
Hierarchical Resources for XACML 2.0

Working Draft, Version 2.0, 25 May 2004

6.3 Element <Resource>

[change to description of the <Attribute> element as follows:]

<Attribute> [Any Number]

A sequence of *resource attributes*.

The <Resource> element MAY contain one or more <Attribute> elements with an AttributeId of “urn:oasis:names:tc:xacml:2.0:resource:resource-id”. Each such <Attribute> SHALL be an absolute and fully-resolved representation of the identity of the single *resource* to which *access* is being requested. If there is more than one such absolute and fully-resolved representation, and if any <Attribute> with this AttributeId is specified, then an <Attribute> for each such distinct representation of the *resource* identity SHALL be specified. All such <Attribute> elements SHALL refer to the same single *resource* instance. A Profile for a particular *resource* MAY specify a single normative representation for instances of the *resource*; in this case, any <Attribute> with this AttributeId SHALL use only this one representation.

A <Resource> element MAY contain additional <Attribute> elements.

11. References

[add following new references:]

- [RFC2396] T. Berners-Lee, et al., Uniform Resource Identifiers (URI): Generic Syntax, <http://www.ietf.org/rfc/rfc2396.txt>, IETF RFC 2396, August 1998.

A.2. Data-types

[add the following new DataType: normative, but not mandatory]

urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression

Attribute values having this data-type are strings that are to be evaluated as XPath expressions. The result of evaluating such an *attribute* value is the nodeset resulting from an evaluation of the XPath expression.

B.6 Resource attributes

[Remove the following existing Resource attributes currently defined:

```
urn:oasis:names:tc:xacml:2.0:resource:simple-file-name
urn:oasis:names:tc:xacml:2.0:resource:xpath
urn:oasis:names:tc:xacml:2.0:resource:ufs-path]
```

[Add the following Resource attributes:]

This identifier indicates the identity of one ancestor node in the hierarchy or hierarchies of which the requested node is a part. Whenever **access** to a node in a hierarchical **resource** is requested, one instance of this **attribute** SHALL be provided for each normative representation of a node that is an ancestor of the requested node in the tree or trees (in the case of a multi-rooted hierarchy, or “forest”) of which the requested node is a part.

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

This identifier indicates the identity of one parent node in the hierarchy or hierarchies of which the requested node is a part. Whenever **access** to a node in a hierarchical **resource** is requested, one instance of this **attribute** SHALL be provided for each normative representation of a node that is a parent of the requested node in the tree or trees (in the case of a multi-rooted hierarchy, or “forest”) of which the requested node is a part.

```
urn:oasis:names:tc:xacml:2.0:resource:resource-parent
```

7.[A] Requests for multiple resources (normative, but not mandatory)

A single XACML request **context** MAY represent a request for **access** to multiple **resources**. The semantics and syntax of such requests are specified in this section.

Requests for **access** to more than one **resource** SHALL be evaluated such that the <Result> elements in the response **context** are equivalent to the set of <Result> elements obtained by evaluating a sequence of request **context** instances, each of which specifies exactly one of the requested **resources**. In the request **context** that is evaluated by the **PDP**, each such individual **resource** SHALL be specified in a **resource attribute** having an AttributeId of “urn:oasis:names:tc:xacml:2.0:resource:resource-id”. Only information related to the one **resource** being evaluated SHALL be available for reference from XACML **policies**.

The response from a request for **access** to more than one **resource** SHALL be equivalent to that produced by the following process. The Context Handler SHALL construct a response **context** that contains exactly one <Result> element for each evaluation that was presented to the **PDP**. Each such <Result> element SHALL include a ResourceId attribute that contains the value of the **resource attribute** with AttributeId of “urn:oasis:names:tc:xacml:2.0:resource:resource-id” that was used in the request for that evaluation.

There are three different normative ways to request **access** to multiple **resources** in a single XACML request **context**. Support for such requests is not required for conformance with

the XACML specification. A single request **context** MAY use more than one way to request **access** to multiple **resources**. The following sections specify these different ways to request **access** to multiple **resources**.

7.[A].1 XPath expression in resource-id

This syntax SHALL be used only with **resources** that are XML documents.

An XACML request **context** <Resource> element MAY contain an **attribute** with an AttributeId of “urn:oasis:names:tc:xacml:2.0:resource:resource-id” (“resource-id” **attribute**) and DataType of “urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression”, such that the <AttributeValue> element evaluates to a nodeset that represents multiple nodes in the <ResourceContent> element. In this case, the <Resource> element SHALL NOT include an **attribute** with AttributeId “urn:oasis:names:tc:xacml:2.0:resource:scope”.

Such a request **context** SHALL be interpreted as a set of requests for **access** to each node in the nodeset represented by the <AttributeValue> of the original “resource-id” **attribute**. Each such request SHALL be evaluated separately by the **PDP**. For each such request, an XPath expression SHALL be constructed that evaluates only to the one node for which **access** is being requested for this evaluation. This XPath expression SHALL be used as the <AttributeValue> for the “resource-id” **attribute** in each request **context** evaluated by the **PDP**. If the original “resource-id” **attribute** contained an Issuer, the “resource-id” **attribute** presented to the **PDP** for evaluation SHALL contain the same Issuer. The original, multi-node XPath expression SHALL NOT be present in the request **context** evaluated by the **PDP** and SHALL NOT be referenced by XACML **policies**.

7.[A].2 Scope Attribute in <Resource>

This syntax MAY be used with any hierarchical **resource**, regardless of whether it is an XML document or not.

An XACML request **context** <Resource> element MAY contain a **resource attribute** with an AttributeId of “urn:oasis:names:tc:xacml:2.0:resource:scope” (“scope” **attribute**) and with a DataType of “<http://www.w3.org/2001/XMLSchema#string>”. The value for this **attribute** SHALL be either “Immediate”, “Children”, or “Descendants”. If the **resource** is an XML document, then the <ResourceContent> element SHALL be included in the <Resource> element. If the **resource** is an XML document, and the “scope” **attribute** is used, then the XPath expression used in the <AttributeValue> element of the **attribute** having AttributeId “urn:oasis:names:tc:xacml:2.0:resource:resource-id” (“resource-id” attribute) SHALL evaluate to a nodeset containing no more than one node.

Such a request **context** SHALL be interpreted as a request for **access** to a set of nodes in a hierarchy relative to the node specified in the “resource-id” **attribute**. If the value of

the “scope” *attribute* is “Immediate”, the requested node is the one node indicated by the “resource-id” *attribute*. If the value of the “scope” *attribute* is “Children”, the requested nodes are the node indicated by the “resource-id” *attribute* and all of its immediate child nodes. If the value of the “scope” *attribute* is “Descendants”, the requested nodes are the node indicated by the “resource-id” *attribute* and all of its descendant nodes.

The evaluation of such a request SHALL be equivalent to that produced by the following process. For each requested node, the Context Handler SHALL present a request *context* to the *PDP* for evaluation that is identical to the original request *context* except that the <Resource> element SHALL contain no more than one “resource-id” *attribute*, and the value of that *attribute* SHALL be the identity of the requested node. If the original “resource-id” *attribute* contained an Issuer, then the “resource-id” *attribute* evaluated by the PDP SHALL have the same Issuer. The “scope” *attribute* SHALL NOT be present in the <Resource> element of the request *context* that is evaluated by the *PDP* and SHALL NOT be referenced in XACML *policies*.

XACML does not specify how the Context Handler obtains the information required to determine which nodes are children or descendants of a given node.

See Section 7.13.1 for the representation of the identity of elements in a hierarchy.

7.[A].3 Multiple <Resource> elements

This syntax MAY be used with any *resource* or *resources*, whether they are XML documents or not and whether they are hierarchical or not.

An XACML request *context* MAY contain multiple <Resource> elements. Such a request *context* SHALL be interpreted as a request for *access* to all *resources* specified in the individual <Resource> elements.

The result of evaluating such a request *context* SHALL be equivalent to that produced by the following process. For each <Resource> element, the Context Handler SHALL present a request *context* to the *PDP* for evaluation that is identical to the original request *context* with one difference: the only <Resource> element that is included is the one being evaluated. The original request *context* with multiple <Resource> elements SHALL NOT be present in the request *context* evaluated by the *PDP* and SHALL NOT be referenced in XACML *policies*.

Note that the semantics for multiple <Resource> elements are very different from the semantics for multiple <Subject> elements in a request *context*.

7.13 Hierarchical resources (normative, but not mandatory)

It is often the case that a *resource* is organized as a hierarchy (e.g. file system, XML document, organization structure). Such resources are called hierarchical *resources*. XACML supports hierarchical *resources* that are trees or forests (i.e. Directed Acyclic Graphs) in two ways.

1. An XACML request *context* MAY request *access* to one or more nodes in a hierarchical *resource*.

2. An XACML **policy** may specify predicates that apply to one or more nodes in a hierarchical **resource**.

Support for hierarchical **resources** is not mandatory for conformance to XACML by a **PDP**.

Ways to request access to more than one node in a hierarchical **resource** are discussed in Section 7.[A]. The other aspects of support for hierarchical **resources** are discussed in the following sections.

In the following Sections, the term “resource-id” **attribute** refers to a **resource attribute** having an AttributeId of “urn::oasis:names:tc:xacml:2.0:resource:resource-id”. The term “scope” **attribute** refers to a **resource attribute** having an AttributeId of “urn::oasis:names:tc:xacml:2.0:resource:scope”. The term “resource-parent” **attribute** refers to a **resource attribute** having an AttributeId of “urn::oasis:names:tc:xacml:2.0:resource:resource-parent”. The term “resource-ancestor” **attribute** refers to a **resource attribute** having an AttributeId of “urn::oasis:names:tc:xacml:2.0:resource:resource-ancestor”.

7.13.1 Requests for access to nodes in a hierarchical resource

Requests for **access** to one or more nodes in a hierarchical **resource** SHALL conform to this Section of the XACML specification. This Section ensures that the following requirements are satisfied. Whenever a requested **resource** is a node in some hierarchical **resource**, the request **context** must reflect the node's position in the **resource** hierarchy as described in this Section, since **policies** may reference the requested node as part of a set of nodes. The Context Handler must be able to determine that a hierarchical **resource** is being **accessed** and to determine the individual identities of the elements in the hierarchy to which **access** is being requested. The Context Handler must be able to represent enough information about the hierarchical **resource** containing the elements being **accessed** that the **PDP** can determine from the **context** certain relationships between the elements being **accessed** and other elements in the hierarchy. A hierarchical **resource** must be represented in such a way that useful **policy** predicates can be applied to the **resource** representation. A given **resource** type must be represented in a consistent way so that **policy** predicates intended to apply to instances of that **resource** will always apply.

XML document **resources** are handled differently from other types of **resources**, so are discussed separately. A given **resource** type MAY be represented as either an XML document or as a non-XML **resource**, but SHALL NOT be represented in more than one way.

Section 7.[A] describes ways in which **access** to more than one node in a hierarchical **resource** MAY be requested. The following description applies to the request **context** that is evaluated by the **PDP** and that is referenced by XACML **policies**, where **access** to only a single node is represented.

7.13.1.1 Requests for XML Document Resources

In each request *context* that is evaluated by the *PDP* for *access* to a node in a hierarchical *resource*, the following elements and *attributes* SHALL be present.

- a <ResourceContent> element that contains the entire XML document instance of which the requested node is a part.
- a “resource-id” *attribute* with DataType [“urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression”](#). The <AttributeValue> of this *attribute* SHALL be an XPath expression that evaluates to a single node in the <ResourceContent> element that is the node to which *access* is requested. The “resource-id” *attribute* MAY specify an Issuer.
- a “resource-parent” *attribute* with DataType [“urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression”](#). The <AttributeValue> of this *attribute* SHALL be an XPath expression that evaluates to a nodeset containing only a single node, and that node SHALL be the immediate parent of the node represented in the “resource-id” *attribute*.
- for each node in the XML document instance that is an ancestor of the node represented by the “resource-id” *attribute*, a “resource-ancestor” *attribute* with DataType [“urn:oasis:names:tc:xacml:2.0:data-type:xpath-expression”](#). The <AttributeValue> of each such *attribute* SHALL be an XPath expression that evaluates to a nodeset containing only a single node, and that node SHALL be the respective ancestor node.

Representation of nodes in an XML document

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.

7.13.1.3 Requests for Non-XML Resources

In each request *context* that is evaluated by the *PDP* for *access* to a node in a hierarchical *resource*, the following *attributes* SHALL be present.

- a “resource-id” *attribute* that represents the identity of the node in the hierarchical *resource* to which *access* is being requested.
- For each immediate parent of the node specified in the “resource-id” *attribute*, a “resource-parent” *attribute* that represents the identity of a node that is an immediate parent of the “resource-id” node in the hierarchy. Note that there may be multiple instances of this *attribute* if the node is part of a forest rather than part of a single tree.
- For each ancestor of the node specified in the “resource-id” *attribute*, a “resource-ancestor” *attribute* that represents the identity of a node that is an ancestor of the “resource-id” node in the hierarchy. Note that there may be

multiple instances of this *attribute*, one for each ancestor of the “resource-id” *attribute* on the path from that *attribute* to any root of the hierarchical forest or tree of which the “resource-id” *attribute* is a part. The values for this *attribute* do not necessarily reflect the position of each ancestor node in the hierarchy.

Additional *resource attributes* MAY be specified.

See the following for a description of how the identities of nodes in a non-XML hierarchical *resource* are represented.

Representation of nodes not in an XML document

A *resource*-specific Profile MAY specify any representation suitable for that *resource* type, including an XML representation. If a *resource*-specific Profile specified an XML representation, then *resource* identity values SHALL be handled as described above for other XML document *resources*.

Unless otherwise specified by a *resource*-specific Profile, the identity of a node in a *resource* that is not an XML document SHALL be in the form specified the remainder of this Section.

The node identity SHALL be a URI that conforms to [RFC2396]. Such URIs are of the 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 another *resource*-specific Profile, then the URI SHALL use the “tree:” 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 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 canonicalizations 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. If multiple DNS names resolve to the given <authority>, the <authority> SHALL be the

DNS name with the fewest name components that is specified first in the DNS record for the <authority>.

- The components of the <pathname> portion of the URI SHALL be specified using the canonical form for 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 “/”. Identities SHALL NOT terminate with the “/” character.
- All links in the <pathname> SHALL be resolved.
- The “..” and “.” <pathname> components used to specify “level above this hierarchy level” and “this hierarchy level”, respectively, SHALL be resolved.
- All <pathname> values SHALL be absolute.
- 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” SHALL be included in the XACML Request for each such path.

7.13.2 Policy predicates applying to multiple nodes in a hierarchical resource

This Section describes ways to specify a **policy** predicate that can apply to multiple nodes in a hierarchical **resource**.

Appendix A.3.14 describes a function with the identifier “urn:oasis:names:tc:xacml:2.0:function:xpath-node-match”. This function MAY be used with the higher-order bag functions described in Appendix A.3.12 to specify predicates that apply one or more nodes in a hierarchical **resource** that is an XML document.

The “resource-ancestor” and “resource-parent” **resource attributes**, along with the Bag functions specified in Appendix A.14.9, the Set functions specified in Appendix A.14.10, and the Higher-order bag functions specified in Appendix A.14.11 MAY be used to specify **policy** predicates that apply to multiple nodes in any hierarchical **resource**. 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.

For nodes that are not in an XML document instance, and where the default URI representation of such nodes specified in Section 7.13.1 is used, the functions with identifiers “urn:oasis:names:tc:xacml:1.0:function:anyURI-equal” and “urn:oasis:names:tc:xacml:2.0:function:anyURI-match” specified in Appendices A.3.1 and A.3.[B] MAY be used to specify **policy** predicates that apply to multiple nodes in a hierarchical **resource**.