SIP
Internet-Draft
Expires: January 19, 2006

H. Tschofenig
Siemens
J. Peterson
NeuStar, Inc.
J. Polk
Cisco
D. Sicker
CU Boulder
M. Tegnander
LYIT

July 18, 2005

Using SAML for SIP
draft-tschofenig-sip-saml-04.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a mechanism for using the Security Assertion

Markup Language (SAML) in concert with the Session Initiation Protocol (SIP). In particular, it provides a way for SIP to refer to SAML objects, and for recipients of SIP messages to use SAML in order to make more informed authorization decisions.

Table of Contents

1. Introduction

This document proposes a method for using the Security Assertion Markup Language (SAML) in collaboration with SIP to accommodate richer authorization mechanisms and enable trait- based authorization where you are authenticated using roles or traits instead of identity. A motivation for trait based authorization and some scenarios are presented in [I-D.ietf-sipping-trait-authz].

Security Assertion Markup Language (SAML) [I-D.saml-tech-overview-1.1-03] is an XML extension for security information exchange that is being developed by OASIS. SAML is a XML-based framework for creating and exchanging security information.

To provide trait-based authorization a few solutions are possible: authorization certificates, SPKI or extensions to the authenticated identity body [I-D.ietf-sip-authid-body]. The authors selected SAML due to its increasing use in environments such as the Liberty Alliance and the Internet2 project, areas where the applicability to SIP is widely desired.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The SIP entity 'Authentication Service' was introduced with [I-D.ietf-sip-identity]. We reuse this term to refer to an entity that authenticates and authorizes a user and creates an assertion. This entity is the equivalent of the asserting party in the SAML terminology.

For terminology related to SAML the reader is referred to [I-D.saml-tech-overview-1.1-03].

3. Goals and Non-Goals

This document tries to accomplish the following goals:

- This document defines how SAML assertions are carried in the SIP. As such, the usage of SAML assertions within SIP can be seen as a SAML profile.

- The requirements and scenarios defined in [I-D.ietf-sipping-trait-authz] are compared to the solution described in this document by utilizing SAML assertions.

The following issues are outside the scope of this document:

- The configuration of the Authentication Service in order to attach certain assertions is outside the scope of this specification and might depend on the environment where SIP is used. To avoid restricting the functionality of SIP either as an in-band or an out-of-band mechanism, it can be defined to trigger the inclusion of SAML assertions. SAML itself provides mechanisms for this purpose.

- The attributes stored in assertions are, for example, roles, membership to a certain organization, specific access rights or information about the authentication. A definition of most of these attributes is application dependent and not defined in this document. The SAML specification itself provides a number of common attributes and provides extension points for future enhancements. A brief overview of the available attributes within an assertion is given in Section 4.1.

  In order for SAML to be used in a practical system, participating entities must agree on attributes and traits that will be used. Without this pre-existing agreement, SAML cannot be usefully deployed. This document does not discuss the manner in which participating entites might discover one another or agree on the syntax and semantics of attributes and traits.

- SIP is not used as a request/response protocol between the Relying Party and the Asserting Party to fetch an assertion based on a received artifact.

4. SAML Introduction

In SAML there are three main entities: the user, the asserting party and the relying party. A user requests assertions and receives them after a successful authentication and authorization protocol execution. The asserting party provides assurance that a particular user has been given proper authorization. The relying party has to trust the asserting party with regard to the provided information and then decides whether or not to accept the assertions provided, giving different levels of privileges.

The components of SAML are:

- Assertions/Artifact

- Request/Response protocols
- Bindings
- Profiles

We describe each in turn below

4.1 Assertions

An assertion is a package of information including authentication statements, attribute statements and authorization decision statements. All of statements do not have to be present, but at least one does. An assertion contains several elements:

Issuing information:

> Who issued the assertion, when was it issued and the assertion identifier.

Subject information:

> The name of the subject, the security domain and optional subject information, like public key.

Conditions under which the assertion is valid:

> Special kind of conditions like assertion validity period, audience restriction and target restriction.

Additional advice:

> Explaining how the assertion was made, for example.

In an authentication statement, an issuing authority asserts that a certain subject was authenticated by certain means at a certain time.

In an attribute statement, an issuing authority asserts that a

certain subject is associated with certain attributes which has certain values. For example, user jon@cs.example.com is associated with the attribute 'Department', which has the value 'Computer Science'.

In an authorization decision statement, a certain subject with a certain access type to a certain resource has given certain evidence that the identity is correct. Based on this, the relying party then makes the decision on giving access or not. The subject could be a human or a program, the resource could be a webpage or a web service, for example.

4.2 Artifact

The artifact used in the Browser/Artifact profile, is a base-64 encoded string that is 40 bytes long. 20 bytes consists of the typecode, which is the source id. The remaining 20 bytes consists of a random number that servers use to look up an assertion. The source server stores the assertion temporarily. The destination server receives the artifact and pulls the assertion from the source site. The purpose of the artifact is to act as a token that references an assertion for the subject who holds the artifact.

Note that artifacts were designed to be used specifically in a web context where the asserting party is clear due to the client/server nature of the protocol. Artifacts are not globally-derefenceable; one cannot tell simply be inspecting an artifact out of context which server generated the artifact. For the more peer-to-peer architecture of SIP, enhancements are required to make the context of artifact generation known to relying parties.

4.3 Request/Response Protocol

SAML defines a request/response protocol for obtaining assertions. The request asks for an assertion or makes queries for authentication, attribute and authorization decisions. The response carries back the requested assertion.

4.4 Bindings

The bindings in SAML maps between the SAML protocol and a transport and messaging protocol. With SAML Version 1.1 there is only one binding specified, which is SAML embedded in SOAP-over-HTTP. In a binding, a transport and messaging protocol is used only for transporting the request/response mechanism.

4.5 Profiles

When using a profile, SAML is used to provide assertions about a resource in the body of the message itself. In Version 1.1 of SAML, there are two profiles specified, the Browser/Artifact profile and the Browser/POST profile. The Browser/Artifact profile represents a "pull" model, where a special reference to the assertion called an artifact, is sent to the relying party from the asserting party. The artifact is then used to "pull" the assertion from the asserting party. The Browser/POST profile represents a "push" model, where an assertion is posted (using the HTTP POST command) directly to the relying party. These two models are described in Figure 2 and Figure 1.

5. Assertion Handling Models

As mentioned in Section 4.5, two main models can be used in SAML and therefore also with the SIP-SAML extension defined in this document: The Push and the Pull model.

A 'Push' model (or mode) means to transmit information towards another entity. Here we call that transmitting the information 'by-value'. An example of this model (or mode) is an email attachment (file). That attachment is included in the original transmission, as is 'by-value'.

Whereas, a 'Pull' model (or mode) means to transmit an identifier for where a piece of information is (located somewhere else). This piece of information still requires retrieval by the receiver of this transmission. Here we call that transmitting the information 'by-reference'. An example of this model (or mode) is including a URI in that email, to be accessed directly by the receiver of the email in a subsequent operation.

Either the end host or an intermediate proxy might request an assertion or an artifact. The Push and the Pull model used for HTTP does not match with its usage in SIP.

With the 'per-value' model either a user requests an assertion from the Asserting Party or the user triggers the Asserting Party to attach an assertion to the outgoing request. The assertion, which is added to the service request, can be verified by the Relying Party without additional protocol interactions with the Asserting Party. The assertion therefore contains enough information to authorize the service request.

Figure 1 shows the protocol exchange where the user fetches the assertion.

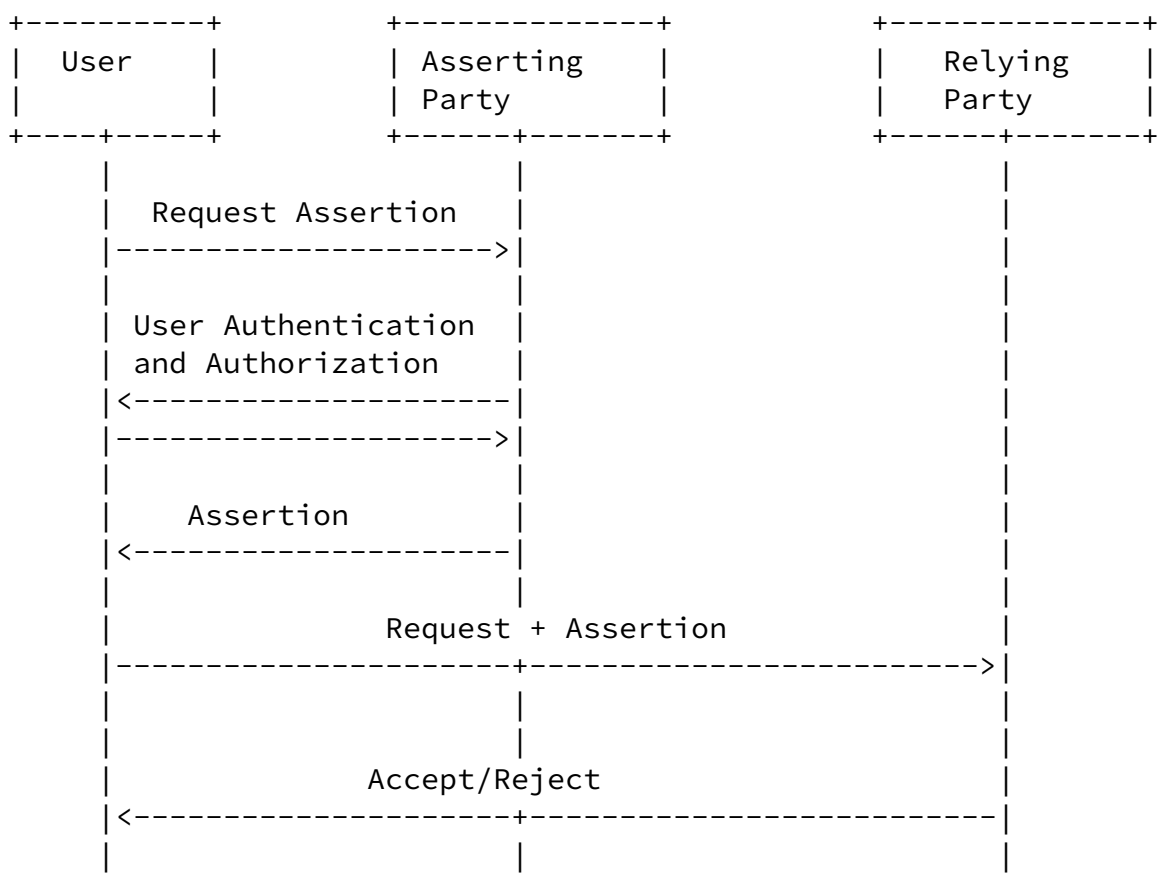


Figure 1: Example for a 'Per-value' Interaction

With the 'per-reference' model either the user contacts the Asserting Party to obtain an artifact or the user triggers the Asserting Party to attach the artifact to the outgoing request. The artifact is a reference to an assertion is stored at the Asserting Party and can later be dereferenced into the assertion on a request. The Relying Party later fetches the SAML assertion after receiving the artifact by the user. For communicating the SAML request and response messages, a separate message exchange is needed with a protocol, such as SOAP. A description of this protocol interaction is outside the scope of this document.

Figure 2 shows an example protocol interaction where the user fetches the artifact.

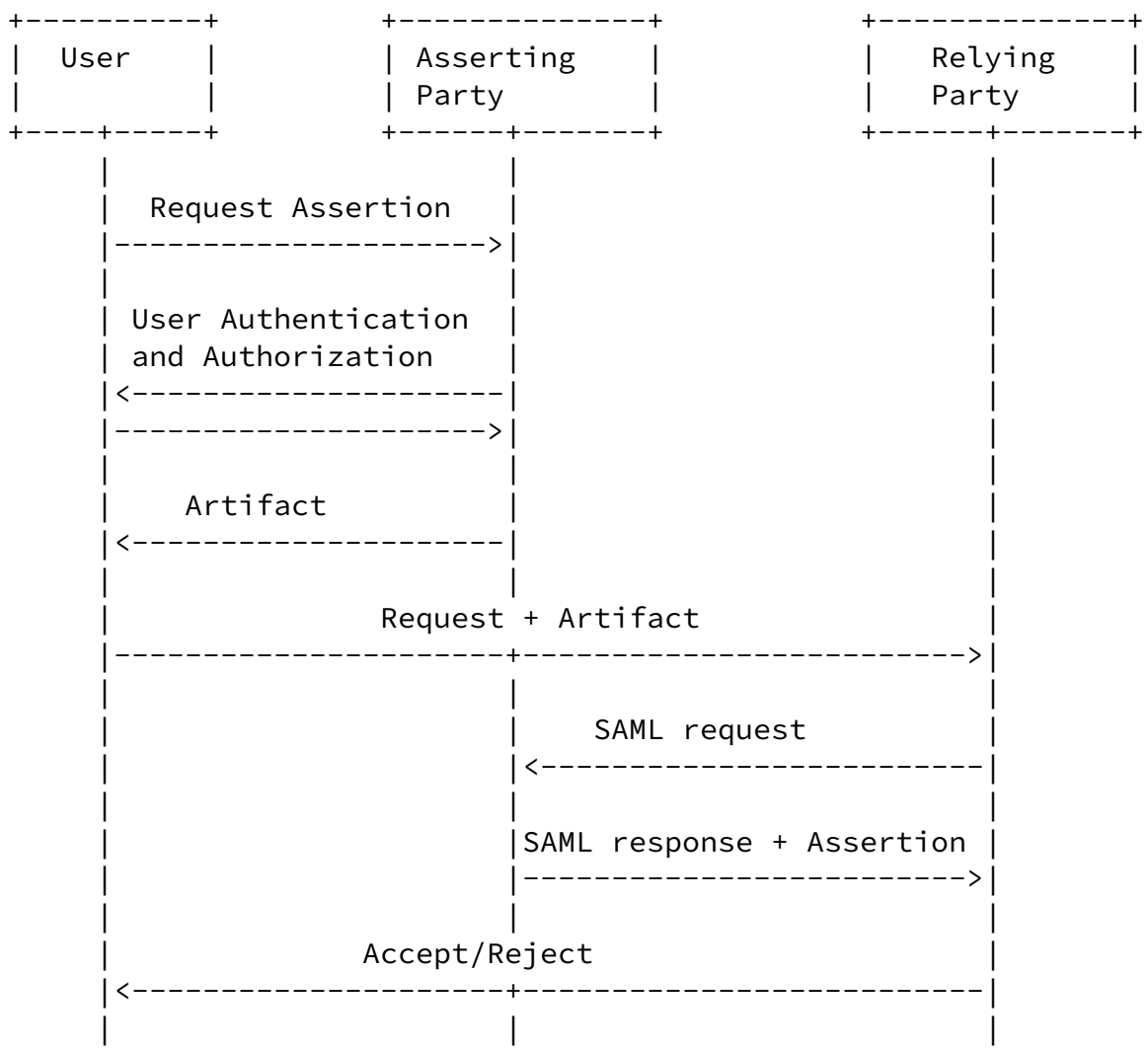


Figure 2: Example for a 'Per-reference' Interaction

The usage of SAML in HTTP-based environments and in SIP is, however,

affected by some architectural differences.

The user has to request an assertion at the Asserting Party and both entities mutually authenticate each other. The requested assertion is sent to the user in a confidential manner to prevent eavesdroppers from obtaining this assertion. The Relying Party trusts the Asserting Party. It is assumed that the accessed resource is located at the Relying Party and that this entity does not become malicious or act on behalf of the user to impersonate him or her to other parties with regard to access to his resources. To prevent some degree of misuse, attributes in the assertion limit its applicability for certain applications, servers or time frame.

Signaling in SIP can, however, involve a number of entities in more complex scenarios. As an example, the scenario addressed in [I-D.ietf-sip-identity] aims to replace end-to-end authentication via

S/MIME by a "mediated authentication architecture". The end hosts only need to be able to verify assertions signed by an Authentication Service which guarantees that the sender was authenticated.

Directly applying SAML to such a scenario, however, causes a problem: a SIP proxy, which securely receives a SAML assertion (such as confidentially protected to prevent eavesdroppers between the SIP UA and the SIP proxy to learn the assertion), can store this assertion to impersonate the user in future requests towards other SIP end users. The fact that multiple parties see the assertion along the path (i.e., SIP proxies) make the situation worse. The assertion might include some attributes which restrict its usage (such as recipient(s), unique identifier for the message or a time-based constraint) but they cannot fully prevent impersonation.

This problem appears if SAML assertions are not bound to a particular SIP transaction or dialog. Binding the assertion to a particular protocol session is not useful if the property of single-sign on is required.

To provide referential integrity the solution described in [I-D.ietf-sip-authid-body] can be reused. Such an approach allows a party in a SIP transaction to cryptographically sign the headers that assert the identity of the originator of a message, and provide some other headers necessary for reference integrity. An authenticated identity

body (AIB) is a MIME body of type 'message/sipfrag'. This MIME body has a Content-Disposition type of 'aib'. The MIME body is optional. The header fields From, Contact, Date and Call-ID must be used for providing identity. Contact and Date header fields are required for providing reference integrity. AIBs may contain other headers that help to uniquely identify the transaction or that provides related reference integrity.

The requirements for a non-INVITE AIB is very much the same as for an INVITE: From, Call-ID, Date and Contact header fields are required. The same that goes for requests also goes for responses with some small differences. The From header field of the AIB in the response to an INVITE must correspond to the address-of-record of the responder and not the From header field in the received request. The To header field of the request must not be included. A new Date header field has to be generated for the response while the Call-ID and CSeq are copied from the request.

Following is an example of the format of an AIB for an INVITE:


```
Content-Type: message/sipfrag
Content-Disposition: aib; handling=optional

From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example2.com>
Contact: <sip:alice@pc33.example.com>
Date: Thu, 26 Aug 2004 13:51:34 GMT
Call-ID: b76m5l94s90835
CSeq: 435431 INVITE
```

Figure 3: AIB Format for an INVITE

The same concept is applicable to this document as well with regard to reference integrity. For a further discussion on this topic see Section 14 and [I-D.peterson-message-identity].

6. Scenarios

This section shows message flows based on scenarios in [I-D.ietf-sipping-trait-authz] enriched with a SAML based solution. Section 6.1 provides an example of enhanced network asserted identities and Section 6.2 shows a SIP conferencing scenario with role-based access control using SAML. A future version of this document will cover more scenarios from [I-D.ietf-sipping-trait-authz].

6.1 Network Asserted Identities

Figure 4 shows an enhanced network asserted identity scenario based on [I-D.ietf-sip-identity] which again utilizes extensions proposed with [I-D.ietf-sip-authid-body]. The enhancement is based on the attributes asserted by the Authentication Service.

Figure 4 shows three entities, Alice@example.com, AS@example.com and Bob@example2.com. If Alice wants to communicate with Bob, she sends a SIP INVITE to her preferred AS. Depending on the chosen SIP security mechanism either digest authentication, S/MIME or Transport Layer Security is used to provide the AS with a strong assurance about the identity of Alice. During this step authorization attributes for inclusion into the SAML assertion can be selected.

After Alice is authenticated and authorized, a SAML assertion is attached to the SIP message. The Authentication Service can be configured to attach a number of assertions, not limited to the authenticated identity.

To bind the SAML assertion to a specific SIP session, it is necessary for the AS to compute a hash of some fields of the message. A list of the fields to hash is described in [I-D.ietf-sip-identity] and particularly in [I-D.ietf-sip-authid-body]. The hash is digitally signed and inserted into the SAML assertion and placed into the SAML header. The SAML header also contains information about the entity which created the digital signature. Upon reception of the message, Bob verifies the signature of the SAML assertion and verifies the binding to the SIP message in order to prevent cut-and-paste attacks. The provided SAML assertion can then be used to assist during the authorization procedure. If Bob does not understand the extension defined in this document, he silently ignores it. When the 200 OK message arrives at Bob's AS, the same procedure as when Alice sent her INVITE to her AS can be performed, if desired. This exchange is not shown in Figure 4.

Note that this scenario does not imply that the SAML assertions are solely used by SIP UAs. Assertions can also be helpful for SIP

proxies or B2B UAs.

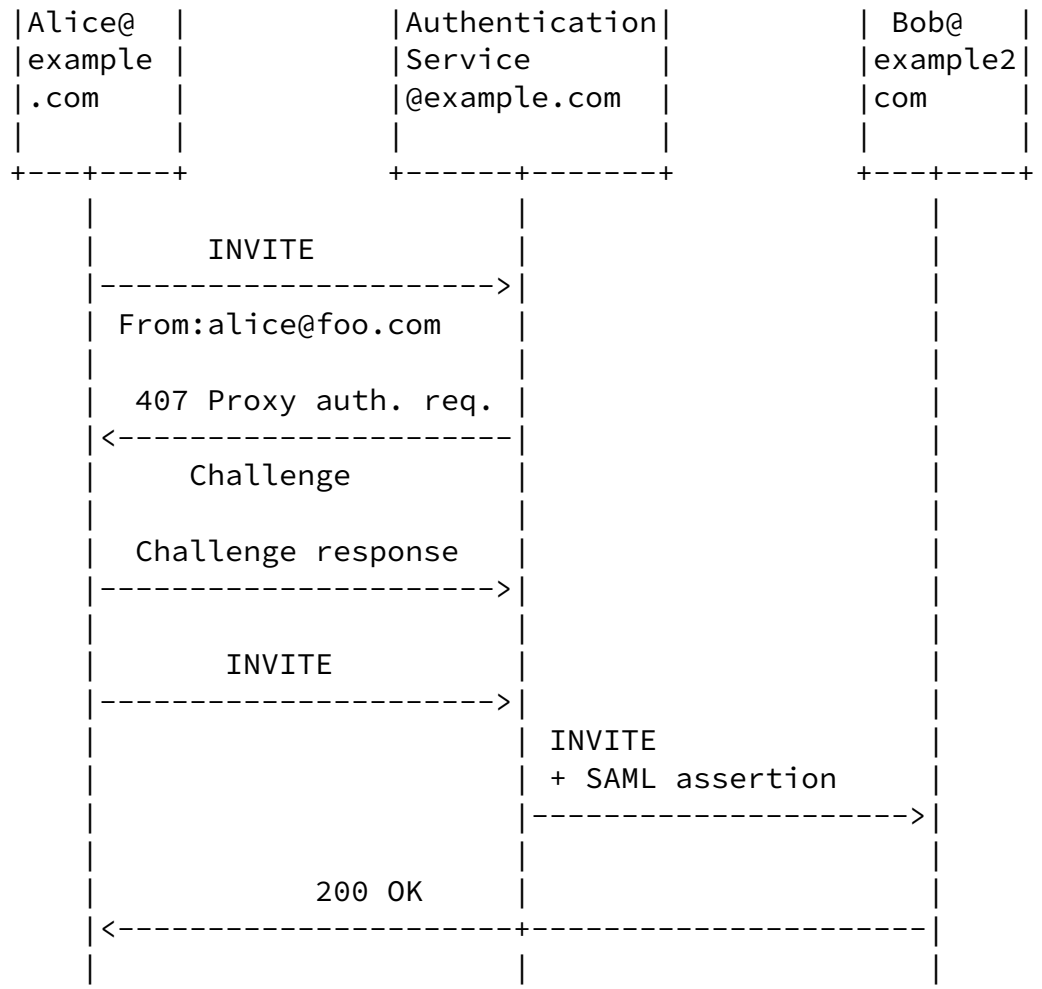


Figure 4: Network Asserted Identities

A variation of the scenario presented in Figure 4 is given in Figure 5 where an end host (Alice@example.com) obtains an assertion from its SIP proxy server which acts as an Authentication Service. This assertion is then attached by the end host to the outgoing INVITE message. Unlike in case of an artifact, Bob@example.com does not need to contact the Proxy Server.

An example of this scenario could be to preempt a lower priority call if enough assurance for doing so is presented in the attached SAML assertion. This would also mean that there is a priority value included in the INVITE (for example in the Resource-Priority Header).

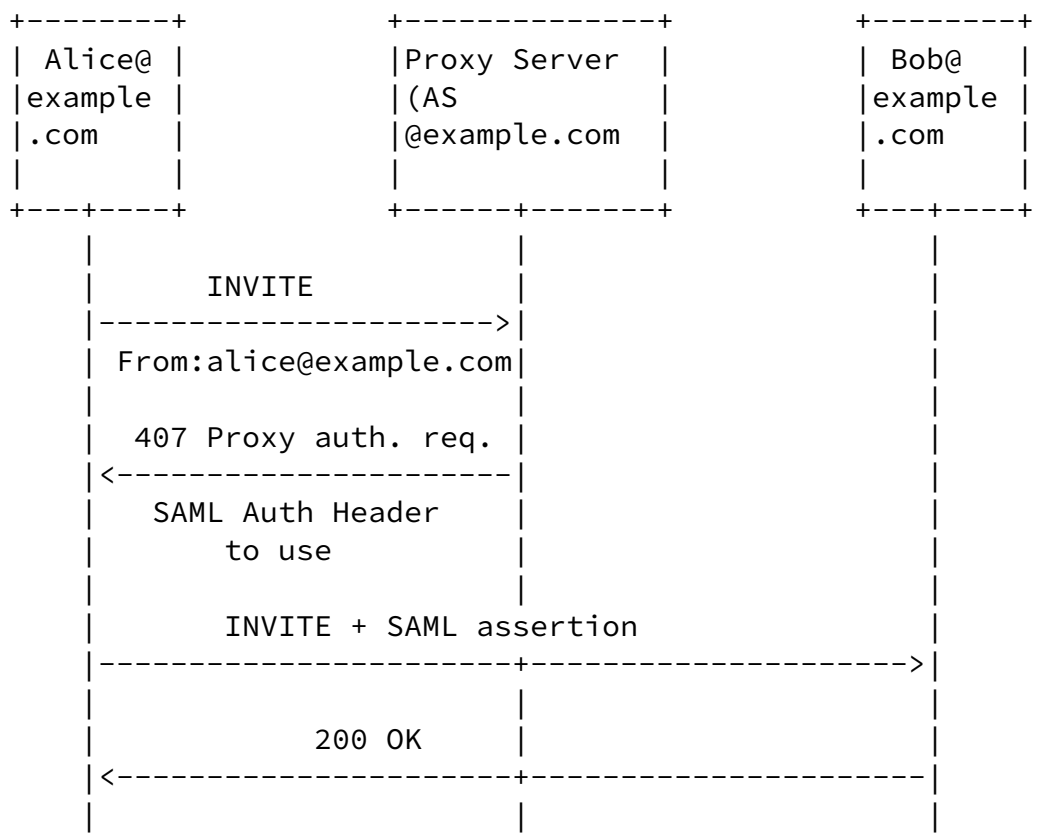


Figure 5: End host attaching SAML Assertion

Note that Bob and Alice do not need to be in the same administrative domain. It is feasible that Bob is in a domain that is federated with Alice's domain.

The assertion obtained by Alice@example.com needs to be associated with a particular SIP messaging session. How to achieve this binding is for further consideration.

6.2 SIP Conferencing

This section is meant to raise some discussions about the usage of SAML in the domain of conferencing. A user who routes its SIP message through the Authentication Service (Asserting Party) towards a conferencing server may want SAML assertions to be included. The following properties could be provided by this procedure:

- The user identity can be replaced to allow the user to be anonymous with regard to the Focus
- The user identity could be asserted to the Focus
- The SAML assertion could provide additional information such as

group membership (belongs to the students, staff, faculty group of

university X). This could, for non-identity-based authorization systems, imply certain rights.

The corresponding SIP message flow (in high level detail) could have the following shape:

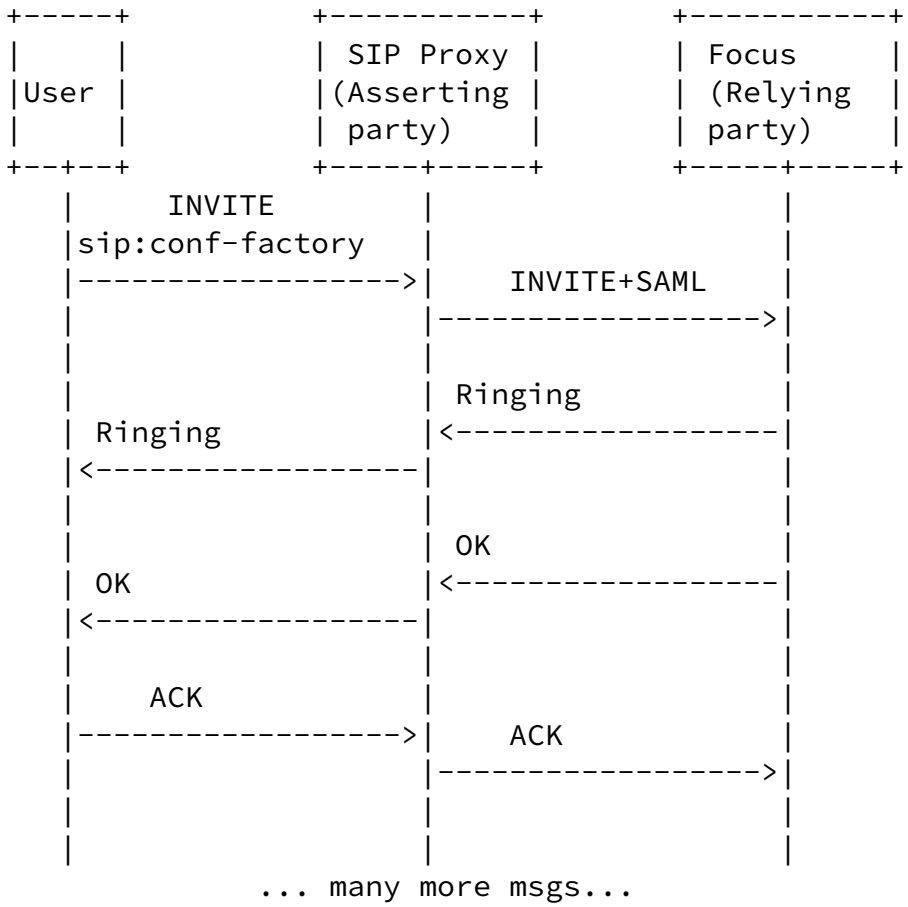


Figure 6: SIP Conferencing and SAML

6.3 PSTN-to-SIP Phone Call

Alice, using a phone connected to the PSTN, wants to make a call to Bob, which resides in a SIP network. Her call is switched through the PSTN by means of PSTN signaling (outside the scope of this document) to the PSTN/SIP gateway. At the gateway, the call is

converted from SS7 signaling to SIP signaling. Since Alice was previously 'authenticated' through PSTN signaling mechanisms, the gateway can create an assertion based on signaling information from Alice and digitally sign it with his private key. Alice's call is forwarded from the SIP/PSTN gateway to Bob's phone. Bob can certify that the identity of Alice is correct by examining the SAML assertion.

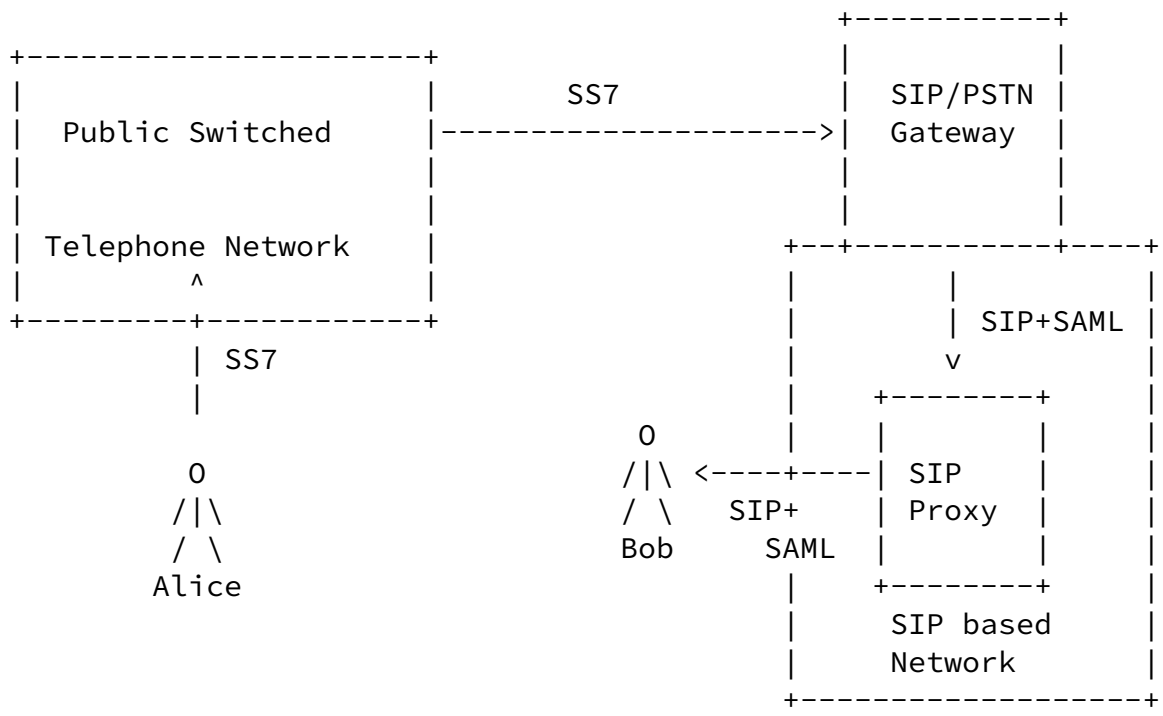


Figure 7: PSTN to SIP call

6.4 Compensation using SIP and SAML

This section briefly elaborates a scenario where SAML is used in SIP to realize compensation functionality as described in [I-D.jennings-sipping-pay]

Section 1 of [I-D.jennings-sipping-pay] shows a message exchange which is used by a number of payment protocols and hence can also be used by a SAML specified protocol. To avoid repetition in this document a second alternative is provided in Figure 8. Alice

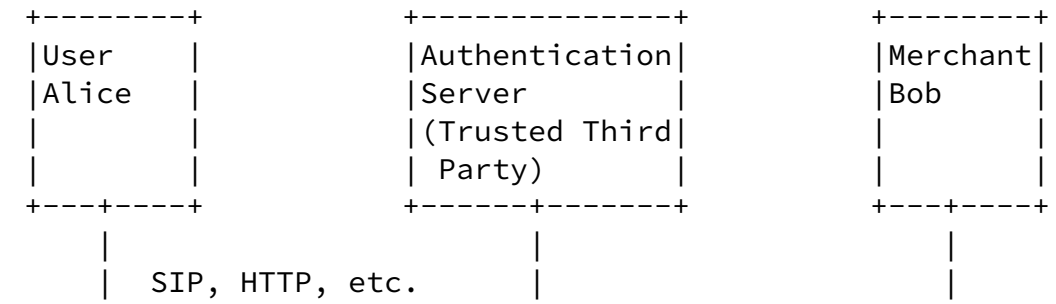
initiates a communication with an Authentication Service which acts as a financial institution. Note that Alice does not necessarily need to use SIP for communication with the Authentication Service. Instead, it might be possible to use HTTP or other protocols which offer the necessary user credential or offer additional information (such as a web page). After a successful authentication and authorization Alice obtains an assertion (or an artifact) which might contain payment relevant information. For a later service access, Alice contacts the merchant Bob with the assertion. Bob verifies the assertion and, it might want to contact the Authentication Service for a credit check. A financial settlement between the merchant Bob and the Trusted Third Party is assumed. Depending on the type of service, a one-time payment with immediate amount deduction may take place (e.g., in case of a prepaid account) or the amount is captured as a batch transaction.

The aspect of lightweight protocol execution is provided by

- The alternative usage of an artifact which leads to a lower bandwidth consumption.

- The ability to use a single assertion for multiple service access protocol executions, if desired.

- SAML, furthermore allows a cryptographic key to be bound to an assertion. A high degree of flexibility is provided with regard to the possible credentials. For example, it might not be necessary to use public key cryptography with every service access. This might be useful if the cost of public key cryptographic is too expensive for a cheap service or when devices have performance limitations. In this case, it might be useful to rely on symmetric cryptographic, such as hash chains.



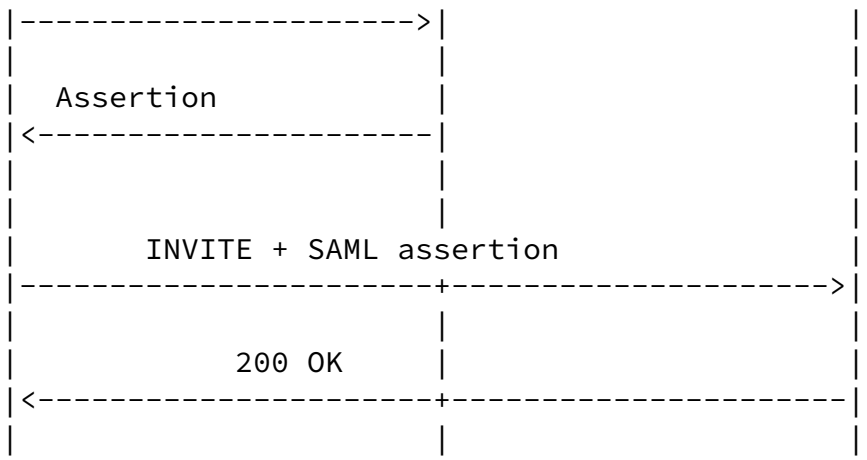


Figure 8: Message flow for SIP payment

The main difference between the above-described mechanism and the one described in Section 1 of [I-D.jennings-sipping-pay] is the degree of user involvement and the type of protocol interaction. In both cases it is possible to provide an indication to the user about the costs of a service access. In fact, the assertion might specify these type of constraints directly or indirectly with the help of recurring service requests/responses.

7. SIP-SAML Extension

To carry SAML assertions and artifacts two mechanisms are defined:

- The SIP header may either carry an Artifcat or a CID URI [RFC2392] pointing to an assertion in the SIP body. The name of this new SIP header is SAML-Assertion. A SAML artifact consists of a TypeCode, SourceID and an AssertionHandle. It is thereby assumed that the Relying Party will maintain a table of sourceID values as well as URLs for Asserting Parties to contact. This information is communicated out-of-band. This document also allows the Asserting Party to add a URL to point to the assertion to prevent this level of indirection.

- The SIP body may carry one or more SAML assertions. The MIME type of this SAML assertion is defined in [I-D.hodges-saml-mediatype].

A SIP user agent may add an assertion to the body of a SIP message or may add a reference to the assertion into the SIP header. SIP

proxies MUST NOT add the assertion to the body. The SIP header MUST be used instead when an assertion has to be added.

A SAML assertion SHOULD be protected and when added by a SIP entity then S/MIME MUST be used rather than XML digital signatures.

To bind a SAML assertion to a SIP message a few selected SIP message fields are input to a hash function. The digest-string, defined in Section 10 of [I-D.ietf-sip-identity], is included into the conditions extension point of the SAML assertion. Details are for further study.

8. Example

This is an example of a SAML assertion and how it is structured in XML.

```
<saml:Assertion
 xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
 MajorVersion="1"
 MinorVersion="1"
 AssertionID="P1YaAz/tP6U/fsw/xA+jax5TPxQ="
 Issuer="www.example.com"
```

```
 IssueInstant="2004-06-28T17:15:32.753Z">
       <saml:Conditions NotBefore="2004-06-28T17:10:32.753Z"
         NotOnOrAfter="2004-06-28T17:20:32.753Z" />
       <saml:AuthenticationStatement
      AuthenticationMethod="urn:ietf:rfc:3075"
      AuthenticationInstant="2004-06-28T17:15:12.706Z">
           <saml:Subject>
                   <saml:NameIdentifier>
                     NameQualifier=alice@example.com
                     Format="urn:oasis:names:tc:SAML:1.1:nameid-
                      format:emailAddress">uid=alice
                   </saml:NameIdentifier>
                   <saml:SubjectConfirmation>
                               <saml:ConfirmationMethod>
                                 urn:oasis:names:tc:SAML:1.0:
                                  cm:SIP-artifact-01
                               </saml:ConfirmationMethod>
                   </saml:SubjectConfirmation>
           </saml:Subject>
    </saml:AuthenticationStatement>
</saml:Assertion>
```

The elements in the assertion have the following meaning:

Tag name	Required	Description
MajorVersion	X	Major version of the assertion

MinorVersion	X	Minor version of the assertion
AssertionID	X	ID of the assertion
Issuer	X	The name of the SAML authority that created the assertion
IssuerInstant	X	Issuing time of the assertion
Conditions		Conditions that MUST be taken into account when assessing the validity of the assertion
AuthenticationMethod	X	Specifies what kind of authentication took place
AuthenticationInstant	X	Specifies the time when the authentication took place
Qualifier		The name by which the subject is recognized
Format		A URI reference representing the format of NameIdentifier
SubjectConfirmation		Specifies a subject by supplying data that allows the subject to be authenticated
ConfirmationMethod		Identifies which method to be used for authenticating the subject

Figure 10: Tag descriptions

9. Requirement Comparison

A future version of this document will compare the requirements listed in [I-D.ietf-sipping-trait-authz] with the solution provided in this document.

10. Security Considerations

This section discusses security considerations when using SAML with SIP.

10.1 Stolen Assertion

Threat:

If an eavesdropper can copy the real user's SAML response and included assertions and construct a SIP message of his own, then the eavesdropper could be able to impersonate the user at other SIP entities.

Countermeasures:

By providing adequate confidentiality, eavesdropping of a SAML assertion can be stopped.

10.2 MitM Attack

Threat:

Since the SAML assertion is carried within a SIP message, a malicious site could impersonate the user at some other SIP entities. These SIP entities would believe the adversary to be the subject of the assertion.

Countermeasures:

If the adversary is a not-participating in the SIP signaling itself (i.e., it is not a SIP proxy or a SIP UA), this threat can be eliminated by employing inherent SIP security mechanisms, such as TLS. However, if this entity is part of the communication itself then reference integrity needs to be provided. Assertions with tight restrictions (e.g., validity of the assertion) can also limit the possible damage.

10.3 Forged Assertion

Threat:

A malicious user could forge or alter a SAML assertion in order to communicate with the SIP entities.

Countermeasures:

To avoid this kind of attack, the entities must assure that proper mechanisms for protecting the SAML assertion needs to be in place. It is recommended to protect the assertion using a digital signature.

10.4 Replay Attack

Threat:

In the case of using SIP with a 'by-reference' model, the threat of replay lies in the fact that the artifact is a one-time-use subject. The same artifact can be used again to gain access to resources.

Countermeasures:

Cases where multiple requests are made which references the same request must be tracked in order to avoid the threat.

11. Contributors

The authors would like to thank Henning Schulzrinne for his contributions to this document.

12. Acknowledgments

We would like to thank RL 'Bob' Morgan and Stefan Goeman for their comments to this draft.

13. IANA Considerations

This document contains a number of IANA considerations. A future version of this document will list them in this section.

14. Open Issues

This document raises a number of issues with regard to the SIP protocol interaction. Some of them are raised in this document (such as reference integrity, who is allowed to add which information, etc.) but a more detailed treatment of this topic with a focus of identity management is described in [I-D.peterson-message-identity].

In particular, the following sections are highly relevant for this document:

Assertion Constraints and Scope:

This aspect deals with the problem of binding a SIP assertion to a specific SIP message. The goal is to provide a security property called reference integrity to prevent replay and impersonation attacks. As described in Section 5 that a number of fields can be used for this purpose. This document also considers scenarios where the SAML assertion was obtained outside a SIP protocol run. In these cases SIP fields are not available to provide reference integrity. The concept of the holder-of-the-key assertion is described below and relevant for this discussion.

Canonicalization versus Replication:

To provide reference integrity a few selected fields need to be hashed, signed and placed into the assertion. Two approaches are available for this purpose. Hence it needs to be studied how the interworking between reference integrity and the usage of obtained SAML assertion can be properly accomplished. For example, who indicates which fields are included?

Alignment with SIP Identity:

It might be good to avoid the definition of a second set of response codes for SAML conditions which will overlap with the response codes defined for SIP Identity draft.

Placement of Assertions and Keys in Messages:

This document assumes that the assertions are added to the SIP body and artifacts or references to assertions located in the SIP body are added to the SIP header. A central question is therefore where these assertions should be attached? Should the SIP user agent or intermediate SIP proxies add assertions/artifacts? In the scenarios depicted in Section 6, we have both approaches depending on what kind of scenario it is. In Figure 4, they are added at the UA and in contrast we have Figure 7, where the assertions are added at the PSTN/SIP gateway.

MIME bodies can only be attached at the UA. Proxies by definition do not attach MIME bodies; if an intermediary were to do so, it would not be playing the proxy server role in the SIP architecture. The SIP content indirection mechanism [I-D.ietf-sip-content-indirect-mech] is also relevant in this discussion.

To provide reference integrity (by binding a SIP session and a SAML assertion together) two alternative approaches exist:

Hashing of SIP message fields:

If a hash is computed over a number of selected SIP fields and subsequently digitally signed then there is a some degree of protection that the assertion cannot be copied to other SIP messages and reused. The drawback thereby is that the assertion has a very limited time period. The hashed fields may vary from context to context.

Holder-of-the-Key Assertion:

SAML introduces the concept of a holder-of-the-key assertion to bind the assertions (authorization information) to a cryptographic key. As a result, the end host which was quite passive when dealing with assertions can be turned into an active protocol participant. The end host obtained the assertion and attached them to selected messages but did not provide any cryptographic computations with regard to the assertion itself. With the end host being active in the protocol exchange security is improved a lot. Furthermore, it is possible to combine the benefits of the work done with SIPPING-CERT [I-D.ietf-sipping-certs] and this document by binding a self-signed certificate created by the user and stored at the credential server to an assertion. As noted in Section 9 of [I-D.ietf-sipping-certs] in the context of signing SIP messages the usage of a self-signed certificate is not very helpful except used with an Authentication Service. Combined with a SAML assertion the signature would protect the SIP message and the SAML assertion would provide authorization information.

A number of credentials can be used with the KeyInfo element of the Holder-of-the-Key assertion as described in Section 4.4 of [xmldsig-core].

Further open issues are:

- Some work on option-tags is required. Are there cases when processing of the assertion would be required by the sender? Or when a proxy server wants to be able to say that a UA must supply this header in order to access a particular resource? If so, an

option-tag should be defined for this extension that can be used in Require, Supported, 420, etc.

- Specific SAML confirmation method identifiers and identifiers for the bindings or profiles must be defined and registered with OASIS. A confirmation method identifier is a URI that specifies which method should be used by the target domain to assure that the identity of the subject is true.

  This mechanism seems to be provide the same reference integrity properties as the hash over the various headers/bodies proposed in the identity draft.

- Further use cases would be interesting. For example, a mechanism to provide additional security for the SIP REFER method [RFC3515].

- A few new URIs need to be registered. The proposed URIs for identification are:

  SIP Binding: urn:oasis:names:tc:SAML:1.0:bindings:SIP-binding

  Artifact
  profile: urn:oasis:names:tc:SAML:1.0:profiles:SIP-artifact-01

  Assertion
  profile: urn:oasis:names:tc:SAML:1.0:profiles:SIP-assertion-01

- The proposed URIs for Confirmation Method Identifiers are:

  Artifact profile: urn:oasis:names:tc:SAML:1.0:cm:SIP-artifact-01

  Assertion profile: urn:oasis:names:tc:SAML:1.0:cm:SIP-bearer

- These are based on the URIs already used in the existing SOAP-SAML binding, specified in Section 3 of [I-D.saml-bindings-1.1].

- An alignment with the work done by Liberty Alliance on Federated Identities as described in [I-D.liberty-idff-arch-overview] would be useful.

- The security consideration needs more details.

15. References

15.1 Normative References

[I-D.hodges-saml-mediatype]
Hodges, J., "application/saml+xml Media Type Registration", draft-hodges-saml-mediatype-01 (work in progress), June 2004.

[I-D.ietf-sipping-trait-authz]
Peterson, J., "Trait-based Authorization Requirements for the Session Initiation Protocol (SIP)", draft-ietf-sipping-trait-authz-01 (work in progress), February 2005.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC2392] Levinson, E., "Content-ID and Message-ID Uniform Resource Locators", RFC 2392, August 1998.

15.2 Informative References

[I-D.ietf-sip-authid-body]
Peterson, J., "SIP Authenticated Identity Body (AIB) Format", draft-ietf-sip-authid-body-03 (work in progress), May 2004.

[I-D.ietf-sip-content-indirect-mech]
Burger, E., "A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages", draft-ietf-sip-content-indirect-mech-05 (work in progress), October 2004.

[I-D.ietf-sip-identity]
Peterson, J. and C. Jennings, "Enhancements for

Authenticated Identity Management in the Session Initiation Protocol (SIP)", draft-ietf-sip-identity-05 (work in progress), May 2005.

[I-D.ietf-sipping-certs]
Jennings, C. and J. Peterson, "Certificate Management Service for The Session Initiation Protocol (SIP)", draft-ietf-sipping-certs-01 (work in progress), February 2005.

[I-D.jennings-sipping-pay]
Jennings, C., "Payment for Services in Session Initiation

Protocol (SIP)", draft-jennings-sipping-pay-01 (work in progress), February 2005.

[I-D.liberty-idff-arch-overview]
Wason, T., "Liberty ID-FF Architecture Overview", 2004.

[I-D.peterson-message-identity]
Peterson, J., "Security Considerations for Impersonation and Identity in Messaging Systems", draft-peterson-message-identity-00 (work in progress), October 2004.

[I-D.saml-bindings-1.1]
Maler, E., Philpott, R., and P. Mishra, "Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML) V1.1", September 2003.

[I-D.saml-core-1.1]
Maler, E., Philpott, R., and P. Mishra, "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1", September 2003.

[I-D.saml-sec-consider-1.1]
Maler, E. and R. Philpott, "Security and Privacy Considerations for the OASIS Security Markup Language (SAML) V1.1", September 2003.

[I-D.saml-tech-overview-1.1-03]
Maler, E. and J. Hughes, "Technical Overview of the OASIS

Security Assertion Markup Language (SAML) V1.1",
March 2004.

[RFC2543] Handley, M., Schulzrinne, H., Schooler, E., and J. Rosenberg, "SIP: Session Initiation Protocol", RFC 2543, March 1999.

[RFC3515] Sparks, R., "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003.

[xmldsig-core]
Eastlake, D., Reagle, J., and D. Solo, "XML-Signature Syntax and Processing, W3C Recommendation (available at http://www.w3.org/TR/xmldsig-core/)", February 2002.

Authors' Addresses

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Hannes.Tschofenig@siemens.com

Jon Peterson
NeuStar, Inc.
1800 Sutter St Suite 570
Concord, CA 94520
US

Email: jon.peterson@neustar.biz

James Polk
Cisco

2200 East President George Bush Turnpike
Richardson, Texas 75082
US

Email: jmpolk@cisco.com

Douglas C. Sicker
University of Colorado at Boulder
ECOT 430
Boulder, CO 80309
US

Email: douglas.sicker@colorado.edu

Marcus Tegnander
Letterkenny Institute of Technology
Port Road
Letterkenny, Donegal
Ireland

Email: schwed@gmail.com

Intellectual Property Statement

specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.