a small defined set that SAML implementations must be able to create/interpret, while allowing arbitrary detail to be nested inside, for applications prepared to interpret further.

I mirrored some of SOAP's top level fault codes, while keeping SAML's Success code, which doesn't exist in SOAP, since faults mean errors, not status. I also eliminated the Error vs Failure distinction, which seems to be intended to "kind of" mean Receiver/Sender, which is better made explicit. Unknown didn't make sense to me either. Please provide clarifications if these original codes should be kept.

The proposed schema is as follows, replacing the current string enumeration of StatusCodeType with the new complex StatusType:

```
<simpleType name="StatusCodeEnumType">
   <restriction base="QName">
      <enumeration value="samlp:Success"/>
      <enumeration value="samlp:VersionMismatch"/>
      <enumeration value="samlp:Receiver"/>
      <enumeration value="samlp:Sender"/>
```

```
    </restriction>
</simpleType>
<complexType name="StatusCodeType">
    <sequence>
        <element name="Value" type="sampl:StatusCodeEnumType"/>
        <element name="Code" type="samlp:SubStatusCodeType"
minOccurs="0"/>
    </sequence>
</complexType>
<complexType name="SubStatusCodeType">
    <sequence>
        <element name="Value" type="QName"/>
        <element name="Code" type="samlp:SubStatusCodeType"
minOccurs="0"/>
    </sequence>
</complexType>
<complexType name="StatusType">
    <sequence>
        <element name="Code" type="samlp:StatusCodeType"/>
        <element name="Message" type="string" minOccurs="0"
maxOccurs="unbounded"/>
        <element name="Detail" type="anyType" minOccurs="0"/>
    </sequence>
</complexType>
```

In Response, delete the StatusCode attribute, and add:

```
<element name="Status" type="samlp:StatusType"/>
```

Champion: Scott Cantor

Status: Open

Miscellaneous Issues

Group 1: Terminology

CLOSED ISSUE:[MS-1-01: MeaningofProfile]

The bindings group has selected the terminology:

- SAML Protocol Binding, to describe the layering of SAML request-response messages on "top" of a substrate protocol, Example: SAML HTTP Binding (SAML request-response messages layered on HTTP).
- a profile for SAML, to describe the attachment of SAML assertions to a packaging framework or protocol, Example: SOAP profile for SAML, web browser profile for SAML

This terminology needs to be reflected in the requirements document, where the generic term "bindings" is used. It needs also to be added to the glossary document.

The conformance group has used the term Profile to define a set of SAML capabilities, with a corresponding set of test cases, for which an implementation or application can declare conformance. This use of profile is consistent with other conformance programs, as well as in ISO/IEC 8632. In order to resolve this conflict, the conformance group has proposed, in sstc-draft-conformance-spec-004, to substitute the word partition instead.

Status: Closed by vote on Sept 4. The terminology of the bindings group, as specified in the second bullet point above, has been accepted by the TC.

Group 2: Administrative

ISSUE:[MS-2-01: RegistrationService]

There is a need for a permanent registration service for publishing bindings and profiles. The bindings group specification will provide guidelines for creating a protocol binding or profile, but we also need to point to some form of registration service.

DS-7-02: AuthN Method also implies a need to register AuthN methods.

How can we take this forward? Is OASIS wiling to host a registry?

Another possibility is IANA.

Status: Open

Group 3: Conformance

CLOSED ISSUE:[MS-3-01: BindingConformance]

Should protocol bindings be the subject of conformance? The bindings sub group is defining both SAML Bindings and SAML Profiles. It has been proposed that both of these would be the subject of independent conformance tests.

The following definitions have been proposed:

SAML Binding: SAML Request/Response Protocol messages are mapped onto underlying communication protocols. (SOAP, BEEP)

SAML Profile: formats for combining assertions with other data objects. These objects may be communicated between various system entities. This might involve intermediate parties.

This suggests that a Profile is a complete specification of the SAML aspects of some use case. It provides all the elements needed to implement a real world scenario, including the semantics of the various SAML Assertions, Requests and Responses.

A Binding would simply specify how SAML Assertions, Requests and Responses would be carried by some protocol. A Binding might be used as a building block in one or more Profiles, or be used by itself to implement some use case not covered by SAML. In the later case, it would be necessary for the parties involved to agree on all aspects of the use case not covered by the Binding.

Thus conformance testing of Bindings might be undesirable for two related reasons:

- The number of independent test scenarios is already large. It seems undesirable to test something that does not solve a complete, real-world problem.
- Parties would be able to claim “SAML Conformance” by conforming to a Binding, although they would not be able to actually interoperate with others in a practical situation, except by reference to a private agreement. This would likely draw a negative response from end users and other observers.

The advantages of testing the conformance of Bindings include:

- Simplifying testing procedures when a Binding is used in several Profiles that a given party wishes to conform to.
- Allow SAML to be used in scenarios not envisioned by the Profiles.

This was identified as F2F#3-2.

Possible Resolutions:

1. Make Bindings the subject of conformance.
2. Do not make Bindings the subject of conformance.

Status: Closed by vote on Sept 4. The conformance group has made a proposal which has been accepted by the TC.

CLOSED ISSUE:[MS-3-02: Browser Partition]

Should the Web Browser be a SAML Conformance Partition, different from the Authentication Authority partition?

This was identified as F2F#3-7.

Status: Closed by vote on Sept 4. The Browser is not a partition.

Group 4: XMLDSIG

ISSUE:[MS-4-01: XMLDsigProfile]

SAML should define an XMLDsig profile specifying which options may be used in SAML, in order to achieve interoperability.

One aspect of this is: which of the signature types: enveloped, enveloping and detached should be supported? See also Issues UC-7-01 and UC-7-02.

Status: Open

ISSUE:[MS-4-02: SOAP Dsig]

Exactly how should the use of digital signatures be specified in the SOAP profile?

The SOAP profile in the bindings-06 draft specifies that all SOAP messages which include SAML assertions must be signed. The current signature requirements are too restrictive; in particular, they are not compatible with SOAP header elements that have "actor" attributes.

I propose that we change lines 828-829 and 978-979 (.pdf version) to read:

The <dsig:Signature> element MUST apply to all the SAML assertion elements in the SOAP <Header>, and all the relevant portions of the SOAP <Body>, as required by the application. Specific applications may require that the signature also apply to additional elements.

(Do we need to say anything about whether the receiver should rely on unsigned portions of the SOAP message? My first inclination is that it's up to the application, so we shouldn't say anything. Perhaps we need something in security considerations?)

Champion: Irving Reid

Status: Open

Group 5: Bindings

ISSUE:[MS-5-01: SSL Mandatory for Web]

Should use of SSL be mandatory for the Web Browser Profile?

The issue originates from the mandatory use of HTTP(S) in 4.1.4.1 (SAML Artifact) and 4.1.4.3 (Form POST) between the browser equipped user and source and destination sites respectively. The essential issue therein is confidentiality of the SAML artifact (4.1.4.1) or SAML assertions (4.1.4.3). If we do not use HTTPS, the HTTP traffic between the user and source or destination can be copied and used for impersonation.

There was concern at this requirement at the F2F#4 and as Gil is away the action item has fallen to me. But I am genuinely puzzled as to how we can move away from this requirement.

(1) Should the text merely state that confidentiality is a requirement (MUST) (could be met in some unspecified way?) and that HTTPS MAY be used? I am opposed to this formulation as it is not specific enough to support inter-operability. How can a pair of sites collaborate to support the web browser profile if each uses some arbitrary method for confidentiality?

(2) Another approach would be to require confidentiality (MUST) and specify HTTPS as a mandatory-to-implement feature. Those sites that prefer to use some other method for confidentiality can do so, but all sites must also support HTTPS. This ensures inter-operability as we can always fall back on HTTPS.

Champion: Prateek Mishra

Status: Open

ISSUE:[MS-5-02: MultipleAssns per Artifact]

In the browser artifact profile as described in the bindings-06 document, section 4.1.5, lines 565-567 imply that more than one authentication assertion could be transferred. This raises all sorts of questions about how the receiver should behave, particularly if the authn assertions refer to different subjects.

Do we want to say anything more about this? Alternatives include:

(a) Make no changes to the spec. Implementers are free to choose whatever behavior they think is appropriate for their solution.

(b) Specify that all authn assertions must contain the same Subject (or at least, the same NameIdentifier within the Subject)

(c) Specify exactly how the receiver should behave. Two possibilities are to say that access should be allowed if any one of the Subjects would be allowed, or that access should only be allowed if all of the Subjects are allowed.

My life would be easiest if we choose (b), though I could see how it might be too severe a constraint on some applications.

Champion: Irving Reid

Status: Open

ISSUE:[MS-5-03: Multiple PartnerIDs]

Can a single URL contain handles to more than one PartnerID?

In Prateek's bindings-06 document on lines 518-519, when a user is transferred, more than one SAML Artifact could be passed on the URL.

The first question this raises is: can the artifacts contain more than one PartnerID? In the paragraph at lines 536-541, the description implies that all the assertions are pulled at once. This won't work if the artifacts have different PartnerIDs, and the partners have different access URLs.

I'd like to propose an addition to the paragraph at 518-519, adding the sentence:

When more than one artifact is carried on the URL query string, all the artifacts MUST have the same PartnerID.

Champion: Irving Reid

Status: Open

Document History

- 5 Feb 2001 First version for Strawman 2.
- 26 Feb 2001 Made the following changes:
    - Changed references to [SAML] to SAML.
    - Added rewrites of Group 1 per Darren Platt.
    - Added rewrites of Group 3 per David Orchard.
    - Added rewrites of Group 5 per Prateek Mishra.
    - Added rewrites of Group 11 per Irving Reid.
    - Converted the abbreviation "AuthC" (for "authentication") to "AuthN."
    - Added Group 13.
    - Added UC-1-12:SignOnService.
    - Converted candidate requirement naming scheme from [R-Name] (as used in the main document) to [CR-issuenumber-Name], per David Orchard.
    - Added UC-0-02:Terminology.
    - Added UC-0-03:Arrows.
    - Updated UC-9-02:PrivacyStatement with suggested requirements from Bob Morgan and Bob Blakley.
    - Added UC-1-13:ProxyModel per Irving Reid.
    - Added status indications for each issue.
    - Recorded votes and conclusions for issue groups 1, 3, and 5.
    - Added Zahid Ahmed's use cases for B2B transactions.
    - Added Maryann Hondo's use case scenario for ebXML.
    - Added comments to votes by Jeff Hodges, Bob Blakley.
- 10 Apr 2001 Made the following changes:
    - Added re-written versions of issue group 2, 3, 6, 7, 8, 9, 10, and 13 by Darren

  Platt and Evan Prodromou.

  - Added re-written versions of issue groups 11 and 12 by Irving Reid.
  - Added re-written version of issue group 4 by Prateek Mishra.
  - Added voting results for groups 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, and 13.

- 22 May 2001 Made the following changes:
    - Changed introduction to reflect conversion to general issues list
    - Added color scheme
    - Closed large number of issues per F2F #2
    - Changed OSSML to SAML everywhere
    - Added design issues section and groups 1-4
    - Added UC-13-07
    - Various minor edits
- 25 May 2001 Made the following changes
    - Various format improvements
    - Closed all Group 0 issues
    - Added DS-4-04
    - Did NOT promote blue issues to gray
- 11 June 2001 Made the following changes
  - Various format improvements, CLOSED in headers
  - Renumber Anonymity to DS-1-02 (was a duplicate)
  - Changed all Blue to Gray
  - Downgraded from Yellow to White UC-13-07, DS-1-01, DS-1-02, DS-4-02 (no recent discussion)
  - Closed DS-2-01 Wildcarded Resources
  - Added new text for DS-3-01, DS-3-02, DS-4-04

  - Added DS-2-02, Groups 5,6,7,8 and 9
- 18 June 2001 Made the following changes
  - Changed from Blue to Gray DS-2-01
  - Downgraded from Yellow to White UC-13-07, DS-2-02, DS-3-01, DS-3-02, DS-3-03, DS-6-01, DS-6-02, DS-6-03, DS-6-04, DS-7-01, DS-7-02, DS-7-03, DS-8-01, DS-8-02, DS-9-01
  - Created Miscellaneous Issues section, added MS-1-01 and MS-2-01
  - Created issue DS-10-01
  - Modified DS-4-01 & DS-4-03
- 9 August 2001 Made the following changes
  - Removed text and voting summaries from old, closed issues
  - Created issues DS-1-03, DS-1-04, DS-1-05, DS-4-05, DS-4-06, DS-4-07, DS-7-04, DS-7-05, DS-8-03, DS-8-04, DS-11-01 thru DS-11-05, DS-12-01 thru DS-12-05, DS-13-01, DS-14-01 thru DS-14-10, MS-3-01, MS-3-02
  - Modified DS-4-04, DS-8-02
  - Color changes to reflect recent discussions
- 22 August 2001 Made the following changes
  - Created issues: UC-14-01, DS-7-06, DS-9-02, DS-9-03, DS-12-06, DS-14-11, MS-4-01
- 16 January 2002 Made the following changes
  - Closed issues: DS-1-01, DS-1-05, DS-2-02, DS-4-01, DS-4-03, DS-4-06, DS-4-07, DS-5-02, DS-5-03, DS-6-02, DS-03, DS-7-01, DS-7-02, DS-8-02, DS-11-03, DS-11-05, DS-12-01, DS-12-02, DS-12-05, DS-14-01, DS-14-03, MS-1-01, MS-3-01, MS-3-02
  - Created issues: DS-1-06 thru DS-1-09, DS-4-08, DS-4-09, DS-6-05, DS-9-04 thru DS-9-10, DS-11-06, DS-14-12, DS-14-13, MS-4-02, MS-5-01 thru MS-5-03
  - Closed issues marked blue, new issues marked yellow