Hierarchical Resources for XACML 2.0:

Modifications to the XACML 2.0 specification if XACML Profile for Hierarchical Resources is approved.

Coordinated with

- XACML Profile for Hierarchical Resources, Working Draft 04, 3 June 2004, and
- **eXtensible Access Control Markup Language (XACML) Version 2.0,** Working draft 11, 14 May 2004

Questions:

1. Do we want to add a **non-normative section on "Profiles"**, in which we could describe all the currently available profiles and their scopes? I think this might be useful, but there is the problem that Profiles will not necessarily become standards on the same schedule as XACML 2.0.
2. **Existing references to XPath expressions**. I think the only ones that perhaps should be changed (via defining new 2.0 xpath-expression based versions) are the XPath functions in Appendix A. The definition of those functions is the only place that "string evaluated as an XPath expression" occurs in the spec.
   - **4.2.4.1 Rule 1**: The `<AttributeSelector>` element selects a ***bag*** of values from the request ***context*** using a free-form XPath expression. In this case, it selects the value of the `patient-number` in the ***resource***. Note that the namespace prefixes in the XPath expression are resolved with the standard XML namespace declarations.
   - **4.2.4.2 Rule 2**: The second argument selects the value of the`<md:parentGuardianId>` element from the ***resource*** content using the `<AttributeSelector>` element. This element contains a free-form XPath expression, pointing into the request ***context***. Note that all namespace prefixes in the XPath expression are resolved with standard namespace declarations. The `AttributeSelector` evaluates to the ***bag*** of values of type "http://www.w3.org/2001/XMLSchema#string
   - **4.2.4.2 Rule 2**: The `<AttributeSelector>` element selects the patient's date of birth by taking the XPath expression over the ***resource***
   - **4.2.4.4 Rule 4**: The second `<ResourceMatch>` element targets the ***rule*** at XML elements that match the XPath expression "/md:record/md:medical".
   - **5.8 Element <SubjectMatch>**: `<AttributeSelector>`[Required choice] MAY be used to identify one or more ***attribute*** values in the request ***context***. The XPath expression SHOULD resolve to an ***attribute*** in a `<Subject>` element of the request ***context***.
   - **5.11 Element <ResourceMatch>**: `<AttributeSelector>`[Required Choice] MAY be used to identify one or more ***attribute*** values in the request ***context***. The XPath expression SHOULD resolve to an ***attribute*** in the `<Resource>` element of the request ***context***.
   - **5.14 Element <ActionMatch>**: `<AttributeSelector>`[Required Choice]

MAY be used to identify one or more ***attribute*** values in the request ***context***. The XPath expression SHOULD resolve to an ***attribute*** in the `<Environment>` element of the request ***context***.

- **5.42 Element <AttributeSelector>**: The `<AttributeSelector>` element's `RequestContextPath` XML attribute SHALL contain a legal XPath expression whose context node is the `<xacml-context:Request>` element. The `AttributeSelector` element SHALL evaluate to a ***bag*** of values whose data-type is specified by the element's `DataType` attribute. If the `DataType` specified in the `AttributeSelector` is a primitive data type defined in **[XF]** or **[XS]**, then the value returned by the XPath expression SHALL be converted to the `DataType` specified in the `AttributeSelector` using the constructor function below [**XF** Section 4] that corresponds to the `DataType`....Each node selected by the specified XPath expression MUST be either a text node, an attribute node, a processing instruction node or a comment node. The string representation of the value of each node MUST be converted to an ***attribute*** value of the specified data type, and the result of the `AttributeSelector` is the ***bag*** of the ***attribute*** values generated from all the selected nodes.
- **5.42 Element <AttributeSelector>**: `RequestContextPath`[Required]
  An XPath expression whose context node is the `<xacml-context:Request>` element. There SHALL be no restriction on the XPath syntax.
- **7.2.1 Structured Attributes**: An `<AttributeSelector>`element MAY be used to select the value of a leaf sub-element of the structured data-type by means of an XPath expression. That value MAY then be compared using one of the supported XACML functions appropriate for its primitive data-type. This method requires support by the ***PDP*** for the optional XPath expressions feature.
- **7.2.1 Structured Attributes**: An `<AttributeSelector>` element MAY be used to select the value of any node in the structured data-type by means of an XPath expression. This node MAY then be compared using one of the XPath-based functions described in Section . This method requires support by the ***PDP*** for the optional XPath expressions and XPath functions features.
- **7.2.2 Attribute Bags**: The `<AttributeSelector>` element uses an XPath expression to specify the selection of data from an XML ***resource***. The result of an XPath expression is termed a *node-set*, which contains all the leaf nodes from the XML ***resource*** that match the predicate in the XPath expression. Based on the various indexing functions provided in the XPath specification, it SHALL be implied that a resultant node-set is the collection of the matching nodes. XACML also defines the `<AttributeDesignator>` **element** to have the same matching methodology for ***attributes*** in the XACML request ***context***.
- **7.2.4 Attribute Matching**: In the case of an ***attribute*** selector, the matching of the ***attribute*** to the ***named attribute*** SHALL be governed by the XPath expression and `DataType`.
- **A.3.14 XPath-based functions**: This section specifies functions that take XPath expressions for arguments. An XPath expression evaluates to a *node-set*, which is a set of XML nodes that match the expression. A node or node-set is not in the formal data-type system of XACML. All comparison or other operations on node-sets are performed in isolation of the particular function specified. That is, the XPath expressions in these functions are restricted to the XACML request ***context***. The

`<xacml-context:Request>` element is the context node for every XPath expression.

- **A.3.14 XPath-based functions**: This function SHALL take [an] "http://www.w3.org/2001/XMLSchema#string" as [an] argument, which SHALL be interpreted as an XPath expression,

3.3.1.1 Rule target

[Should we say "All three options are supported in XACML or by XACML Profiles."?]

On the other hand, in the case of ***resource*** names and ***resources*** themselves, three options exist. The name could refer to:
1. the contents of the identified node only,
2. the contents of the identified node and the contents of its immediate child nodes or
3. the contents of the identified node and all its descendant nodes.

All three options are supported in XACML.

4.2.2 Example request context

```
[a1]                                    <Resource>
[a2]                                                    <ResourceContent>
[a3]
            <md:record xmlns:md="http:www.medco.com/shemas/record.xsd">
[a4]
                                                                    <md:patient>
[a5]

                    <md:patientDoB>1992-03-21</md:patientDoB>
[a6]
                                                                  </md:patient>
[a7]
                                  </md:record>
[a8]                                                          </ResourceContent>
[a9]                                        <Attribute
      AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
      DataType="xs:string">
[a10] [CHANGE above DataType="urn:oasis:names:tc:xacml:2.0:data-type:xpath-
      expression"
[a11]
                                  <AttributeValue>
[a12]
                                                        //medco.com/records/bart-
      simpson.xml#
[a13]  xmlns(md=http:www.medico.com/schemas/record.xsd)xpointer
      (/md:record/md:patient/md:patientDoB)
[a14] [REPLACE everything up through "xpointer" in the above <AttributeValue>
      with xacml-context:Resource/ResourceContent/]
[a15]
                                  </AttributeValue>
[a16]                                                             </Attribute>
[a17]                                       <Attribute
      AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
      DataType="xs:string">
[a18] [CHANGE above AttributeId to "...:resource:document-id"]
[a19] [CHANGE above DataType to "...#anyURI"]
[a20]
                                  <AttributeValue>
```

```
[a21]   xmlns(md=http:www.medico.com/schemas/record.xsd)xpointer
       (/md:record/md:patient/md:patientDoB)
[a22]  [CHANGE above AttributeValue to "http://www.medico.com/medical_records"]
[a23]
                                    </AttributeValue>
[a24]                                                          </Attribute>
[a25]                                  </Resource>
```

[CHANGE following paragraphs:

The identifier of the ***Resource*** instance for which access is requested.

The ***Resource*** type, for which access is requested, is identified using an Xpointer expression that names the URI of the resource type, its target namespace and the XPath location path to the requested element.

TO:]

The identifier of the ***Resource*** instance for which access is requested, which is an XPath expression into the <ResourceContent> element which selects the data to be accessed.

The ***Resource*** type, for which access is requested, is identified using the "document-id" ***attribute*** having a value that is the URI of the resource type.

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.

7.12 Hierarchical resources

[remove entire section]

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]
```