XML Digital Signature profile of XACML

Committee Draft 01, 16 September 2004

Document identifier:
access_control-xacml-2.0-dsig_profile-spec-cd-01

Location:
http://docs.oasis-open.org/xacml/access_control-xacml-2.0-dsig_profile-spec-cd-01.pdf

Editor:
Anne Anderson, Sun Microsystems <anne.anderson@sun.com>

Abstract:
This specification profiles use of the W3C XML-Signature Syntax and Processing Standard in providing authentication and integrity protection for XACML schema instances.

Status:
This version of the specification is an approved Committee Draft within the OASIS Access Control TC.

Access Control TC members should send comments on this specification to the xacml@lists.oasis-open.org list. Others may use the following link and complete the comment form: http://oasis-open.org/committees/comments/form.php?wg_abbrev=xacml.

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 section of the Access Control TC web page (http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml).

For any errata page for this specification, please refer to the Access Control TC web page (http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml).

Table of Contents

1 Introduction (non-normative)

This document provides a profile for use of the W3C XML-Signature Syntax and Processing Standard in providing authentication and integrity protection for OASIS eXtensible Access Control Markup Language [XACML] schema instances. Sections 9.2.1 Authentication and 9.2.4 Policy integrity in [XACML] describe requirements and considerations for such authentication and integrity protection.

A digital signature is useful for authentication and integrity protection only if the signed information includes a specification of the identity of the signer and a specification of the period during which the signed data object is to be considered valid. XACML itself does not define the format for such information, as XACML is intended to use other standards for functions other than the actual specification and evaluation of access control policies, requests, and responses.

One appropriate format that has been defined elsewhere is [SAML].. A profile for the use of SAML with XACML schema instances is available in [XACML-SAML]. This profile therefore RECOMMENDS use of XACML schema instances in SAML Assertions, Requests, and Responses, which MAY then be digitally signed as specified in the SAML specification.

This profile also notes various canonicalization issues that must be resolved in order for signed documents to be verified by a relying party.

This profile specification assumes that the reader is familiar with the concept of a digital signature, with the W3C XML-Signature Syntax and Processing Standard, and with XACML.

1.1 Terminology

(This section is not normative.)

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

data object – used in this profile to refer to a digital object that is being signed. A data object could be an XACML PolicySet, Policy, Request context, Response context, or any associated schemas. A data object is referenced inside an [XMLDSIG] <Reference> element using a URI as defined by [RFC2396].

2 XML Digital Signature profile of XACML

2.1 Use of SAML

(This section is normative)

This Profile RECOMMENDS use of XACML schema instances embedded in SAML Assertions, Requests, and Responses as described in [XACML-SAML]. Such SAML objects SHALL be digitally signed as described in *Section 5: SAML and XML Signature Syntax and Processing* of [SAML].

2.2 Canonicalization

In order for a digital signature to be verified by a relying party, the byte stream that was signed MUST be identical to the byte stream that is verified. To ensure this, the XML document being signed MUST be *canonicalized*. *Section 5: SAML and XML Signature Syntax and Processing* of [SAML] specifies use of Exclusive Canonicalization [ExclC14N].

2.2.1 Namespace elements in XACML data objects

Any XACML **data object** that is to be signed MUST specify all namespace elements used in the **data object**. If this is not done, then the **data object** will attract namespace definitions from ancestors of the **data object** that may differ from one envelope to another.

When [ExclC14N] is used as the **canonicalization** or transform method, then the namespace of XACML schemas used by elements in an XACML **data object** MUST be bound to prefixes and included in the *InclusiveNamespacesPrefixList* parameter to [ExclC14N].

2.2.2 Additional canonicalization considerations

Additional transformations on the XACML data object must usually be performed in order to ensure that the data object signed will match the data object that is verified. Some of these transformations are listed here, but this Profile does not attempt to specify algorithms for performing these.

If an XACML **data object** includes data elements that may be represented in more than one form (such as (TRUE, FALSE), (1,0), (true,false)), then a Transform method MUST be defined and specified for normalizing those data elements.

This Profile RECOMMENDS applying the following canonicalizations to values of the corresponding datatypes, whether occurring in XML attribute values or in XACML Attributes.

1. Where a canonical representation for an XACML-defined datatype is defined in http://www.w3.org/2001/XMLSchema, then the value of the datatype MUST be put into the canonical form specified in http://www.w3.org/2001/XMLSchema. This includes boolean {“true”, “false”}, double, dateTime, time, date, and hexBinary (upper-case).

2. http://www.w3.org/2001/XMLSchema#anyURI - use the canonical form defined in [RFC2396]

3. http://www.w3.org/2001/XMLSchema#base64Binary - remove all line breaks and white space. Remove all characters following the first sequence of "=" characters. The *Base64 Transform* (identifier: http://www.w3.org/TR/xmldsig-core/#sec-Base-64) MAY be useful in performing this canonicalization.

4. urn:oasis:names:tc:xacml:1.0:data-type:x500Name - first normalize according to [RFC2253]. If any RDN contains multiple attributeTypeAndValue pairs, re-order the AttributeValuePairs in that RDN in ascending order when compared as octet strings (described in Section 11.6 "Set-of components" of [X.690]).

5. urn:oasis:names:tc:xacml:1.0:data-type:rfc822Name - normalize the domain-part of the name to lower case.

6. XPath expression – apply [XPath2Filt] to put the XPath expression into canonical form.

Schema Centric XML Canonicalization Version 1.0 [ScC14N] describes many canonicalization issues for XML documents that should be addressed.

2.3 Signing schemas

The parsing of any XACML **data object** depends on having an accurate copy of all schemas on which the XACML **data object** depends. Note that the inclusion of a schema URI in the XACML schema instance attributes does not guarantee that an accurate copy of the schema will be used: an attacker may substitute a bogus schema that contains the correct identifier. **Signatures** can help protect against substitution or modification of the schemas on which an XACML **data object** depends. Use of **signatures** for this purpose are described in this section.

In most cases, a **data object** signer SHOULD include a <Reference> element for each schema on which the XACML **data object** depends in the <SignedInfo> element that contains the <Reference> to or including the XACML **data object** itself.

In some cases, the **data object** signer knows that all PDPs that will evaluate a given XACML **data object** will have accurate copies of certain schemas needed to parse the **data object**, and does not want to force the PDP to verify the message digest for such schemas. In these cases the **data object** signer MAY omit <Reference> elements for any schema whose verification is not needed.

3 References .

[ExclC14N] J. Boyer et al., *Exclusive Canonicalization Version 1.0*, 18 January 2002, World Wide Web Consortium, http://www.w3.org/TR/xml-exc-c14n/.

[RFC2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, IETF RFC 2119, March 1997, http://www.ietf.org/rfc/rfc2119.txt.

[RFC2253] M. Wahl, et al., *Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names*, IETF RFC 2253, September 1997, http://www.ietf.org/rfc/rfc2253.txt.

[RFC2396] T. Berners-Lee, et al., *Uniform Resource Identifiers (URI): Generic Syntax,* August 1998, http://ftp://ftp.isi.edu/in-notes/rfc2396.txt.

[SAML] S. Cantor, et al., eds., *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML} V2.0,* Committee Draft 01c, 18 September 2004, http://www.oasis-open.org/committees/documents.php?wg_abbrev=security.

[ScC14N] S. Aissi, M. Hondo, eds., *Schema Centric XML Canonicalization, Version 1.0*, 20 May 2003, http://uddi.org/pubs/SchemaCentricCanonicalization.htm.

[XACML] S. Godik, T. Moses, eds., *OASIS eXtensible Access Control Markup Language (XACML) Version 2.0*, Committee Draft 01, 16 September 2004, http://docs.oasis-open.org/xacml/access_control-xacml-2.0-core-spec-cd-01.pdf.

[XACML-SAML] A. Anderson, H. Lockhart, eds., *SAML 2.0 profile of XACML*, Committee Draft 01, 16 September 2004, http://docs.oasis-open.org/xacml/access_control-xacml-2.0-saml_profile-spec-cd-01.pdf.

[**XMLDSIG]** D. Eastlake, et al., *W3C XML-Signature Syntax and Processing*, W3C Recommendation, 12 February 2002, http://www.w3.org/TR/xmldsig-core.

[XPath2Filt] J. Boyer, M. Hughes, J. Reagle, editors, *XML-Signature XPath Filter 2.0*, 8 November 2002 http://www.w3.org/TR/xmldsig-filter2/.

[X.690] ITU-T Recommendation X.690 Information Technology – Open Systems Interconnection - Procedures for the operation of OSI Registration Authorities: General procedures, 1992.

Appendix A. Acknowledgments

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

- Frank Siebenlist, Argonne National Laboratory
- Daniel Engovatov, BEA Systems, Inc.
- Hal Lockhart, BEA Systems, Inc.
- Rebekah Metz, Booz Allen Hamilton
- Ronald Jacobson, Computer Associates
- Tim Moses, Entrust
- Simon Godik, GlueCode Software
- Bill Parducci, GlueCode Software
- Michiharu Kudo, IBM
- Michael McIntosh, IBM
- Anthony Nadalin, IBM
- Steve Anderson, OpenNetwork
- Anne Anderson, Sun Microsystems
- Seth Proctor, Sun Microsystems
- Polar Humenn, Syracuse University
- Edward Coyne, Veterans Health Administration

Appendix B. Revision History

Rev	Date	By Whom	What
CD-01	16 Sept 2004	XACML committee	Committee Draft

Appendix C.Notices

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

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