XACML Profile for Hierarchical Resources

Working Draft 05, 29 July 2004

Document identifier:
oasis-xacml-profile-hierarchical-resources-wd-05

Location:

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

Abstract:
This document provides a profile for the use XACML with resources that are structured as
hierarchies. The profile includes both XML document resources and resources that are not XML
documents. The profile covers requesting access to hierarchical resources and specifying
policies that apply to hierarchical resources.

Status:
This version of the specification is a working draft of the committee. As such, it is expected to
change prior to adoption as an OASIS Standard.

Committee members should send comments on this specification to the xacml@lists.oasis-
open.org list. Others should subscribe to and send comments to the xacml-
comment@lists.oasis-open.org list. To subscribe, send an email message to xacml-comment-
request@lists.oasis-open.org with the word "subscribe" as the body of the message.

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 XACML TC web page (http://www.oasis-
open.org/committees/xacml/).

For any errata page for this specification, please refer to the XACML Profile for Hierarchical
Resources section of the XACML TC web page (http://www.oasis-open.org/committees/xacml/).
## Table of Contents

1. Introduction ........................................................................................................... 3
   1.1 Terminology ........................................................................................................ 3
   1.2 Notation .............................................................................................................. 4

2. Representing the identity of a node ........................................................................ 5
   2.1 Nodes in XML documents .................................................................................... 5
   2.2 Nodes in resources that are not XML documents ................................................ 5

3. Requesting access to a node .................................................................................... 7
   3.1 Nodes in an XML document .................................................................................... 7
   3.2 Nodes in a resource that is not an XML document ................................................ 8

4. Stating policies that apply to nodes ......................................................................... 9
   4.1 Policies applying to nodes in any hierarchical resource ......................................... 9
   4.2 Policies applying to nodes only in XML documents .............................................. 9
   4.3 Policies applying to nodes only to non XML resources ......................................... 9

5. New data-types ....................................................................................................... 11
   5.1 xpath-expression ................................................................................................ 11

6. New attribute identifiers .......................................................................................... 12
   6.1 document-id ........................................................................................................ 12
   6.2 resource-ancestor ................................................................................................ 12
   6.3 resource-parent ..................................................................................................... 12

7. New profile identifiers .............................................................................................. 13

8. References ............................................................................................................... 14
1 Introduction

It is often the case that a resource is organized as a hierarchy. Examples include file systems, XML documents, and organizations. This Profile specifies how XACML can provide access control for a resource that is organized as a hierarchy.

In XACML, a resource organized as a hierarchy may be a “tree” (a hierarchy with a single root) or a “forest” (a hierarchy with multiple roots), but the hierarchy may not have cycles. Another term for this type of hierarchy is a Directed Acyclic Graph or DAG. Such resources are called hierarchical resources in this Profile.

In XACML, the nodes in a hierarchical resource are treated as individual resources. An authorization decision that permits access to an interior node does not imply that access to its descendant nodes is permitted. An authorization decision that denies access to an interior node does not imply that access to its descendant nodes is denied.

There are three types of facilities specified in this Profile for dealing with hierarchical resources:

• Representing the identity of a node.
• Requesting access to a node.
• Stating policies that apply to one or more nodes.

Support for each of these facilities is optional.

In dealing with a hierarchical resource, it may be useful to request authorization decisions for multiple nodes in the resource in a single authorization decision request. Ways to make such requests are specified in a separate profile – the XACML Profile for Requests for Multiple Resources [MULTIPLE]. That profile also provides a way to return a single authorization decision when multiple nodes are requested. That profile may be considered to be layered on top of this Profile, which in turn is layered on top of the behavior specified in the core XACML specification [XACML].

This profile for hierarchical resources assumes that all requests for access to multiple nodes in a hierarchical resource [MULTIPLE] have been resolved to individual requests for access to a single node.

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.

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.

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 RFC3198. This term corresponds to "Access Decision Function" (ADF) in ISO10181-3.

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 RFC3198. This term corresponds to "Access Enforcement Function" (AEF) in ISO10181-3.

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 RFC 2119 [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.

Example code listings appear like this.

In descriptions of syntax, elements in angle brackets ("<", ">") are to be replaced by appropriate values, square brackets ("[", "]") enclose optional elements, elements in quotes are literal components, and "***" indicates that the preceding element may occur zero or more times.
2 Representing the identity of a node

{Normative}

In order for XACML policies to apply consistently to nodes in a hierarchical resource, it is necessary for the nodes in that resource to be represented in a consistent way. If a policy refers to a node using one representation, but a request refers to the node using a different representation, then the policy will not apply, and security may be compromised.

The following sections describe RECOMMENDED representations for nodes in hierarchical resources. Alternative representations of nodes in a given resource are permitted so long as all Policy Administration Points and all Policy Enforcement Points that deal with that resource have contracted to use the alternative representation.

2.1 Nodes in XML documents

{Normative, but optional}

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:hierarchical:xml-node-id.

The identity of a node in a resource that is an XML document instance SHALL be an XPath expression that evaluates to exactly that one node in the copy of the resource that is contained in the <ResourceContent> element of the <Resource> element of the <Request>.

2.2 Nodes in resources that are not XML documents

{Normative, but optional}

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:hierarchical:non-xml-node-id.

The identity of a node in a hierarchical resource that is not an XML document instance SHALL be represented as a URI that conforms to [RFC2396]. Such URIs are of the following form.

<scheme>"://"<authority>"/"<pathname>

File system resources SHALL use the "file:" scheme. If no standard <scheme> for the resource type is specified in [RFC2396] or in a related standard for a registered URI scheme, then the URI SHALL use the "file:" scheme.

The <pathname> portion of the URI SHALL be of the form

<root name> [ "/" <node name> ]*

The sequence of <root name> and <node name> values SHALL correspond to the individual hierarchical component names of ancestors of the represented node along the path from a <root> node to the represented node.

The following canonicalization SHALL be used.

- The encoding of the URI SHALL be UTF8.
- Case-insensitive portions of the URI SHALL be lower case.
- Escaping of characters SHALL conform to [RFC2396].
- The <authority> portion of the URI SHALL be specified and SHALL be the standard authority representation for the given resource type. Where the <authority> could be specified using either a Distributed Name Service (DNS) name or a numeric IPv4 or IPv6 address, the DNS name SHALL be used.
The components of the `<pathname>` portion of the URI SHALL be specified using the canonical form for such path components at the `<authority>`.

In accordance with [RFC2396], the separator character between hierarchical components of the `<pathname>` portion of the URI SHALL be the character `/`. Sequences of the `/` character SHALL be resolved to a single `/`. Node identities SHALL NOT terminate with the `/` character.

All links in the `<pathname>` SHALL be resolved.

All `<pathname>` values SHALL be absolute.

The `..` and `.` `<pathname>` components used to specify “level above this hierarchy level” and “this hierarchy level”, respectively, SHALL be resolved to their actual component values.

If there is more than one fully resolved, absolute path from a `<root>` at the `<authority>` to the represented node, then a separate resource attribute with AttributeId "urn:oasis:names:tc:xacml:2.0:resource:resource-id" and DataType http://urn:oasis:names:tc:xacml:1.0:datatypes:attribute SHALL be present in the Request Context for each such path.
3 Requesting access to a node

{Normative}

In order for XACML policies to apply consistently to nodes in a hierarchical resource, it is necessary for each request context that represents a request for access to a node in that resource to use a consistent description of that node access. If a policy refers to certain expected attributes of a node, but the request context does not contain those attributes, or if the attributes are not expressed in the expected way, then the policy may not apply, and security may be compromised.

The following sections describe RECOMMENDED request context descriptions of access to nodes in hierarchical resources. Alternative representations of such requests are permitted so long as all Policy Administration Points and all Policy Enforcement Points that deal with that resource have contracted to use the alternative representation.

3.1 Nodes in an XML document

{Normative, but optional}

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:hierarchical:xml-node-req.

In order to request access to a node in an XML document, the request context <Resource> element SHALL contain the following elements and attributes.

- A <ResourceContent> element that contains the entire XML document instance of which the requested node is a part.
- An <Attribute> element with an AttributeId of "urn:oasis::names:tc:xacml:2.0:resource:resource-id" and a DataType of "urn:oasis:names:tc:xacml:2.0:datatype:xpath-expression". The <AttributeValue> of this <Attribute> SHALL be an XPath expression that evaluates to a nodeset containing a single node in the <ResourceContent> element. That single node SHALL be the node to which access is requested. This <Attribute> MAY specify an Issuer.
- An <Attribute> element with an AttributeId of "urn:oasis::names:tc:xacml:2.0:resource:resource-parent" and a DataType of "urn:oasis:names:tc:xacml:2.0:datatype:xpath-expression". The <AttributeValue> of this attribute SHALL be an XPath expression that evaluates to a nodeset containing only a single node in the <ResourceContent> element. That single node SHALL be the immediate parent of the node represented in the "resource-id" attribute. This <Attribute> MAY specify an Issuer.
- For each node in the XML document instance that is an ancestor of the node represented by the "resource-id" attribute, an <Attribute> element with an AttributeId of "urn:oasis::names:tc:xacml:2.0:resource:resource-ancestor" and a DataType of "urn:oasis:names:tc:xacml:2.0:datatype:xpath-expression". The <AttributeValue> of each such attribute SHALL be an XPath expression that evaluates to a nodeset containing only a single node in the <ResourceContent> element. That single node SHALL be the respective ancestor node. This <Attribute> MAY specify an Issuer.

Additional attributes MAY be included in the <Resource> element. In particular, the following attribute MAY be included.

- An <Attribute> element with an AttributeId of "urn:oasis::names:tc:xacml:2.0:resource:document-id" and a DataType of "urn:oasis:names:tc:xacml:2.0:datatype:anyURI". The <AttributeValue> of this <Attribute> SHALL be a URI that identifies the XML document of which the requested resource is a part, and of which a copy is present in the <ResourceContent> element. This <Attribute>
MAY specify an Issuer.

### 3.2 Nodes in a resource that is not an XML document

**{Normative, but optional}**

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:hierarchical:non-xml-node-req`.

In order to request access to a node in a **hierarchical resource** that is not an XML document, the request context `<Resource>` element SHALL contain the following elements and attributes.

- For each normative representation of the requested node, an `<Attribute>` element with `AttributeName` "urn:oasis::names:tc:xacml:2.0:resource:resource-id". The `<AttributeValue>` of this `<Attribute>` SHALL be a unique, normative identity of the node to which access is requested. The `DataType` of this `<Attribute>` SHALL depend on the representation chosen for the identity of nodes in this particular resource. This `<Attribute>` MAY specify an Issuer.

- For each immediate parent of the node specified in the "resource-id" attribute or attributes, and for each normative representation of that parent node, an `<Attribute>` element with `AttributeName` "urn:oasis::names:tc:xacml:2.0:resource:resource-parent". The `<AttributeValue>` of this `<Attribute>` SHALL be the normative identity of the parent node. The `DataType` of this `<Attribute>` SHALL depend on the representation chosen for the identity of nodes in this particular resource. This `<Attribute>` MAY specify an Issuer. If the requested node is part of a forest rather than part of a single tree, or if the parent node has more than one normative representation, there SHALL be at least one instance of this attribute for each parent along each path to the multiple roots of which the requested node is a descendant, and for each normative representation of each such parent.

- For each ancestor of the node specified in the "resource-id" attribute or attributes, and for each normative representation of that ancestor node, an `<Attribute>` element with `AttributeName` "urn:oasis::names:tc:xacml:2.0:resource:resource-ancestor". The `<AttributeValue>` of this `<Attribute>` SHALL be the normative identity of the ancestor node. The `DataType` of this `<Attribute>` SHALL depend on the representation chosen for the identity of nodes in this particular resource. This `<Attribute>` MAY specify an Issuer. For each "resource-parent" attribute, there SHALL be a corresponding "resource-ancestor" attribute. If the requested node is part of a forest rather than part of a single tree, or if the ancestor node has more than one normative representation, there SHALL be at least one instance of this attribute for each ancestor along each path to the multiple roots of which the requested node is a descendant, and for each normative representation of each such ancestor. The values for this attribute do not necessarily reflect the position of each ancestor node in the hierarchy.

Additional attributes MAY be included in the `<Resource>` element.
4 Stating policies that apply to nodes

{Non-normative}

This Section describes various ways to specify a policy predicate that can apply to multiple nodes in a hierarchical resource. This is not intended to be an exhaustive list.

4.1 Policies applying to nodes in any hierarchical resource

{Non-normative}

Resource attributes with the following AttributeId values, described in Section 6: New attribute identifiers for hierarchical resources of this Profile, MAY be used to state policies that apply to one or more nodes in any hierarchical resource.

urn:oasis:names:tc:xacml:2.0:resource:resource-ancestor


Note that a <ResourceAttributeDesignator> that refers to the "resource-ancestor" or "resource-parent" attribute will return a bag of values representing all ancestors or parents, respectively, of the resource to which access is being requested. The representations of the identities of these ancestors or parents will not necessarily indicate the path from the root of the hierarchy to the respective ancestor or parent unless the representation recommended in Section 3.2: Nodes in a resource that is not an XML document is used.

The standard XACML [XACML] bag and higher-order bag functions MAY be used to state policies that apply to one or more nodes in any hierarchical resource. The nodes used as arguments to these functions MAY be specified using a <ResourceAttributeDesignator> with the "resource-parent" or "resource-ancestor" AttributeId value.

4.2 Policies applying to nodes only in XML documents

{Non-normative}

For hierarchical resources that are XML document instances, the following function, described in the XACML 2.0 Specification [XACML] MAY be used to state policy predicates that apply to one or more nodes in that resource.

urn:oasis:names:tc:xacml:2.0:function:xpath-node-match

The standard XACML <AttributeSelector> element MAY be used in policies to refer to all or portions of an XML document contained in the <ResourceContent> element of a request context. It may also be used to refer to all or portions of an XML document or schema element used as the <AttributeValue> for any XACML attribute.

The standard XACML [XACML] bag and higher-order bag functions MAY be used to state policies that apply to one or more nodes in an XML document resource. The nodes used as arguments to these functions MAY be specified using an <AttributeSelector> that selects a portion of the <ResourceContent> element of the <Resource> element.

4.3 Policies applying to nodes only to non XML resources

{Non-normative}

For hierarchical resources that are not XML document instances, and where the URI representation of nodes specified in Section 2 of this Profile is used, the following functions described in the XACML 2.0 Specification [XACML] MAY be used to state policies that apply to one or more nodes in that resource.

urn:oasis:names:tc:xacml:1.0:function:anyURI-equal
5 New data-types

{Normative, but optional}

The following Data types values MAY be supported for use with hierarchical resources or with other uses of XML schema instances within XACML request contexts or policies.

5.1 xpath-expression

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:data-type:xpath-expression.

Attribute values having the following Data type SHALL be strings that SHALL be evaluated as XPath expressions. The result of evaluating such an attribute value SHALL be the nodeset resulting from an evaluation of the XPath expression.
6 New attribute identifiers

{Normative, but optional}

6.1 document-id

The following identifier indicates the identity of the XML document of which the requested resource is a part, and of which a copy is present in the <ResourceContent> element. Whenever access to a node in an XML document is requested, one or more instances of an attribute with this AttributeId MAY be provided in the <Resource> element of the request context. The DataType of these attributes SHALL be "urn:oasis:names:tc:xacml:2.0:data-type:anyURI".

urn:oasis:names:tc:xacml:2.0:resource:document-id

6.2 resource-ancestor

The following identifier indicates the identity of one ancestor node in the tree or forest of which the requested node is a part. Whenever access to a node in a hierarchical resource is requested, one instance of an attribute with this AttributeId SHALL be provided in the <Resource> element of the request context for each normative representation of each node that is an ancestor of the requested node.

urn:oasis:names:tc:xacml:2.0:resource:resource-ancestor

6.3 resource-parent

The following identifier indicates the identity of one parent node in the tree or forest of which the requested node is a part. Whenever access to a node in a hierarchical resource is requested, one instance of an attribute with this AttributeId SHALL be provided in the <Resource> element of the request context for each normative representation of each node that is a parent of the requested node.

7  New profile identifiers

{normative}

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

Section 2.1: Nodes in XML documents

urn:oasis:names:tc:xacml:2.0:profile:hierarchical:xml-node-id

Section 2.2: Nodes in resources that are not XML documents


Section 3.1: Nodes in an XML document

urn:oasis:names:tc:xacml:2.0:profile:hierarchical:xml-node

Section 3.2: Nodes in a resource that is not an XML document


Section 5.1: xpath-expression

urn:oasis:names:tc:xacml:2.0:datatype:xpath-expression
8 References


[MULTIPLE] A. Anderson, ed., XACML Profile for Requests for Multiple Resources,

A. Acknowledgments

The editor would like to acknowledge the contributions of the OASIS XACML 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.
- 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
### B. Revision History

<table>
<thead>
<tr>
<th>Rev</th>
<th>Date</th>
<th>By Whom</th>
<th>What</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>14 Apr 2004</td>
<td>Anne Anderson</td>
<td>Initial rewrite of Section 7.13.</td>
</tr>
<tr>
<td>02</td>
<td>13 May 2004</td>
<td>Anne Anderson</td>
<td>&quot;xpath-expression&quot; DataType. Remove resource attributes no longer needed. New section for requesting multiple resources. Require &lt;ResourceContent&gt; for XML resources. Added &quot;resource-ancestor&quot; and &quot;resource-parent&quot;.</td>
</tr>
<tr>
<td>04</td>
<td>3 Jun 2004</td>
<td>Anne Anderson</td>
<td>Formatted as a separate profile, making each feature optional.</td>
</tr>
<tr>
<td>05</td>
<td>29 Jul 2004</td>
<td>Anne Anderson</td>
<td>Editorial corrections for clarity. Included document-id in the list of new AttributeId values. Used regexp-uri-match as the name of the URI matching function. Added identifier URI values for each implementable option.</td>
</tr>
</tbody>
</table>
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.