



# SAML X.509 Authentication-based Attribute Sharing Profile

OASIS Draft, 3 February 2005

**Document identifier:**

sstc-saml-x509-authn-based-attribute-protocol-profile-2.0-draft-02

**Location:**

[http://www.oasis-open.org/committees/documents.php?wg\\_abbrev=security](http://www.oasis-open.org/committees/documents.php?wg_abbrev=security)

**Editor:**

Rick Randall, Booz Allen Hamilton

**Contributors:**

Santosh Chokhani, Orion Security  
Rebekah Metz, Booz Allen Hamilton  
Robert Mingo, SAIC

**Abstract:**

This profile specifies the use of SAML attribute queries and assertions to support distributed authorization in support of X.509v3-based authentication.

**Status:**

This is a Draft.

Committee members should submit comments and potential errata to the [security-services@lists.oasis-open.org](mailto:security-services@lists.oasis-open.org) list. Others should submit them to the [security-services-comment@lists.oasis-open.org](mailto:security-services-comment@lists.oasis-open.org) list (to post, you must subscribe; to subscribe, send a message to [security-services-comment-request@lists.oasis-open.org](mailto:security-services-comment-request@lists.oasis-open.org) with "subscribe" in the body) or use other OASIS-supported means of submitting comments. The committee will publish vetted errata on the Security Services TC web page (<http://www.oasis-open.org/committees/security/>).

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

---

# 1 X.509 Authentication-based Attribute Sharing Profile

3 [SAMLCore] defines an Attribute Query/Response Protocol for retrieving a principal's attributes. This  
4 profile describes the use of this protocol with the SOAP binding defined in [SAMLBind], and additional  
5 guidelines for protecting the privacy of the principal with encryption, to support the retrieval of attributes of  
6 a principal authenticated using an X.509v3 certificate.

## 7 1.1 Required Information

8 **Identification:** urn:oasis:names:tc:SAML:2.0:profiles:x509authattributesharing

9 **Contact information:** [security-services-comment@lists.oasis-open.org](mailto:security-services-comment@lists.oasis-open.org)

10 **SAML Confirmation Method Identifiers:** This profile uses either of the SAML "bearer" or "holder-of-key"  
11 subject confirmation methods, identified by, respectively:

12 urn:oasis:names:tc:SAML:2.0:cm:bearer

13 urn:oasis:names:tc:SAML:2.0:cm:holder-of-key

14 **Description:** Given below.

15 **Updates:** NA

16 **Extends:** Attribute Query/Response Profile (Should we capture these relationships?)

## 18 1.2 Motivating Use Case

### 19 1.2.1 Overview

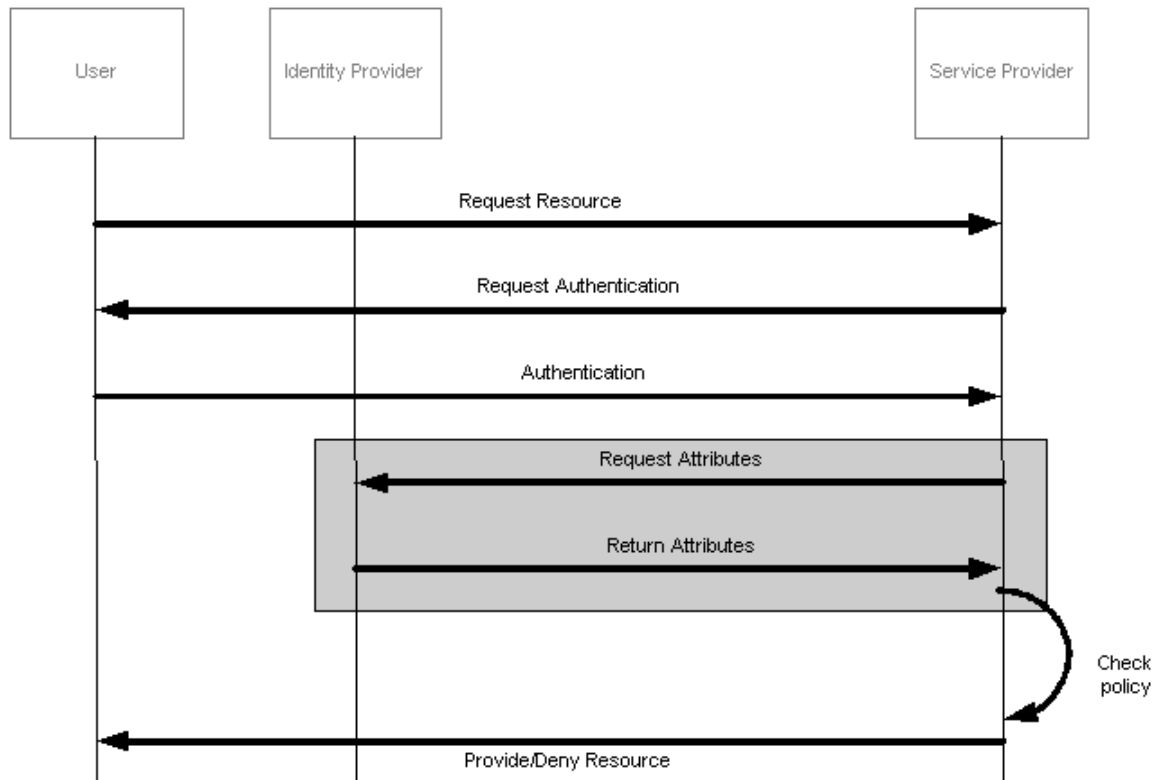
20 A principal attempts to access a web resource maintained at a service provider. Principal authentication is  
21 accomplished through the presentation of a trusted X.509v3 certificate (i.e. the federated credential is a  
22 certificate and not a SAML assertion) and by the demonstration of proof of possession of the associated  
23 private key.

24 Even after the principal has been authenticated, the service provider requires additional information about  
25 the principal in order to determine whether to grant access to some privileged resource(s). To get this  
26 information the service provider uses a synchronous 'back-channel' to query an identity provider for the  
27 required information about the principal. This is configured outside of SAML. When the identity provider  
28 returns to the service provider the relevant attributes, the service provider is now able to make an  
29 informed authorization decision.

### 30 1.2.2 Sequence

31 The sequence of steps for the full use case is shown below.

32 **Note:** those steps constrained by this profile are hilted with a gray box - the other steps are shown only for  
33 completeness, the profile does not constrain them.



2

3 **1. HTTP Request to Service Provider**

4 In step 1, the principal, via an HTTP User Agent, makes an HTTP request for a secured resource  
 5 at the service provider without a security context.

6 **2. Service Provider requests Authentication**

7 In step 2, the service provider requests that the principal be authenticated.

8 **3. Authentication**

9 In step 3, the principal authenticates to the service provider with an X.509v3 certificate. The  
 10 service provider will authenticate to the principal at the same time (i.e., SSL mutual authentication  
 11 MUST be performed).

12 **4. Request Attributes**

13 In step 4, the service provider sends a SAML `<AttributeQuery>` to the identity provider over  
 14 the SAML SOAP Binding, using the DistinguishedName (DN) (in encrypted format) from the  
 15 principal's X.509v3 certificate (presented in step 3 above) within the `<Subject>` element. The  
 16 service provider will sign the attribute request such that the identity provider will be able to  
 17 determine its origin and integrity.

18 The location to which the service provider sends the `<AttributeQuery>` is determined by a  
 19 service provider configuration setting.

20 **5. Return Attributes**

21 In step 5, after verifying that the service provider is a valid requester, the identity provider issues a  
 22 `<Response>` message containing appropriate attributes (in encrypted format) for the principal.  
 23 The identity provider will sign the response such that the service provider will be able to determine

1 its origin and integrity. The <Subject> element in response MUST contain a  
2 <ConfirmationMethod> of either 'bearer' or 'holder-of-key' within the  
3 <SubjectConfirmation>. If 'holder-of-key', the identity provider MUST populate the  
4 <SubjectConfirmationData> with a <KeyInfo> element that will allow the principal to  
5 subsequently confirm themselves to the service provider.

## 6 6. Confirm Subject

7 In this step, the service provider verifies whether the <KeyInfo> matches the public key used by  
8 the service provider to authenticate the principal. If the public key matches exactly, proceed to  
9 step 7. If the public key does not match exactly, the service provider MUST reconfirm the  
10 principal using <KeyInfo> prior to granting access to the resource.

## 11 7. Return resource

12 Based on the results of steps 5 and 6, having received the response from the identity provider, the  
13 service provider evaluates the principal's user agent with an error, or can return the requested  
14 resource.

15  
16 Of the sequence steps described above, it is steps 4 and 5 that are profiled below.  
17

## 18 1.3 Profile Description

19 In this profile, a service provider uses the SAML SOAP Binding to send an <AttributeQuery> message  
20 directly to an identity provider. This message contains an encrypted name identifier assigned to a  
21 principal that authenticated to the service provider using an X.509v3 certificate.

22 The service provider MUST authenticate to the identity provider by signing the <AttributeQuery>  
23 message. In addition, the requester MAY use TLS or SSL client authentication.

24 If the identity provider receiving the request can

25 (1) Decrypt and recognize the name identifier

26 (2) Fulfill the request based on authentication of the requester and any applicable policies

27 it will respond with a successful <Response> containing the relevant attributes for the identified principal.  
28 The returned attributes MUST be encrypted as described below.

29 The responding identity provider MUST authenticate to the requester, both by signing the <Response>  
30 message and through TLS or SSL server authentication.

### 31 1.3.1 <AttributeQuery> Issued by Service provider to Identity Provider

32 The identity provider MUST process the <AttributeQuery> message and any enclosed <Assertion>  
33 elements as described in [SAMLCore] and in the [Attribute Request/Response Profile].

34 All HTTP requests/responses MUST be made over either SSL 3.0 ([SSL3]) or TLS 1.0 ([RFC2246]) to  
35 maintain confidentiality and message integrity.

#### 36 1.3.1.1 <AttributeQuery> Usage

37 The <AttributeQuery> MUST conform to the following:

- 38 • The <Subject> element must contain an <EncryptedID> element carrying the encrypted  
39 value of the principal's DN. See section for details on the use of encryption.

- 1           • It MUST contain a <Signature> element carrying the signature of the service provider over  
2           the request.

### 3   **1.3.1.2 Use of Encryption**

4   [SAMLCore] defines the <EncryptedID> element as a means of applying confidentiality to a name  
5   identifier.

6   In this profile the service provider MUST use the <EncryptedID> to carry the DN of the principal in the  
7   <AttributeQuery>.

8   The service provider MAY use a previously established symmetric key to encrypt the principal's DN. If the  
9   service provider reuses a symmetric key to encrypt the DN, the resulting ciphertext is placed in the  
10  <xenc:EncryptedData> element and the <EncryptedID> element MUST NOT contain an  
11  <xenc:EncryptedKey> element.

12  Alternatively, the service provider MAY generate a new symmetric key for encrypting the principal's DN.  
13  The service provider then encrypts the DN with that key, and places the resulting ciphertext in the  
14  <xenc:EncryptedData> element. The symmetric key MUST be encrypted with the identity provider's  
15  public key and the resulting ciphertext placed in the <xenc:EncryptedKey> element.

16  Encryption MUST be performed before the digital signature operation.

### 17  **1.3.1.3 Use of Digital Signatures**

18  [SAMLCore] provides the <Signature> element as a mean of providing integrity and authenticity for a  
19  message.

20  In this profile, a service provider MUST sign the <AttributeQuery> containing the <EncryptedID> to  
21  allow the identity provider to verify both its origin and that it was not modified in transit.

## 22  **1.3.2 <Response> issued by Identity Provider to Service Provider**

23  The service provider MUST process the <Response> message and any enclosed <Assertion>  
24  elements as described in the [Attribute Request/Response Profile].

25  All HTTP requests/responses MUST be made over either SSL 3.0 ([SSL3]) or TLS 1.0 ([RFC2246]) to  
26  maintain confidentiality and message integrity.

### 27  **1.3.2.1 <Response> Usage**

28  If the identity provider wishes to return an error, it MUST NOT include any assertions in the <Response>  
29  message. Otherwise, if the request is successful the <Response> element MUST conform to the  
30  following:

- 31           • It MUST contain exactly one <EncryptedAssertion> element.
- 32           • The encrypted content of the <EncryptedAssertion> is an <Assertion> element that  
33           MUST satisfy the following conditions:
- 34               • The <Subject> element MUST contain a <ConfirmationMethod> of either  
35                'bearer' or 'holder-of-key' within the <SubjectConfirmation>. If 'holder-of-key', the  
36                identity provider MUST populate the <SubjectConfirmationData> with a  
37                <KeyInfo> element that will allow the principal to subsequently confirm themselves to  
38                the service provider. The identity provider MAY use the public key from within the  
39                principal's X.509v3 certificate for this purpose.
  - 40               • It MUST contain exactly one <AttributeStatement> that reflects the attributes of

- 1 the principal to the service provider.
- 2 • The <Assertion> MUST contain a <Signature> element carrying the signature of  
3 the identity provider.
  - 4 • The <Assertion> MUST contain an <AudienceRestrictionCondition>  
5 including the service provider's unique identifier as an <Audience>.
  - 6 • Other conditions (and other <Audience> elements) MAY be included as requested by  
7 the service provider or at the discretion of the identity provider. Any additional  
8 conditions MUST be understood by and accepted by the service provider in order for  
9 the assertion to be considered valid.

### 10 **1.3.2.2 Use of Digital Signatures**

11 [SAMLCore] provides the <Signature> element as a mean of providing integrity and authenticity for a  
12 message.

13 In this profile, the identity provider MUST sign the <Assertion> in order to allow the service provider to  
14 verify both its origin and that it was not modified in transit.

15 The signature is calculated before the encryption operation.

### 16 **1.3.2.3 Use of Encryption**

17 [SAMLCore] defines the <EncryptedAssertion> element as a mean of applying confidentiality to the  
18 contents of an assertion.

19 In this profile the identity provider MUST use the <EncryptedAssertion> element to carry the returned  
20 attribute values for the principal.

21 The identity provider MAY use the symmetric key used for encrypting the principal's DN on the initial  
22 <AttributeQuery> in order to encrypt the returned <Assertion> . If the identity provider reuses a key  
23 in this manner, the <EncryptedAssertion> element MUST NOT contain an <xenc:Encryptedkey>  
24 element.

25 Alternatively, the identity provider MAY generate a symmetric key for encrypting the assertion carrying the  
26 attributes and place the resulting ciphertext in the <xenc:EncryptedData> element. The symmetric key  
27 used to encrypt the assertion MUST be encrypted with the service provider's public key and the resulting  
28 ciphertext placed in the <xenc:EncryptedKey> element within the <EncryptedAssertion> element.

29

## 30 **1.4 Implementation Guidance (informative)**

31 The following guidance is informatively provided for implementers.

### 32 **1.4.1.1 Identify Provider Policy**

33 The motivation for this profile is to specify a secure means of using X.509 authentication in association  
34 with SAML attributes. As such, security considerations are highly important from the perspective of the  
35 profile. The policy configuration of Identity Providers SHOULD permit only a strictly limited list of attribute  
36 responses in SAML assertions. Wildcard-like <AttributeQuery> requests SHOULD be prohibited by  
37 policy, and rejected by Identity Providers.

38

### 1 **1.4.1.2 Caching of Attributes**

2 A capability to cache user attributes that are returned in assertions SHOULD be provided. Cache  
3 expiration settings SHOULD be configurable by administrators. The identity of the user for which the  
4 assertion was issued SHOULD NOT be human readable (i.e., clear text) in cache files. Attributes  
5 SHOULD NOT be cached for a period that extends beyond their lifetime.

6

---

## 2 References

- 1
- 2
- 3 **[SAMLBind]** Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0, DRAFT
- 4 **[SAMLCore]** S. Cantor et al. *Assertions and Protocol for the OASIS Security Assertion Markup Lan-*
- 5 *guage (SAML)*. OASIS,. Document ID sstc-saml-core-2.0-cd-03c.
- 6 <http://www.oasis-open.org/committees/security/>.
- 7 **[SAMLProfiles]** E. Maler et al. *Assertions and Protocol for the OASIS Security Assertion Markup*
- 8 *Language (SAML)*. OASIS, September 2003. Document ID sstc-saml-profiles-2.0-cd-03c.
- 9 <http://www.oasis-open.org/committees/security/>.
- 10 **[SSL3]** A. Frier et al., *The SSL 3.0 Protocol*, Netscape Communications Corp, November 1996.
- 11 **[RFC2246]** The TLS Protocol Version 1.0, <http://www.ietf.org/rfc/rfc2246.txt>.
- 12
- 13

---

## 1 A. Acknowledgments

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

- 4 Conor P. Cahill, AOL
- 5 John Hughes, Atos Origin
- 6 Hal Lockhart, BEA Systems
- 7 Michael Beach, Boeing
- 8 Rebekah Metz, Booz Allen Hamilton
- 9 Tim Alsop, CyberSafe Limited
- 10 Paul Madsen, Entrust
- 11 Irving Reid, Hewlett-Packard
- 12 Paula Austel, IBM
- 13 Maryann Hondo, IBM
- 14 Michael McIntosh, IBM
- 15 Tony Nadalin, IBM
- 16 Nick Ragouzis, Individual
- 17 Scott Cantor, Internet2
- 18 RL 'Bob' Morgan, Internet2
- 19 Peter C Davis, Neustar
- 20 Jeff Hodges, Neustar
- 21 Frederick Hirsch, Nokia
- 22 John Kemp, Nokia
- 23 Charles Knouse, Oblix
- 24 Steve Anderson, OpenNetwork
- 25 Prateek Mishra, Principal Identity
- 26 John Linn, RSA Security
- 27 Rob Philpott, RSA Security
- 28 Jahan Moreh, Sigaba
- 29

30

---

## 1 B. Notices

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

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

13 **Copyright © OASIS Open 2003-2004. All Rights Reserved.**

14 This document and translations of it may be copied and furnished to others, and derivative works that  
15 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and  
16 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and  
17 this paragraph are included on all such copies and derivative works. However, this document itself may  
18 not be modified in any way, such as by removing the copyright notice or references to OASIS, except as  
19 needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights  
20 defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it  
21 into languages other than English.

22 The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors  
23 or assigns.

24 This document and the information contained herein is provided on an "AS IS" basis and OASIS  
25 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY  
26 WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR  
27 ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

28 JavaScript is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and  
29 other countries.