Multiple Resource profile of XACML

Committee Draft 01, 30 September 2004

Document identifier:

access_control-xacml-2.0-mult_profile-spec-cd-01

Location:

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

Editor:

Anne Anderson, Sun Microsystems (anne.anderson@sun.com)

Abstract:

This document provides a profile for requesting access to more than one resource in a single XACML Request Context, or for requesting a single response to a request for an entire hierarchy.

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).

Copyright © OASIS Open 2004 All Rights Reserved.

Table of Contents

Copyright © OASIS Open 2004. All Rights Reserved.

1 Introduction

{Non-normative}

The ***policy*** evaluation performed by an XACML ***Policy Decision Point***, or ***PDP***, is defined in terms of a single requested ***resource*** in the XACML Specification [XACML], with the ***authorization decision*** contained in a single `<Result>` element of the response ***context***. A ***Policy Enforcement Point***, or ***PEP***, however, may wish to submit a single request ***context*** for ***access*** to multiple ***resources***, and may wish to obtain a single response ***context*** that contains a separate ***authorization decision*** (`<Result>` element) for each requested ***resource***. Such a request ***context*** might be used to avoid sending multiple ***decision request*** messages between a ***PEP*** and ***PDP,*** for example. Alternatively, a ***PEP*** may wish to submit a single request ***context*** for all the nodes in a hierarchy, and may wish to obtain a single ***authorization decision*** (`<Result>` element) that indicates whether ***access*** is permitted to all of the requested nodes. Such a request ***context*** might be used when the requester wants ***access*** to an entire XML document, to an entire sub-tree of elements in such a document, or to an entire file system directory with all its subdirectories and files, for example.

This Profile describes three ways in which a ***PEP*** can request a***uthorization decisions*** for multiple ***resources*** in a single request ***context,*** and how the result of each such ***authorization decision*** is represented in the single response ***context*** that is returned to the ***PEP***.

This Profile also describes two ways in which a ***PEP*** can request a single ***authorization decision*** in response to a request for all the nodes in a hierarchy***.***

Support for each of the mechanisms described in this Profile is optional for compliant XACML implementations.

1.1 Terminology

Access - Performing an ***action***.

Access control - Controlling ***access*** in accordance with a ***policy.***

Action – An operation on a ***resource***.

Applicable policy - The set of ***policies*** and ***policy sets*** that governs ***access*** for a specific ***decision request.***

Attribute - Characteristic of a ***subject***, ***resource, action*** or ***environment*** that may be referenced in a ***predicate*** or ***target*** (see also – ***named attribute***) or provided in a ***context***.

Authorization decision - The result of evaluating ***applicable policy,*** returned by the ***PDP*** to the ***PEP.*** A function that evaluates to "`Permit`", "`Deny`", "`Indeterminate`" or "`NotApplicable`", and (optionally) a set of ***obligations.***

Bag – An unordered collection of values, in which there may be duplicate values.

Context - The canonical representation of a ***decision request*** and an ***authorization decision.***

Context Handler – the component of an XACML ***PDP*** that maps `<AttributeSelector>` and `<AttributeDesignator>` references in a ***policy*** or ***policy set*** into ***attribute*** values and supplies those values to the ***PDP*** policy evaluation process. In this Profile, the ***context handler*** is also responsible for performing specified pre-processing operations on a request ***context*** and specified post-processing operations on a response ***context.***

Decision – The result of evaluating a ***rule, policy*** or ***policy set.***

Decision request - The request by a ***PEP*** to a ***PDP*** to render an ***authorization decision.***

Hierarchical resource – A ***resource*** that is organized as a tree or forest (Directed Acyclic Graph) of individual ***resources*** called ***nodes.***

Copyright © OASIS Open 2004. All Rights Reserved.

Node – An individual ***resource*** that is part of a ***hierarchical resource.***

Obligation - An operation specified in a ***policy*** or ***policy set*** that should be performed by the ***PEP*** in conjunction with the enforcement of an ***authorization decision.***

Policy - A set of ***rules,*** an identifier for the ***rule-combining algorithm*** and (optionally) a set of ***obligations.*** May be a component of a ***policy set.***

Policy administration point (PAP) - The system entity that creates a ***policy*** or ***policy set.***

Policy decision point (PDP) - The system entity that evaluates ***applicable policy*** and renders an ***authorization decision***. This term is defined in a joint effort by the IETF Policy Framework Working Group and the Distributed Management Task Force (DMTF)/Common Information Model (CIM) in . This term corresponds to "*Access Decision Function*" (ADF) in .

Policy enforcement point (PEP) - The system entity that performs ***access control***, by making ***decision requests*** and enforcing ***authorization decisions***. This term is defined in a joint effort by the IETF Policy Framework Working Group and the Distributed Management Task Force (DMTF)/Common Information Model (CIM) in . This term corresponds to "*Access Enforcement Function*" (AEF) in .

Policy set – A set of ***policies,*** other ***policy sets,*** a policy-combining algorithm and {optionally} a set of ***obligations*****.** May be a component of another ***policy set.***

Resource - Data, service or system component. The object for which ***access*** is requested in a ***decision request***.

1.2 Notation

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

> "they MUST only be used where it is actually required for interoperation or to limit behavior which has potential for causing harm (e.g., limiting retransmissions)"

These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

The phrase ***{Normative, but optional}*** means that the described functionality is optional for compliant XACML implementations, but, if the functionality is claimed as being supported according to this Profile, then it SHALL be supported in the way described.

Commonly used ***resource attributes*** are abbreviated as follows:

- "`resource-id`" ***attribute –*** a ***resource attribute*** with an `AttributeId` of "`urn:oasis:names:tc:xacml:1.0:resource:resource-id`".
- "scope" ***attribute*** - a ***resource attribute*** with an `AttributeId` of "`urn:oasis:names:tc:xacml:2.0:resource:scope`". See Section 4.1 for more information about this ***attribute.***

Copyright © OASIS Open 2004. All Rights Reserved.

2 Requests for multiple resources

{Normative, but optional}

A single XACML request ***context*** MAY represent a request for ***access*** to multiple ***resources***, with a separate ***authorization decision*** desired for each ***resource***. The syntax and semantics of such requests and responses are specified in this Section.

The <Result> elements produced by evaluating a request for ***access*** to multiple ***resources*** SHALL be identical to those that would be produced from a series of requests, each requesting ***access*** to exactly one of the ***resources.*** Each such resource is called an ***Individual Resource.*** The conceptual request ***context*** that corresponds to each <Result> element is called an ***Individual Resource Request.*** The ResourceId value in the <Result> element SHALL be the <AttributeValue> of the "resource-id" ***attribute*** in the corresponding ***Individual Resource Request.*** This mapping of an original request ***context*** containing multiple ***authorization decision requests*** to ***Individual Resource Requests,*** and the corresponding mapping of multiple ***authorization decisions*** to multiple <Result> elements in a single response ***context*** MAY be performed by the ***Context Handler*** described in the non-normative Data-flow model of the core XACML specification [XACML]***.*** This Profile does NOT REQUIRE that the implementation of the evaluation of a request for ***access*** to multiple ***resources*** conform to the preceding model or that actual ***Individual Resource Requests*** be constructed. The Profile REQUIRES only that the <Result> elements SHALL be the same as if the preceding model were used.

Three ways of specifying requests for ***access*** to multiple ***resources*** are described in the following Sections. Each way of specifying requests describes the ***Individual Resource Requests*** that correspond to the <Result> elements in the response ***context.***

A single XACML request ***context*** submitted by a PEP MAY use more than one of these ways of requesting ***access*** to multiple ***resources*** in different <Resource> elements.

2.1 Nodes identified by "scope"

{Normative, but optional}

This Section describes the use of two values for the "scope" ***resource attribute*** to specify a request for ***access*** to multiple ***resources*** in a hierarchy***.*** This syntax MAY be used with any ***hierarchical resource [Hierarchical]***, regardless of whether it is an XML document or not. The "scope" ***resource attribute*** is defined in Section 4.

2.1.1 Profile URI

The following URIs SHALL be used as URI identifiers for the functionality specified in this Section of this Profile. The first identifier SHALL be used when the functionality is supported for XML ***resources***, and the second identifier SHALL be used when the functionality is supported for ***resources*** that are not XML documents:

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:scope:xml
urn:oasis:names:tc:xacml:2.0:profile:multiple:scope:non-xml
```

2.1.2 Original request context syntax

The original XACML request ***context*** <Resource> element SHALL contain a "scope" ***attribute*** with a value of either "Children", or "Descendants". If the requested ***resources*** are in an XML document, then the <ResourceContent> element SHALL be present and SHALL contain the entire XML document of which the requested elements are a part. Also, if the requested ***resources*** are in an XML document, then the XPath [XPath] expression used as the value of the "resource-id" ***attribute*** SHALL evaluate to a nodeset containing exactly one ***node***.

Copyright © OASIS Open 2004. All Rights Reserved.

2.1.3 Semantics

Such a request ***context*** SHALL be interpreted as a request for ***access*** to a set of ***nodes*** in a hierarchy relative to the single ***node*** specified in the "`resource-id`" ***attribute***. If the value of the "`scope`" ***attribute*** is "`Children`", each ***Individual Resource*** is the one ***node*** indicated by the "`resource-id`" ***attribute*** (or ***attributes***, where the single ***resource*** has multiple normative identifiers) and all of its immediate child ***nodes***. If the value of the "`scope`" ***attribute*** is "`Descendants`", the ***Individual Resource*** is the one ***node*** indicated by the "`resource-id`" attribute and all of its descendant ***nodes***.

Each ***Individual Resource Request*** SHALL be identical to the original request ***context*** with two exceptions: the "`scope`" ***attribute*** SHALL NOT be present and the `<Resource>` element SHALL represent a single ***Individual Resource.*** This `<Resource>` element SHALL contain at least one "`resource-id`" ***attribute,*** and all values for such ***attributes*** SHALL be unique, normative identities of the ***Individual Resource***. If the "`resource-id`" ***attribute*** in the original request ***context*** contained an `Issuer`, the "`resource-id`" ***attributes*** in the ***Individual Resource Request*** SHALL contain the same `Issuer`. If a `<ResourceContent>` element was present in the original request ***context,*** then that same `<ResourceContent>` element SHALL be included in each ***Individual Resource Request***.

Neither XACML nor this Profile specifies how the ***Context Handler*** obtains the information required to determine which ***nodes*** are children or descendants of a given ***node,*** except in the case of an XML document, where the information SHALL be obtained from the `<ResourceContent>` element.

2.2 Nodes identified by XPath

{Normative, but optional}

This Section describes use of an XPath [XPath] expression in the "`resource-id`" ***attribute,*** together with an "`XPath-expression`" value in the "`scope`" ***attribute*** to specify a request for ***access*** to multiple nodes in an XML document. This syntax SHALL be used only with ***resources*** that are XML documents.

2.2.1 Profile URI

The following URI SHALL be used as the URI identifier for the functionality specified in this Section of this Profile:

`urn:oasis:names:tc:xacml:2.0:profile:multiple:xpath-expression`

2.2.2 Original request context

The original XACML request ***context*** `<Resource>` element SHALL contain a `<ResourceContent>` element and a "`resource-id`" ***attribute*** with a `DataType` of "`urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression`" (defined in [Hierarchical]), such that the `<AttributeValue>` of the "`resource-id`" ***attribute*** is an XPath expression that evaluates to a nodeset that represents multiple ***nodes*** in the `<ResourceContent>` element. The `<Resource>` element SHALL contain a "`scope`" ***attribute*** with a value of "`XPath-expression`".

2.2.3 Semantics

Such a request ***context*** SHALL be interpreted as a request for ***access*** to the multiple ***nodes*** in the nodeset represented by the `<AttributeValue>` of the "`resource-id`" ***attribute***. Each such ***node*** SHALL represent an ***Individual Resource.***

Each ***Individual Resource Request*** SHALL be identical to the original request ***context*** with two exceptions: the "`scope`" ***attribute*** SHALL NOT be present and the "`resource-id`" ***attribute*** value SHALL be an XPath expression that evaluates to a single ***node*** in the `<ResourceContent>` element. That node SHALL be the ***Individual Resource***. If the "`resource-id`" ***attribute*** in the original request ***context*** contained an `Issuer`, the "`resource-id`" ***attribute*** in the ***Individual Resource Request*** SHALL contain the same `Issuer`.

Copyright © OASIS Open 2004. All Rights Reserved.

2.3 Multiple <Resource> elements

{Normative, but optional}

This Section describes use of multiple `<Resource>` elements in a request ***context*** to specify a request for ***access*** to multiple ***resources.*** This syntax MAY be used with any ***resource*** or ***resources***, regardless of whether they are XML documents or not and regardless of whether they are ***hierarchical resources*** [Hierarchical] or not.

2.3.1 Profile URI

The following URI SHALL be used as the URI identifier for the functionality specified in this Section of this Profile:

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:multiple-resource-elements
```

2.3.2 Original request context

The XACML request ***context*** SHALL contain multiple `<Resource>` elements.

2.3.3 Semantics

Such a request ***context*** SHALL be interpreted as a request for ***access*** to all ***resources*** specified in the individual `<Resource>` elements. Each `<Resource>` element SHALL represent one ***Individual Resource*** unless that element utilizes the other mechanisms described in this Profile.

For each `<Resource>` element, one ***Individual Resource Request*** SHALL be created. This ***Individual Resource Request*** SHALL be identical to the original request ***context*** with one exception: only the one `<Resource>` element SHALL be present. If such a `<Resource>` element contains a "`scope`" ***attribute*** having any value other than "`Immediate`", then the ***Individual Resource Request*** SHALL be further processed according to the corresponding Section of this Profile listed in Section 4.1. This processing may involve decomposing the one ***Individual Resource Request*** into other ***Individual Resource Requests*** before evaluation by the ***PDP***.

Note that the semantics for multiple `<Resource>` elements are very different from the semantics for multiple `<Subject>` elements in a request ***context*** as described in the XACML core specification [XACML].

Copyright © OASIS Open 2004. All Rights Reserved.

3 Requests for an entire hierarchy

{Normative, but optional}

In some cases, a ***resource*** is hierarchical, but the ***authorization decision request*** is intended to request ***access*** to all the ***nodes*** within that ***resource*** or to an entire sub-hierarchy of ***nodes*** within that ***resource***. This might be the case when ***access*** to an XML document is being requested for purposes of making a copy of the entire document, or where ***access*** to an entire file system directory with all its subdirectories and files is being requested. A single `<Result>` is desired, indicating whether the requester is permitted to ***access*** the entire set of ***nodes***.

The `<Result>` element produced by evaluating such a request for ***access*** SHALL be identical to that produced by the following process. A series of request ***contexts*** is evaluated, each requesting ***access*** to exactly one ***node*** of the hierarchy. The `<Decision>` in the single `<Result>` that is returned to the ***PEP*** SHALL be "`Permit`" if and only if all `<Result>` elements resulting from the evaluation of the individual ***nodes*** contained a `<Decision>` of "`Permit`". Otherwise, the `<Decision>` in the single `<Result>` returned to the ***PEP*** SHALL be "`Deny`". This Profile does NOT REQUIRE that the implementation of the evaluation of a request for ***access*** to such a ***hierarchical resource*** conform to the preceding model or that actual request ***contexts*** corresponding to the individual ***nodes*** in the hierarchy be constructed. This Profile REQUIRES only that the `<Result>` element SHALL be the same as if the preceding model were used.

Two syntax's for this functionality are specified in the following Sections, one for use with ***resources*** that are XML documents, and the other for use with ***resources*** that are not XML documents.

3.1 XML resources

{Normative, but optional}

This Section describes the syntax for requesting ***access*** to an entire XML document, or to an element within that document with all its recursive sub-elements.

3.1.1 Profile URI

The following URI SHALL be used as the identifier for the functionality specified in this Section of this Profile:

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:entire-hierarchy:xml
```

3.1.2 Original request context

The `<Resource>` element in the original request ***context*** SHALL contain a "`scope`" ***attribute*** with a value of "`EntireHierarchy`".

The `<Resource>` element in the original request ***context*** SHALL contain a single "`resource-id`" ***attribute*** with a `DataType` of "`urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression`" (defined in [Hierarchical]), such that the `<AttributeValue>` evaluates to a nodeset that represents exactly one ***node*** in the `<ResourceContent>` element.

The `<Resource>` element in the original request ***context*** MAY contain other ***attributes.***

3.1.3 Semantics

The `<Result>` of such a request SHALL be equivalent to that produced by the following process. For each ***node*** in the requested hierarchy, the ***Context Handler*** SHALL create a new request ***context***. Each such request ***context*** SHALL contain a single `<Resource>` element having a "`resource-id`" ***attribute*** with a `DataType` of "`urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression`" (defined in [Hierarchical]) and a value that is an XPath [XPath] expression that evaluates

Copyright © OASIS Open 2004. All Rights Reserved.

to a nodeset that contains exactly that one ***node*** in the <ResourceContent> element. The ***Context Handler*** SHALL submit each such new request ***context*** to the ***PDP*** for evaluation and SHALL keep track of the <Decision> in the corresponding <Result> elements. If and only if all the new request ***contexts*** evaluate to "Permit", then a single <Result> containing a <Decision> of "Permit" SHALL be placed into the response ***context*** returned to the ***PEP.*** If any of the new request ***contexts*** evaluates to "Deny", "Indeterminate", or "NotApplicable", then a single <Result> containing a <Decision> of "Deny" SHALL be placed into the response ***context*** returned to the ***PEP.***

3.2 Non-XML resources

{Normative, but optional}

This Section describes the syntax for requesting ***access*** to an entire hierarchy of ***nodes*** within a ***hierarchical resource*** that is not an XML document.

3.2.1 Profile URI

The following URI SHALL be used as the identifier for the functionality specified in this Section of this Profile:

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:entire-hierarchy:non-xml
```

3.2.2 Original request context

The <Resource> element in the original request ***context*** SHALL contain a "scope" ***attribute*** with a value of "EntireHierarchy".

The <Resource> element in the original request ***context*** SHALL contain a single "resource-id" ***attribute*** that represents a single ***node*** in a ***hierarchical resource.***

The <Resource> element in the original request ***context*** MAY contain other ***attributes.***

The representation of ***nodes*** in a ***hierarchical resource*** specified in Section 2.2 of the *XACML Profile for Hierarchical Resources* [Hierarchical] MAY be used to represent the identity of the single ***node.***

3.2.3 Semantics

The <Result> of such a request SHALL be equivalent to that produced by the following process. For each ***node*** in the requested hierarchy, the ***Context Handler*** SHALL create a new request ***context***. Each such request ***context*** SHALL contain a single <Resource> element having a "resource-id" ***attribute*** with a value that is the identity of that one ***node*** in the hierarchy. The ***Context Handler*** SHALL submit each such new request ***context*** to the ***PDP*** for evaluation and SHALL keep track of the <Decision> in the corresponding <Result> elements. If and only if all the new request ***contexts*** evaluate to "Permit", then a single <Result> containing a <Decision> of "Permit" SHALL be placed into the response ***context*** returned to the ***PEP.*** If any of the new request ***contexts*** evaluates to "Deny", "Indeterminate", or "NotApplicable", then a single <Result> containing a <Decision> of "Deny" SHALL be placed into the response ***context*** returned to the ***PEP.***

Neither XACML nor this Profile specifies how the ***Context Handler*** obtains the information required to determine which ***nodes*** are descendants of the originally specified ***node,*** or how to represent the identity of each such ***node.*** The representation of ***nodes*** in a ***hierarchical resource*** specified in Section 2.2 of the *XACML Profile for Hierarchical Resources [Hierarchical]* MAY be used to represent the identity of each such ***node.***

Copyright © OASIS Open 2004. All Rights Reserved.

4 New attribute identifiers

{Normative}

4.1 "scope"

The following identifier is used as the `AttributeId` of a ***resource attribute*** that indicates the scope of a request for ***access*** in a single `<Resource>` element of a request ***context.***

`urn:oasis:names:tc:xacml:2.0:resource:scope`

The ***attribute*** SHALL have a `DataType` of "`http://www.w3.org/2001/XMLSchema#string`***".***

The valid values for this ***attribute*** are listed below, along with a reference to the Section of this Profile or to the core XACML specification that describes how the `<Resource>` element is to be processed. An implementation MAY support any subset of these values, including the empty set.

- "`Immediate`" - The `<Resource>` element refers to a single non-***hierarchical resource*** or to a single ***node*** in a ***hierarchical resource.*** This is the default value, if no "`scope`" ***attribute*** is present. The `<Resource>` element SHALL be processed according to the core XACML specification [XACML].
- "`Children`" - The `<Resource>` element refers to multiple ***resources*** in a hierarchy. The set of ***resources*** consists of a single ***node*** described by the "`resource-id`" ***resource attribute*** and of all that ***node's*** immediate children in the hierarchy. The `<Resource>` element SHALL be processed according to Section 2.1 of this Profile.
- "`Descendants`***"*** - The `<Resource>` element refers to multiple ***resources*** in a hierarchy. The set of ***resources*** consists of a single ***node*** described by the "`resource-id`" ***resource attribute*** and of all that ***node's*** descendants in the hierarchy. The `<Resource>` element SHALL be processed according to Section 2.1 of this Profile.
- "`XPath-expression`" - The `<Resource>` element refers to multiple ***resources.*** The set of ***resources*** consists of the ***nodes*** in a nodeset described by the "`resource-id`" ***resource attribute.*** Each of the ***nodes*** SHALL be contained in the `<ResourceContent>` element of the `<Resource>` element. The `<Resource>` element SHALL be processed according to Section 2.2 of this Profile.
- "`EntireHierarchy`" - The `<Resource>` element refers to a single ***resource.*** The ***resource*** consists of a ***node*** described by the "`resource-id`" ***resource attribute*** along with all that ***node's*** descendants. All of the ***nodes*** SHALL be ***nodes*** in an XML document that is contained in the `<ResourceContent>` element of the `<Resource>` element. The `<Resource>` element SHALL be processed according to Section 3.

Copyright © OASIS Open 2004. All Rights Reserved.

5 New profile identifiers

{Normative}

The following URI values SHALL be used as URI identifiers for the functionality specified in various Sections of this Profile:

Section 2.1: "scope attribute of "children" or "descendants" in <Resource>: XML resources

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:scope:xml
```

Section 2.1: "scope attribute of "children" or "descendants" in <Resource>: Non-XML resources

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:scope:non-xml
```

Section 2.2: XPath expression in "resource-id" attribute

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:xpath-expression
```

Section 2.3: Multiple <Resource> elements

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:multiple-resource-elements
```

Section 3.1: Requests for an entire hierarchy: XML resources

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:entire-hierarchy:xml
```

Section 3.2: Requests for an entire hierarchy: Non-XML resources

```
urn:oasis:names:tc:xacml:2.0:profile:multiple:entire-hierarchy:non-xml
```


Copyright © OASIS Open 2004. All Rights Reserved.

6 References

[Hierarchical] A. Anderson, ed., *Hierarchical Resource profile of XACML,* Committee Draft 01, 30 September 2004, http://docs.oasis-open.org/xacml/access_control-xacml-2.0-hier_profile-spec-cd-01.pdf.

[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.

[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.

[XPath] *XML Path Language (XPath)*, Version 1.0, W3C Recommendation 16, November 1999. Available at http://www.w3.org/TR/xpath.

Copyright © OASIS Open 2004. All Rights Reserved.

A. Acknowledgments

The editor 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

Copyright © OASIS Open 2004. All Rights Reserved.

B. Revision History

	Date	By Whom	What
CD-01	21 Sept 2004	XACML committee	Committee Draft

Copyright © OASIS Open 2004. All Rights Reserved.

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.

Copyright © OASIS Open 2004. All Rights Reserved.

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

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

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