XML Message Encoding for KMIP

Hal Lockhart
Problem Statement

• KMIP 1.0 defines a single message encoding format called TTLV
• It is well suited for simple and efficient processing, especially in resource-constrained environments
• An equivalent XML encoding may be valuable in environments where most messages are already in XML
Design Goals

• Both encodings should convey the same semantics
• All differences between encodings should be confined to one place in the spec (Chapter 9)
• Changes to other parts of the specs should automatically work with both encodings
• TTLV -> XML, XML -> TTLV and round tripping should all work
• Should be simple to build Server which supports both
• Should be simple to build bi-directional gateway
• This implies no information loss in moving from one form to the other
KMIP XML Conversion Principles

- Use non-empty XML Elements to represent KMIP Objects
- Use the current names given in KMIP spec as basis for naming in XML
  - Remove spaces between words
  - Use “upper camel case” e.g. KeyWrappingData
  - Dash character is ok as is, e.g. CommonTemplate-Attribute
  - Need to map “/” to something else e.g. MAC/Sigature, perhaps dash, e.g. MAC-Sigature?
  - Single letter names might cause confusion, e.g. P, Q, X, Y
    - Perhaps CryptoP, CryptoQ, etc.?
Data Type

- In TTLV data type is passed with object
- I assume it is only used to interpret Value
- In XML data type is in Schema
- Parser needs to have access to Schema
## KMIP Data Types

<table>
<thead>
<tr>
<th>KMIP Data Type</th>
<th>XML Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integer</td>
<td>XML Schema int</td>
</tr>
<tr>
<td>Long Integer</td>
<td>XML Schema long</td>
</tr>
<tr>
<td>Big Integer</td>
<td>XML Signature CryptoBinary</td>
</tr>
<tr>
<td>Enumeration</td>
<td>XML Schema string with enumeration</td>
</tr>
<tr>
<td>Boolean</td>
<td>XML Schema boolean</td>
</tr>
<tr>
<td>Text String</td>
<td>XML Schema string</td>
</tr>
<tr>
<td>Byte String</td>
<td>XML Schema base64Binary</td>
</tr>
<tr>
<td>Date-Time</td>
<td>XML Schema dateTime ?</td>
</tr>
<tr>
<td>Interval</td>
<td>XML Schema duration (constrained)</td>
</tr>
<tr>
<td>Structure</td>
<td>XML Schema Derived Type</td>
</tr>
</tbody>
</table>
Data Type Notes

• CryptoBinary is defined in the W3C XML Signature schema. Should we import or copy it?
• dateTime is what you would expect. Years, months, days, hours, minutes, seconds. Intuitive to human readers
  – For compatibility we can prohibit fractional seconds
  – Libraries for conversion to and from POSIX Time
  – Alternatively, express it as a duration from 1 Jan 1970
• duration can be expressed in any combination of time units.
  – Prohibit fractional seconds
  – Allow only hours, minutes, seconds
  – Alternatively, only seconds as in TTLV
XML Encoding Approach 1

<Attribute>
  <AttributeName> Unique Identifier </AttributeName>
  <AttributeIndex> 0 </AttributeIndex>
  <AttributeValue> OBID-123456789 </AttributeValue>
</Attribute>
XML Encoding Approach 2

<Object name="Attribute">
    <Object name="AttributeName">
        Unique Identifier </Object>
    <Object name="AttributeIndex"> 0 </Object>
    <Object name="AttributeValue">
        OBID-1234567789 </Object>
</Object>