Abstract:
This document defines an HTTP-based resolution mechanism for Extensible Resource Identifiers (XRIs), specifically XRIs conforming to the XRI Generic Syntax Specification v2.0 [link] or higher. For a non-normative introduction to the uses and features of XRIs, see the Introduction to XRIs at http://www.oasis-open.org/committees/xri/xri-primer-2.0.
For the set of XRIs defined by the XRI TC to provide metadata about other XRIs, see the XRI Metadata Specification v2.0 at @@[link].

Status:
This document is a Working Draft. This document is updated periodically on no particular schedule. Send comments to the editor.
Committee members should send comments on this specification to the xri@lists.oasis-open.org list. Others should subscribe to and send comments to the xri-comment@lists.oasis-open.org list. To subscribe, send an email message to xri-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 XRI TC web page (http://www.oasis-open.org/committees/xri).
The errata page for this specification is at http://www.oasis-open.org/committees/xxx/yyy.
# Table of Contents

1 Introduction .......................................................................................................................... 4  
2 Generic Resolution .............................................................................................................. 5  
3 Trusted Resolution ............................................................................................................. 26  
4 Extensibility and Versioning .............................................................................................. 37  
5 Other HTTP Features ........................................................................................................... 24  
6 HTTP Headers ...................................................................................................................... 24  
7 Caching and Efficiency ......................................................................................................... 24  
8 Local Access Service Types ................................................................................................. 22  
9 The X2R Local Access Service ............................................................................................ 22  
10 XRI Authority Resolution .................................................................................................... 7  
11 XRI Authority Descriptors ................................................................................................... 8  
12 Starting the Chain of XRI Descriptors with the Root XRID .............................................. 11  
13 Default HTTP(S)-based Authority Resolution Service .................................................... 11  
14 Examples (Non-Normative) ............................................................................................... 15  
15 Resolving Cross-References in XRI Authorities .................................................................. 19  
16 XRI Redirects ...................................................................................................................... 20  
17 IRI Authority Resolution .................................................................................................... 21  
18 IRI Authority Descriptors ................................................................................................... 7  
19 Phases of Resolution ........................................................................................................... 5  
20 Assumptions ....................................................................................................................... 5  
21 XRI Metadata Reserved for XRI Resolution ....................................................................... 6  
22 XRI vs. IRI Authorities ...................................................................................................... 6  
23 Server Behavior .................................................................................................................. 35  
24 Client Behavior .................................................................................................................. 33  
25 Additional Requirements of Authorities Offering Trusted Resolution .......................... 36  
26 Use and Correlation of AuthorityID Elements ................................................................... 33  
27 Extensibility and Notation ................................................................................................. 4  
28 Terminology and Notation ................................................................................................. 4  
29 Reader’s Guide .................................................................................................................... 4  
30 Generic Resolution ............................................................................................................. 5  
31 Introduction ......................................................................................................................... 5  
32 Overview and Example (Non-Normative) ......................................................................... 26  
33 Trusted Resolution Protocol .............................................................................................. 32  
34 XML Elements and Attributes ........................................................................................... 32  
35 Client Behavior .................................................................................................................. 33  
36 Server Behavior .................................................................................................................. 35  
37 Additional Requirements of Authorities Offering Trusted Resolution .......................... 36  
38 Versioning ........................................................................................................................... 38  
39 Specific Points of Extensibility ........................................................................................... 37  
40 Versioning ........................................................................................................................... 38  
41 Versioning of the XRI Resolution Specification ................................................................. 38  
42 Versioning of the XRI Resolution Specification ................................................................. 38  
43 Specific Points of Extensibility ........................................................................................... 37  
44 Versioning ........................................................................................................................... 38  
45 Versioning of the XRI Resolution Specification ................................................................. 38
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.2</td>
<td>Versioning of XRI Descriptor Elements</td>
<td>38</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Versioning of Protocols</td>
<td>39</td>
</tr>
<tr>
<td>5</td>
<td>Security and Data Protection</td>
<td>40</td>
</tr>
<tr>
<td>5.1</td>
<td>DNS Spoofing</td>
<td>40</td>
</tr>
<tr>
<td>5.2</td>
<td>HTTP Security</td>
<td>40</td>
</tr>
<tr>
<td>5.3</td>
<td>Caching Authorities</td>
<td>40</td>
</tr>
<tr>
<td>5.4</td>
<td>Lookahead and Proxy Resolution</td>
<td>40</td>
</tr>
<tr>
<td>5.5</td>
<td>SAML Considerations</td>
<td>40</td>
</tr>
<tr>
<td>5.6</td>
<td>Community Root Authorities</td>
<td>40</td>
</tr>
<tr>
<td>5.7</td>
<td>DOS Attacks</td>
<td>41</td>
</tr>
<tr>
<td>5.8</td>
<td>Limitations of Trusted Resolution</td>
<td>41</td>
</tr>
<tr>
<td>5.9</td>
<td>Media Type Registration for application/xrid+xml</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>Media Type Registration for application/xrid-t-saml+xml</td>
<td>43</td>
</tr>
<tr>
<td>8</td>
<td>References</td>
<td>44</td>
</tr>
<tr>
<td>8.1</td>
<td>Normative</td>
<td>44</td>
</tr>
<tr>
<td>8.2</td>
<td>Informative</td>
<td>45</td>
</tr>
<tr>
<td>9</td>
<td>Appendix A. Revision History</td>
<td>47</td>
</tr>
<tr>
<td>9.1</td>
<td>Appendix B. XML Schema for XRI Descriptor (Normative)</td>
<td>48</td>
</tr>
<tr>
<td>9.2</td>
<td>Appendix C. RelaxNG Compact Syntax Schema for XRI Descriptor (Non-normative)</td>
<td>51</td>
</tr>
<tr>
<td>9.3</td>
<td>Appendix D. Notices</td>
<td>54</td>
</tr>
<tr>
<td>9.4</td>
<td>Appendix E. LAST MINUTE TODOs</td>
<td>55</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 XRI Resolution Framework

Extensible Resource Identifiers (XRIs) provide a uniform syntax for abstract identifiers as defined in [XRISyntax]. Because XRIs may be used across a wide variety of communities and applications (as database keys, filenames, directory keys, object IDs, XML IDs etc.), no single resolution mechanism may prove appropriate for all XRIs. However, in the interest of promoting interoperability, this specification defines a standard framework for XRI resolution consisting of two parts:

- **Generic resolution** (section 2) is a simple, flexible resolution protocol for the authority segment of an XRI that relies exclusively on HTTP/HTTPS as a transport.
- **Trusted resolution** (section 3) is an extension of the generic resolution protocol that uses SAML assertions to create a chain of trust between the participating authorities.

Both of these protocols are extensible as described in section 4. In addition, other XRI resolution services or protocols may be defined by future versions of this specification or by other specifications.

1.2 General Format and Reader’s Guide

[TODO – Drummond]

1.3 Terminology and Notation

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “NOT RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119]. When these words are not capitalized in this document, they are meant in their natural language sense.

Examples look like this.

XML elements and attributes that appear in text look like this.

Throughout this document, the XML namespace prefix saml: stands for the Security Assertion Markup Language [SAML] namespace “urn:oasis:names:tc:SAML:2.0:assertion” whether or not it is explicitly declared in the example or text. Similarly, the XML namespace prefix ds: stands for the W3C Digital Signature [XMLDSig] Namespace “http://www.w3.org/2000/09/xmldsig#”, and the namespace prefix xrid: stands for the namespace “xri://$res*/schema/XRIDescriptor*($v%2F2.0)”, whether or not they are explicitly declared in the example or text. These namespace prefixes are summarized in Table 1.

<table>
<thead>
<tr>
<th>Namespace</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>saml</td>
<td>urn:oasis:names:tc:SAML:2.0:assertion</td>
</tr>
<tr>
<td>ds</td>
<td><a href="http://www.w3.org/2000/09/xmldsig#">http://www.w3.org/2000/09/xmldsig#</a></td>
</tr>
<tr>
<td>xrid</td>
<td>xri://$res*/schema/XRIDescriptor*($v%2F2.0)</td>
</tr>
</tbody>
</table>

Table 1: XML namespace prefixes used in this specification.

Terms used in this document are defined in the glossary in Appendix @@@ of [XRISyntax].
2 Generic Resolution

2.1 Introduction

Generic XRI resolution is the process of dereferencing an XRI to a network endpoint in order to obtain metadata about or further interact with the resource identified by the XRI. To provide a simple, general-purpose mechanism for accomplishing this task, this specification defines a generic resolution protocol based on HTTP/HTTPS. Other XRI resolution services may be defined by future versions of this specification or by other specifications.

Identifier management policies are defined on a community-by-community basis. With XRIs, the resolution community is specified by the authority segment of the XRI. When a resolution community chooses to create a new identifier authority, it SHOULD define a policy for assigning and managing identifiers under this authority. Furthermore, it SHOULD define what resolution protocol(s) can be used for resolving identifiers assigned by the authority.

2.1.1 Assumptions

The generic resolution protocol makes several minimal assumptions about the XRIs being resolved:

- The endpoints representing the top-level authority for any absolute XRI are identified with the authority segment (xri-authority or i-authority productions) of the XRI as defined in section 2.2.1 of [XRISyntax].
- Only absolute XRIs are resolved using this protocol. To resolve a relative XRI reference, it must be converted into an absolute XRI using the procedure defined in section 2.4 of [XRISyntax].
- The XRI being resolved has been converted into URI-normal form, following the rules in section 2.3.1 of [XRISyntax].
- A resource represented by a single XRI may be accessed by multiple protocols at multiple protocol endpoints. For example, it is possible that a resource represented by a single XRI may be accessed through multiple HTTP URIs, or through both HTTP and another network protocol. Only HTTP access to resources is defined in this specification, however a mechanism for specifying access via URIs using other URI schemes is also defined.
- Each network endpoint associated with a resource identified by an XRI may present a different subset, type, or representation of data or metadata associated with the identified resource. For example, two HTTP URIs may be associated with a single XRI, one for data access and one for metadata access. This specification allows XRI authorities to define different types of access using extensible descriptor fields based on content type and the semantics of the interaction.

2.1.2 Phases of Resolution

The generic resolution protocol is designed to be as simple and flexible as possible given the assumptions above. Based on the structure of XRIs, it consists of two phases:

- Authority resolution
- Local access

Authority resolution is the process of a) finding the endpoint or endpoints that are authoritative for access to resources under that authority’s control or b) discovering further information about the authority itself. In the case where the desired goal is access to a resource, the result of authority resolution will be a list of local access endpoints, identified by one or more URIs, that support...
least one local access protocol. The calling application may then choose one of these endpoints and access it using the desired local access protocol.

In the case where the goal of resolution is to discover more information about an authority, such as XRI synonyms, public keys, or other XRI resolution metadata, this information will be returned by the authority resolution process itself.

Figure 1 illustrates the two main phases of XRI resolution: authority resolution and local access.

Figure 1: Phases of Resolution

2.1.3 XRI vs. IRI Authorities

As described in section 2.2.1 of [XRISyntax], XRI authorities and IRI authorities have different syntactic structures, partially due to the higher layer of abstraction represented by XRI authorities. For this reason, XRI authorities are resolved into authority descriptor documents sub-segment by sub-segment as described in section 2.2. IRI authorities, since they are based on DNS names or IP addresses, are resolved into an authority descriptor through a special HTTP(S) request based on the DNS name or IP address identified by the IRI authority segment.

2.1.4 XRI Metadata Reserved for XRI Resolution

As defined in section 2.2.1.2 of [XRISyntax], the GCS symbol "$" is reserved for special identifiers assigned by XRI TC specifications, other OASIS specifications, or other standards bodies. (See also [XRIMetadata] .) Within the "$" namespace, the identifier "$res" is reserved for identifiers assigned by this XRI resolution specification. Table 2 summarizes these identifiers.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Use</th>
<th>See Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>xri://$res*schema</td>
<td>XML namespace for XRI resolution schema</td>
<td>0</td>
</tr>
<tr>
<td>xri://$res*auth.res</td>
<td>Namespace for authority resolution protocol</td>
<td>2.2.4</td>
</tr>
<tr>
<td>xri://$res*local.access</td>
<td>Namespace for local access protocol types</td>
<td>2.4.1</td>
</tr>
<tr>
<td>xri://$res*trusted</td>
<td>Namespace for trust mechanisms</td>
<td>0 and 3</td>
</tr>
</tbody>
</table>

Comment [gmw8]: Section references have been mangled somehow – they need to be fixed, which may be a lot of work.

Deleted: URI vs.
Deleted: URI/
Deleted: IRI authorities
Deleted: and
Deleted: URI/
Deleted: to describe XRIs, XRI resolution, or other special characteristics of XRI-identified resources
Deleted: $s
Deleted: authres
Deleted: $res*localaccess
Deleted: xri://$res*types
2.2 XRI Authority Resolution

2.2.1 Overview

XRI authority resolution is an iterative process that resolves the qualified sub-segments within the XRI authority segment from left to right. A qualified sub-segment is either: a) a global context symbol as defined by section 2.2.1.2 of [XRISyntax] or b) a sub-segment as defined by the productions whose names start with "xri-subseg" in section 2.2.3 of [XRISyntax] including the leading syntactic delimiter ("*" or "!"). Note that in the latter case a qualified sub-segment always includes the leading syntactic delimiter even if it was optionally omitted in the original XRI (see the productions names containing "-od" in section 2.2.3 of [XRISyntax]).

Each qualified sub-segment is resolved in the context of the sub-segment immediately to the left, with the first (or leftmost) sub-segment identifying the root of the identifier community. Each sub-segment is resolved to a corresponding XRI Descriptor (often abbreviated as "XRID"), an XML document that specifies one or more network endpoints (HTTP or HTTPS URIs) that answer XRI resolution requests. As resolution proceeds, the XRI resolver is building a "chain" (an ordered list) of XRIDs for all sub-segments in the XRI authority segment. Figure 2 below depicts this resolution process for the XRI "@a*b*c".

Figure 2: XRI Descriptors, XRI Authorities and Sub-Segments for @a*b*c

Each HTTP(S) resolution request may ask for resolution of more than one sub-segment at a time—a process called lookahead resolution. If lookahead resolution is used, each HTTP(S) response may contain one or more XRIDescriptors inside a XML container document. The number of sub-segments resolved in one resolution request depends on:

- How many sub-segments the resolving client presents to a responding XRI Authority for lookahead resolution; and
• Configuration, policy, and state of the responding XRI Authority (e.g. previously cached requests).

Each XRI Descriptor in the chain contains one or more of three basic types of information about the XRI authority it describes:

• URIs describing network endpoints for XRI authority resolution services;
• URIs describing network endpoints for local access services;
• XRI synonyms (equivalent XRIs) for the resolved sub-segment.

All three options—authority resolution, local access, or synonyms—may be available at each step of resolution. For example, the XRI authority identifier "@a*b*c" may be the prefix to another XRI authority with the XRI "@a*b*c*d." "@a*b*c" may also be a local access endpoint itself, in which case its XRI Descriptor will contain references to local access services. Finally, "@a*b*c" may have a persistent XRI synonym such as "xri://@1000123/", which may also be included in the XRID in order to provide resolvers or caches with an equivalent persistent XRI.

2.2.2 XRI Descriptors

To provide a straightforward, flexible resolution mechanism, XRI authority endpoints are described using a simple, flexible XML document, called an XRI Descriptor (often abbreviated "XRID"). While this specification defines only XRID elements necessary to support delegated resolution and access of XRI-identified authorities and resources, an XRID can easily be extended to publish any form of metadata about the described authority.

The formal XML Schema definition of an XRI Descriptor is provided in Appendix B. The following example instance document illustrates the fields defined in this schema:

```xml
<XRIDescriptors xmlns="xri://$res*schema/XRIDescriptor*($v%2F2.0)"/>
<XRIDescriptor xrid:id="first">
  <Resolved>x</Resolved>
  <AuthorityID>urn:uuid:c9f812f3-6544-4e3c-874e-d3ae79f4ef7b</AuthorityID>
  <Expires>2002-05-30T09:30:10-06:00</Expires>
  <Authority>
    <AuthorityID>urn:uuid:f0502a17-4503-4463-8516-f1225b330e4d</AuthorityID>
    <Type>xri://$res*auth.res/XRIA</Type>
    <URI>http://xri.example.com</URI>
    <URI>https://xri.example.com</URI>
  </Authority>
  <Service>
    <Type>xri://$res*local.access/X2R</Type>
    <URI>http://xri.example.com</URI>
    <MediaType>application/rdf+xml</MediaType>
  </Service>
  <Service>
    <Type>xri://$res*local.access/X2R</Type>
    <URI>http://pictures.xri.example.com</URI>
    <MediaType>image/jpeg</MediaType>
  </Service>
  <Synonyms>
    <Internal>xri://@!1!2!3</Internal>
    <External>xri://@!4!5!6</External>
  </Synonyms>
  <TrustMechanism>xri://$res*trusted/None</TrustMechanism>
</XRIDescriptor>
```
Other XRI Descriptor elements here
</XRIDescriptor>

All schema elements in the basic XML Descriptor are in the XML namespace
<xri://$res*schema/XRIDescriptor"{SV'2F2.0}". Following are the elements and attributes that
comprise the XRI Descriptor document type (all XPATHs are relative to the enclosing
XRIDescriptor document element):

- **xrid:**
  - **xrid:**
    - **xrid:**
      - **xrid:**
        - **xrid:**
          - **xrid:**
            - **xrid:**
              - **xrid:**
                - **xrid:**
                  - **xrid:**
                    - **xrid:**
                      - **xrid:**
                        - **xrid:**
                          - **xrid:**
                            - **xrid:**
                              - **xrid:**
                                - **xrid:**
                                  - **xrid:**
                                    - **xrid:**
                                      - **xrid:**
                                        - **xrid:**
                                          - **xrid:**
                                            - **xrid:**
                                              - **xrid:**
                                                - **xrid:**
                                                  - **xrid:**
                                                    - **xrid:**
                                                      - **xrid:**
                                                        - **xrid:**
                                                          - **xrid:**
                                                            - **xrid:**
                                                              - **xrid:**
                                                                - **xrid:**
                                                                  - **xrid:**
                                                                    - **xrid:**
                                                                      - **xrid:**
                                                                        - **xrid:**
                                                                            - **xrid:**
                                                                                - **xrid:**
                                                                                    - **xrid:**
                                                                                        - **xrid:**
                                                                                            - **xrid:**
                                                                                                - **xrid:**
                                                                                                    - **xrid:**
                                                                                                      - **xrid:**
                                                                                                       - **xrid:**
                                                                                                        - **xrid:**
                                                                                                         - **xrid:**
                                                                                                          - **xrid:**
                                                                                                           - **xrid:**
                                                                                                              - **xrid:**
                                                                                                               - **xrid:**
                                                                                                                  - **xrid:**
                                                                                                                     - **xrid:**
                                                                                                                       - **xrid:**
                                                                                                                          - **xrid:**
                                                                                                                             - **xrid:**
                                                                                                                                - **xrid:**
                                                                                                                                   - **xrid:**
                                                                                                                                      - **xrid:**
                                                                                                                                        - **xrid:**
                                                                                                                                             - **xrid:**
                                                                                                                                                 - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xrid:**
                                                                                                                                                    - **xri...
this URI MUST be an HTTP or HTTPS URI. Future versions of this specification or other specifications may specify other transport protocols. Each URI element has an optional attribute called "trusted" that indicates whether or not the particular service endpoint provides trusted resolution (section 3). The trust mechanism is described using the TrustMechanism element (below).

### xrid:XRIDescriptor/xrid:Service

0 or more. Describes a local access service endpoint, provided by the described authority.

### xrid:XRIDescriptor/xrid:Service/xrid:Type

0 or 1 per Service element. Indicates the type of local service being described. This specification defines one service: "xri://$res*local.access/X2R" (The X2R local access service as defined in section 2.4.2). This is the default value if this element is not present.

### xrid:XRIDescriptor/xrid:Service/xrid:URI

1 or more per Service element. Indicates the transport level URI where the service described may be accessed. For the X2R local access service defined in section 2.4.2, this URI MUST be an HTTP or HTTPS URI. Other services may use other transport protocols.

### xrid:XRIDescriptor/xrid:Service/xrid:MediaType

0 or more per Service element. The media type of content available at this service. If this element is not present, then no assumption can be made about the type of data available at this endpoint. The value of this element must be of the form of a media type defined in [RFC2046]. This element may appear multiple times to indicate that multiple media types are available through this local access service.

### xrid:XRIDescriptor/xrid:Synonyms

0 or 1. Contains statements about the equivalence of the resolved identifier to other XRIs.

### xrid:XRIDescriptor/xrid:Synonyms/xrid:Internal

0 or more. Represents another XRI assigned to the described authority by the current describing authority. Must be an absolute XRI ("absolute-xri" in section 2.2 of [XRISyntax]). An internal synonym may be used, for example, to assert that a XRI authority known by a reassignable XRI may also be known by one or more persistent XRIs, or by a different reassignable XRI than the one being resolved. Both cases may be particularly useful in populating or querying a cache, since resolution of an internal synonym will typically result in an XRID containing the same information as the current XRID.

### xrid:XRIDescriptor/xrid:Synonyms/xrid:External

0 or more. Represents another XRI assigned to the described authority by a different authority, than the current describing authority. Must be an absolute XRI ("absolute-xri" in section 2.2 of [XRISyntax]). Resolution of an external synonym will typically result in an XRID containing different information than the current XRID. External synonyms are used, for example, in XRI redirects, described in Section 2.2.7. They can also be used to identify alternative sources of local access descriptors if those in the current XRID do not satisfy the needs of the client.

### xrid:XRIDescriptor/xrid:TrustMechanism

0 or 1. Identifies the mechanism for trusted resolution associated with this XRID. This specification defines two values: "xri://$res*trusted/XRITrusted" (for Trusted Resolution as described in section 3) and "xri://$res*trusted/None" (for generic resolution as described here in section 2). If this element does not appear, the default value is "xri://$res*trusted/None".
XRI Descriptor documents have an “open schema” that allows other elements and attributes from other namespaces to be added throughout. These points of extensibility can be used to deploy new identifier authority resolution schemes, new local access resolution schemes, additional XRI synonym metadata, or other metadata about the described authority. See section 4.1 for more about XRID extensibility.

See section 3.3.1 for additional XRI Descriptor elements defined for trusted resolution.

### 2.2.3 Starting the Chain of XRI Descriptors with the Root XRID

With an XRI authority, the first sub-segment corresponding to the community root may be either a global context symbol (GCS) character or top-level cross-reference as specified in section 2.2.1 of [XSIRSyntax]. In either case, the corresponding root XRID (or its equivalent) specifies the top-level authority resolution endpoints for that community. The root XRID is known a priori and is part of the configuration of a resolver, similar to the configuration of root DNS servers in a DNS resolver. (Note that it is not strictly necessary to publish this information in an XRID—it may be supplied in any format that enables configuration of the XRI resolvers in the community—but providing an XRID at a known location simplifies the process.)

If the first sub-segment of an XRI authority is a GCS character and the following sub-segment does not begin with a ‘*’ (indicating a reassignable sub-segment) or a ‘!’ (indicating a persistent sub-segment), then an '*' is implied and must be added when constructing the qualified sub-segment. Table 3 and Table 4 illustrate the differences between parsing a reassignable sub-segment following a GCS character and a cross-reference, respectively.

<table>
<thead>
<tr>
<th>XRI</th>
<th>xri://@example*internal/foo</th>
</tr>
</thead>
<tbody>
<tr>
<td>XRI Authority</td>
<td>@example*internal</td>
</tr>
<tr>
<td>Community Root Authority</td>
<td>@</td>
</tr>
<tr>
<td>First Qualified Sub-Segment Resolved</td>
<td>*example</td>
</tr>
</tbody>
</table>

Table 3: Parsing the first sub-segment of an XRI that begins with a global context symbol.

<table>
<thead>
<tr>
<th>XRI</th>
<th>xri://(<a href="http://www.example.com)*internal/foo">http://www.example.com)*internal/foo</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>XRI Authority</td>
<td>(<a href="http://www.example.com)*internal">http://www.example.com)*internal</a></td>
</tr>
<tr>
<td>Community Root Authority</td>
<td>(<a href="http://www.example.com">http://www.example.com</a>)</td>
</tr>
<tr>
<td>First Qualified Sub-Segment Resolved</td>
<td>*internal</td>
</tr>
</tbody>
</table>

Table 4: Parsing the first sub-segment of an XRI that begins with a cross-reference.

### 2.2.4 Default HTTP(S)-based Authority Resolution Service

This section defines the default authority resolution service for generic XRI resolution. When explicitly declared, it uses the xri:XRIDescriptor/xri:Authority/xri:Type element value “xri://@res*auth.res/XRIA”.

Generic XRI authority resolution service allows a client to request resolution of multiple authority sub-segments in one transaction (“lookahead” resolution). If a client makes such a request, the responding authority MAY perform the additional lookahead resolution steps requested. In this case the responding authority will act as a client to the other authorities that need to be queried for the lookahead segments.
If an authority performs lookahead resolution, it **MUST** return an ordered list of `xrid:XRIDescriptor` elements in an XRIDescriptors document. Each XRI Descriptor **MUST** correspond to a sub-segment resolved by the authority on behalf of the resolving client. The list of `xrid:XRIDescriptor` elements in the XRIDescriptors document **MUST** appear in the same order as the sub-segments in the original request. The responding authority **MAY** resolve fewer sub-segments than requested by the client. The responding authority is under no obligation to resolve more than the first sub-segment (for which it is authoritative.)

```
Resolve @a*b*c

Figure 3: Lookahead Resolution

Figure 3 illustrates a resolving client requesting lookahead resolution for the XRI authority "@a*b*c". The "@" authority is willing to resolve "@a*b" on behalf of the resolving client. The "@" authority can accomplish this either by acting as a resolving client itself, or by examining a cache it may have built through previous resolutions. In this example, the "@" authority is only willing or able to resolve the descriptor for "*a" (for which it is authoritative) plus "@a*b". Therefore the resolving client must resolve "*c" itself. The resolving client will know the "@" authority only resolved two segments ("a" and "b") because it only returned two XRI Descriptors corresponding to those two sub-segments.
```

If the **responding** authority does not resolve the entire set of sub-segments presented, the resolving client **MUST** continue the authority resolution process itself. At any stage, however, the resolving client **MAY** request that the next authority resolve any additional unresolved sub-segments. For example, in Figure 3, if the "@" authority had refused to do any lookahead, the resolving client could have asked the "@a" authority to resolve the unresolved "*b*c" portion of the XRI authority segment.

2.2.4.1 Determining the URI for the Next Resolution Step

Before each authority resolution step is performed, a URI must be constructed for the next HTTP(S) request. This URI establishes the context of that authority. Initially the current context is the root authority, and the current context shifts to a **new** authority each time a resolution step is performed. After a lookahead resolution request, the current context is the last authority whose sub-segment was resolved by the authority performing the lookahead request.

This "Next Authority URI" is constructed from two strings:

```
- The contents of the `xrid:XRIDescriptor/xrid:Authority/xrid:URI` element extracted from the XRI Descriptor corresponding to the current context, and
```

Deleted: the ...the authority resolver...will ...
Deleted: Authority ...with Loc ...
Deleted: Authority identifier ...an ...the entire unresolved portion... of the XRI Authority ...
Deleted: "*
Deleted: e...Resolution ...pieces of information ...
Deleted: XRI Authority identifier ...an ...the entire unresolved portion... of the XRI Authority ...
Deleted: xrid:XRIDescriptor/xrid:Authority/xrid:URI element
• The next qualified sub-segment to be resolved. (Note that this sub-segment must begin with an XRI syntax delimiter, i.e., "#" or "@"—see section 2.2.6.)

If the path portion of the first URI does not end with a "/" character, one must be appended before proceeding. Then the URI-normal form (section 2.3.1 of [XRISyntax]) of the next qualified sub-segment being resolved is appended to the path portion of this URI. For example, when resolving the "c" sub-segment of "xri://@a"b"c"@", if the XRI Authority URI resulting from the resolution of "xri://@a"b" is "http://example.com/xri-authority/", then the Next Authority URI is the concatenation of "http://example.com/xri-authority/" with "c", yielding "http://example.com/xri-authority/"c". An HTTP GET request is made to this URI, and the XRID for the context "xri://@a"b"c" is retrieved.

In lookahead resolution (Figure 3), any portion of the remaining XRI authority segment may be appended. For example, if the resolving client wanted to resolve "c"d", it would append this entire string to "http://example.com/xri-authority/" yielding "http://example.com/xri-authority/"c"d". Construction of the Next Authority URI is more formally described in this pseudo-code for resolving a "sub-segment-list" via an HTTP URI called "xa-uri":

\[ xa-uri = xri-authority-uri \]
\[ \text{if (path portion of xa-uri doesn't end in "/")} \]
\[ \quad \text{append "/" to path portion of xa-uri} \]
\[ \text{if (sub-segment-list isn't preceded with "+" or "/!" separator):} \]
\[ \quad \text{prepend "+" to sub-segment-list} \]
\[ \quad \text{append uri-escape(sub-segment-list) to path portion of xa-uri} \]

### 2.2.2.2 Making HTTP(S) Resolution Requests

Once the Next Authority URI is constructed, an HTTP or HTTPS GET request is made using this URI. Each GET request results in either a 2XX or 304 HTTP response. The HTTP request SHOULD contain an Accept: header with the value of "application/xrd+xml". See section 3.3.3 for a different value that may appear in the Accept header during trusted resolution.

The HTTP/HTTPS response for a successful resolution MUST contain either: a) a 2XX response with an XRI Descriptors document containing a list of one or more \( \text{xri:XRIDescriptor} \) elements, or b) a 304 response signifying that the cached version on the client is still valid (depending on the client's HTTP request). HTTP caching semantics should be leveraged as much as possible to support the efficiency and scalability of this HTTP-based resolution system. The recommended use of HTTP caching headers is described in more detail in section 2.5.1.

Any ultimate response besides an HTTP 2XX or 304 SHOULD be considered an error in the resolution process. There is no restriction on intermediate redirects (i.e., 3XX result codes) or other result codes (e.g., a 100 HTTP response) that eventually result in a 2XX or 304 response through normal operation of [RFC2616]. Ultimately, the content of a successful response will be an XRI Descriptors document containing one or more \( \text{xri:XRIDescriptor} \) elements for the qualified sub-segment[s] being resolved.

If there are no more sub-segments, the final context (as described by the final XRID retrieved) can be used for local access services as described in section 2.4, or to obtain synonyms or other metadata about the final authority.

### 2.2.3 Proxied Resolution

In some cases it may be desirable for an HTTP(S) server to do fully proxied XRI resolution on behalf of a client. While this is very similar to lookahead resolution, it differs in that a lookahead resolution request is always sent to the authority for the first sub-segment in the resolution process.
request. A proxied resolution request, in contrast, may be sent to any XRI proxy server that will accept the request.

The proxy resolution service is very simple: an HTTP GET is performed on a URI constructed by concatenating the base URI for the proxy resolution service and the XRI authority segment for which proxy resolution is being requested. As with standard resolution: a) this XRI authority segment MUST be in URI-normal form, and b) if the base proxy URI does not contain a trailing forward slash "/", one MUST be inserted between the base URI and the XRI authority segment. The proxy answering this request MUST perform XRI authority resolution as specified in this document and MUST return either: a) an XRI Descriptors document containing a entire chain of xri:X RIDescriptor elements for the segments of the authority it resolves, or b) an HTTP error code as described in section 2.2.4.4.

Note that because a proxy is not associated with any specific authority, proxied resolution requests MUST be comprised of authority segments starting with a GCS character or a cross-reference identifying a community root authority. In addition, a proxy resolver MUST return an XRI Descritor chain that begins with an XRID describing the community root authority. If the community root authority does not publish an XRID itself, a proxy MUST construct one from the equivalent information published by the community root authority.

The following example illustrates a proxied resolution request for "xri://=example*home*base". It assumes that the URI for a local proxy server is "http://proxy.example.com/xri-proxy". First the following HTTP GET request is made to "proxy.example.com":

```
GET /xri-proxy/=example*home*base HTTP/1.1
<other HTTP headers>
```

The proxy resolver then performs authority resolution, behaving as a resolving client as described in section 1. After completing this resolution process, the proxy resolver might produce the following HTTP response:

```
HTTP/1.1 200 OK
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
<other HTTP headers>

<xrid:xridDescriptors xmlns="...">
  <xrid:XRIDescriptor>
    <xrid:Resolved>=</xrid:Resolved>
    ...
  </xrid:XRIDescriptor>
  <xrid:XRIDescriptor>
    <xrid:Resolved>*example</xrid:Resolved>
    ...
  </xrid:XRIDescriptor>
  <xrid:XRIDescriptor>
    <xrid:Resolved>*home</xrid:Resolved>
    ...
  </xrid:XRIDescriptor>
  <xrid:XRIDescriptor>
    <xrid:Resolved>*base</xrid:Resolved>
  </xrid:XRIDescriptor>
  <xrid:Service>
    <xrid:Type>xri://$res*local.access/X2R</xrid:Type>
    <xrid:URI>
      http://xri.other.example.com/xri-local/base/
    </xrid:URI>
  </xrid:Service>
</xrid:xridDescriptors>
```

wd-xri-resolution-v2.0-wd05

Copyright © OASIS Open 2005. All Rights Reserved.
2.2.4.4 Errors During Proxied and Lookahead Authority Resolution

Proxies and lookahead resolvers SHOULD return any HTTP error codes returned during resolution back to the resolving client. For example, if during resolution on behalf of a client, a proxy is given back a 404 error code from an authoritative server, it should return 404 code to its client.

When encountering an HTTP error code, the proxy or lookahead resolver SHOULD return an XRIDescriptors document in the body of the HTTP error response. This XRIDescriptors document SHOULD contain the list of XRI Descriptor elements corresponding to the sub-segments successfully resolved or retrieved from cache. For example, if a proxy is asked to resolve @a*b*c, and successfully resolves @a*b, but receives a HTTP 404 on resolving *c, it should return an HTTP 404 response to its client that include xrid:XRIDescriptor elements for @a, and *b. This will indicate to the resolving client that *c is the sub-segment causing the 404 response.

This use of error codes, while slightly unusual, conforms to the requirements of [RFC2616], specifically sections 10.4 and 10.5 which state that "the server SHOULD include an entity containing an explanation of the error situation." The combination of the error code and the list of successfully resolved xrid:XRIDescriptor elements indicates to the client exactly which sub-segment caused the error. This should save both the client and the authority returning the error code an extra HTTP request/response cycle.

Even when given an HTTP error response, resolving clients SHOULD consider the xrid:XRIDescriptor elements returned in the content of the HTTP response as valid cacheable responses (if a client does caching in the first place). All other rules about XRI Descriptors (including those specified in Section 3 for trusted resolution) also apply.

2.2.5 Examples (Non-Normative)

2.2.5.1 Authority Resolution without Lookahead

Following is an example of resolving the authority portion of an XRI without lookahead resolution.

That is, for each resolution step, the resolving client requests resolution of only one authority sub-segment of the following XRI:

```
xri://example@home/base/foo@bar
```
This example assumes that the URI for the “=” global context symbol is http://equals.example.org/xri-resolve, found in xrid:XRIDescriptor/xrid:Authority/xrid:URI of the XRID for this community.

Resolving “=example”

The following HTTP request is made to “equals.example.org”:

```
GET /xri-resolve/*example HTTP/1.1
Accept: application/xrid+xml
```

The following HTTP response is received from “equals.example.org” (the content has changed since “Fri, 31 Oct 2003 19:43:31 GMT”, the value specified in the “If-Modified-Since” header):

```
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
```

```
<XRIDescriptors xmlns="…">
    <XRIDescriptor>
        <Resolved>*example</Resolved>
        <Authority>
            <URI>
                http://xri.example.com/xri-resolve/
            </URI>
        </Authority>
        <Service>…</Service>
    </XRIDescriptor>
</XRIDescriptors>
```

Resolving “=example*home”

Appending the next qualified sub-segment “*home” to the URI “http://xri.example.com/xri-resolve/” yields the URI “http://xri.example.com/xri-resolve/*home”, and the following HTTP request is made to xri.example.com:

```
GET /xri-resolve/*home HTTP/1.1
If-Modified-Since: Fri, 31 Oct 2003 19:43:32 GMT
Accept: application/xrid+xml
```

The following HTTP response is received from xri.example.com:
Resolving “=example*home*base”

Appending the next qualified sub-segment “*base” to the URI “http://xri.other.example.com/xri-resolve/*home/*base” gives the URI “http://xri.other.example.com/xri-resolve/*home/*base”:

```
GET /xri-resolve/*home/*base HTTP/1.1
If-Modified-Since: Fri, 31 Oct 2003 19:43:32 GMT
Accept: application/xrid+xml
<br>other HTTP headers>
```

The following HTTP response is received from xri.other.example.com:

```
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:33 GMT
<br>other HTTP headers>

<XRIDescriptors xmlns="...">
  <XRIDescriptor>
    <Resolved>"base"</Resolved>
    <Service>
      <Type>
        xri://$res*local.access/X2R
      </Type>
      <URI>
        http://xri.other.example.com/xri-local/base/
      </URI>
      <URI>
        https://xri.other.example.com/xri-local/base/
      </URI>
    </Service>
  </XRIDescriptor>
</XRIDescriptors>
```

The result of the final XRI authority resolution step is the set of HTTP and HTTPS URIs shown in the “Service” element above that can be used for local access services (specifically, the X2R local access service as identified by the xri://$res*local.access/X2R type).
The next example shows the interaction between a client and server using lookahead resolution when resolving the authority portion of the following XRI:

\[ xri://=example*home*base/foo*bar \]

Assume as in the previous example that the URI for the “=” global context symbol is “http://equals.example.org/xri-resolve”. In this example, the client will always request lookahead resolution of all unresolved authority sub-segments.

**Resolving “=example*home*base”**

The following HTTP request is made to “equals.example.org”:

```
GET /xri-resolve/*example*home*base HTTP/1.1
Accept: application/xrid+xml
<other HTTP headers>
```

The following HTTP response is received from “equals.example.org” (the content has changed since “Fri, 31 Oct 2003 19:43:31 GMT”, the value specified in the the “If-Modified-Since” header).

The response contains two XRIDescriptor elements, one for “=example” and one for “=home”. This indicates to the resolving client that the “equals.example.org” authority has either cached or performed its own resolution to retrieve the descriptor for =example*home:

```
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
<other HTTP headers>

<XRIDescriptors xmlns="…">
  <XRIDescriptor>
    <Resolved>*example</Resolved>
    <Authority>
      <URI>
        http://xri.example.com/xri-resolve/
      </URI>
    </Authority>
    <Service>…</Service>
  </XRIDescriptor>
  <XRIDescriptor xmlns="…">
    <Resolved>*home</Resolved>
    <Authority>
      <URI>
        http://xri.other.example.com/xri-resolve/*home/
      </URI>
    </Authority>
    <Service>…</Service>
  </XRIDescriptor>
</XRIDescriptors>
```

Note that the XRI Descriptor elements must appear in resolution order, i.e. the first XRI Descriptor describes the authority “=example” and the second describes the authority “=home” within the “=example” namespace.
The resolving client, assuming it trusts the resolver’s response (see section 3 for more details on trusted resolution), then resolves the “*base” authority sub-segment using the authority URI "http://xri.othersite.com/xri-resolve/*home/" as identified in the last XRI Descriptor above.

The following HTTP request is made to "xri.othersite.com":

```
GET /xri-resolve/*home/*base HTTP/1.1
Accept: application/xrid+xml
<other HTTP headers>
```

The following HTTP response is received from xri.othersite.com:

```
200 OK HTTP/1.1
Content-type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:33 GMT
<other HTTP headers>
```

```xml
<XRIDescriptors xmlns="…">
  <XRIDescriptor>
    <Resolved>*base</Resolved>
    <Service>
      <Type>xri://$res*local.access/X2R</Type>
      <URI>http://xri.othersite.com/xri-local/base/</URI>
      <URI>https://xri.othersite.com/xri-local/base/</URI>
    </Service>
  </XRIDescriptor>
  …
</XRIDescriptors>
```

Note that the three XRI Descriptor elements in this example (two from the first HTTP resolution from equals.example.org and the one from xri.othersite.com) are exactly the same three XRI Descriptors retrieved from the separate resolution requests showed in section 2.2.5.1.

### 2.2.6 Resolving Cross-References in XRI Authorities

A sub-segment within an XRI authority segment may be a cross-reference. Resolving a cross-reference is identical to resolving any other sub-segment because the cross-reference is considered opaque by generic XRI resolution. In other words, the value of the cross-reference (including the parentheses) is the literal value of the sub-segment for the purpose of authority resolution. The one exception is the special cross-reference for XRI annotations as defined in the XRI Metadata Specification [XRIMetadata]. This cross-reference begins with the GCS symbol "$" followed by the hyphen character "-". As specified in [XRIMetadata] such a cross-reference and the delimiter that precedes it MUST be ignored entirely during resolution.

Table 5 provides several examples of resolving cross-references. In each example, sub-segment "b" resolves to an XRI Authority URI of "http://example.com/xri-authority/" and lookahead resolution is not being requested.

| Cross-reference type | Example XRI | Next Resolution URI after resolving “xri://@a|b” |
|----------------------|-------------|-----------------------------------------------|
|                     |             |                                               |

wd-xri-resolution-v2.0-wd05

Copyright © OASIS Open 2005. All Rights Reserved.
### 2.2.7 XRI Redirects

An XRIDescriptor may contain an `xrid:XRIDescriptor/xrid:Synonyms/xrid:External` element and not contain any `xri:XRIDescriptor/xrid:Authority` or `xrid:XRIDescriptor/xrid:LocalAccess` elements. This is called an "XRI redirect" because the XRIDescriptor is effectively redirecting to a new XRI Authority. In this case, the unresolved portion of the original XRI (i.e. the XRI being resolved) is added to contents of the `/XRIDescriptor/Synonyms/External` element to create a new XRI. This new XRI is then resolved as described in Section 0 of this document.

The example in Section 2.2.5 demonstrates the resolution of `xri://=example*home*base/foo*bar`. The first request is to "equals.example.org". The following XRI redirect could be received as a response.

```xml
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT

<XRIDescriptors xmlns="...">
  <XRIDescriptor>
    <Resolved>*example</Resolved>
    <Synonyms>
      <External>
        xri://=example2
      </External>
    </Synonyms>
  </XRIDescriptor>
</XRIDescriptors>
```

In this case, a new XRI would be constructed as "xri://=example2*home*base/foo*bar" and the resolution process would begin again with this new XRI.

If the original XRI has additional sub-segments in the XRI Authority component and the `xrid:XRIDescriptor/xrid:Synonyms/xrid:External` element contains a local-path component, the client SHOULD consider this an error condition and fail. For example, consider if the XRI redirect above had been as follows:

```xml
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT

<XRIDescriptors xmlns="...">
  <XRIDescriptor>
    <Resolved>*example</Resolved>
    <Synonyms>
      <External>
        xri://example2
      </External>
    </Synonyms>
  </XRIDescriptor>
</XRIDescriptors>
```

Deleted: XRIDescriptor/Synonyms/External
Deleted: the response
Deleted: instead
2.3 IRI Authority Resolution

From the standpoint of generic XRI resolution as defined in this specification, an IRI authority segment represents either a DNS name or an IP address at which an XRID for the authority may be retrieved. Requesting the corresponding XRID is a simple matter of making an HTTP(S) GET request using a URL created from the IRI authority segment. The resulting XRI Descriptor is then used to retrieve Local Access URIs or other XRI authority synonyms or metadata as described in section 2.2.

The HTTP URI is constructed by extracting the entire IRI authority segment and prepending the string "http://", Then an HTTP GET is performed using an HTTP Accept header containing only the following:

```xml
Accept: application/xrid+xml
```

Additionally, the HTTP GET request MUST have a Host: header (as defined in section 14.23 of [RFC2616]) containing the value of the IRI authority segment. The resolving authority MUST use the value of the Host header to populate the XRIDescriptor/Resolved element in the resulting XRIDescriptors document. For example:

```xml
Host: example.com
```

An HTTP server acting as an IRI authority SHOULD respond with the XRI Descriptors document for that authority. Section 3 of this document defines trusted resolution only for XRI authorities. This document does not define trusted resolution for IRI Authorities. If, however, an IRI authority is known to respond to HTTPS requests (by some means not described in this document) then the resolving client MAY use HTTPS as the access protocol for retrieving the authority’s XRID. If the resolving client is satisfied, via transport level security mechanisms, that the response is from the expected IRI authority, then the resolving client may place a higher level of trust on the contents of the XRIDescriptor/Resolved element in the resulting XRIDescriptor document. For example:

```xml
<xri://@example2/path
```

The following example demonstrates how the IRI authority segment of the IRI xri://example.com/local*path*home*base/foo would be resolved into an XRI Descriptor. First the IRI authority is extracted ("example.com"), then the following HTTP Request is made of the server:

```xml
GET /local*path
```

Now the resulting XRI would be "xri://@example2/path*home*base/foo". Unless the client application has specific reasons to believe otherwise, this is an error.
The HTTP server acting as the authority might provide the following HTTP response, using the value of the Host: header to fill the contents of the XRIDescriptor/Resolved element:

```
GET / HTTP/1.1
Accept: application/xrid+xml
Host: example.com
<other HTTP headers>

200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
<other HTTP headers>

<XRIDescriptors xmlns="...">
  <XRIDescriptor>
    <Resolved>example.com</Resolved>
    <Synonyms>
      <External>
        xri://@example2*path
      </External>
    </Synonyms>
  </XRIDescriptor>
</XRIDescriptors>
```

The use of IRI authorities provides backwards compatibility with the large installed base of DNS- and IP-identifiable resources. However because IRI authorities do not support the additional layer of abstraction and extensibility represented by XRI authority syntax, IRI authorities are not recommended for new deployments of XRI identifiers.

### 2.4 Local Access

Local access is the process of interacting with a network endpoint to retrieve a representation of or metadata about a resource identified by an XRI.

#### 2.4.1 Local Access Service Types

Any number of protocols may be used for local access. This specification defines an HTTP(S) local access protocol with the name "X2R". Other local access services could be defined such as an LDAP or DSML local access protocol that specified the appropriate transformation of the XRI local part into an LDAP distinguished name (including normalization of the XRI local path to the LDAP distinguished name syntax.)

Work on such protocols is left to future specifications. To accommodate such work, this specification reserves a namespace, "$res*local.access", for enumerating local access service types. The "$res" namespace can also be extended by other authorities besides the XRI Technical Committee. See [XRIMetadata] for more information about extending "$" namespaces.

New local access service types intended for widespread use must be identified with XRIs in the $res*local.access namespace. Local access service types defined solely for use within a private or closed community may have service types identified by any XRI.

#### 2.4.2 The X2R Local Access Service

The X2R local access service is derived from the I2R service defined in section 4.3 of [RFC2483]. X2R is the default local access service defined in this specification; it is available
when the associated `<xrid:Descriptor/xrid:Service/xrid:Type>` element is not present or when it explicitly contains the value "xri://$res*local.access/X2R".

At a high level, X2R is defined as the use of HTTP to interact with a resource using the full extent of the HTTP semantics as defined in [RFC2616]. Special attention should be paid to the semantics of the four main HTTP verbs: GET, PUT, POST, and DELETE. For example, clients performing local access typically will use GET to retrieve representations of a resource on the network.

This specification does not impose particular semantics beyond what is defined in [RFC2616], but users of this specification are encouraged to review the [REST] architecture when building applications using XRIs. Local access is not limited to the REST model of interaction, however, HTTP local access could be leveraged for the delivery of SOAP messages over HTTP POST, for example, or via use of the GET HTTP verb as a generic read-only operation.

The HTTP/HTTPS local access binding defined in this section is flexible enough to be used for a variety of resources. By itself it makes no assumptions about the type of resource identified by the XRI being resolved, however such metadata can be supplied using the `<xrid:Descriptor/xrid:Service/xrid:Type>` element in an XRID. The resource type may also be established through the context in which the XRI was originally used (e.g., an XML document) or discovered through the HTTP Content-Type header.

2.4.2.1 Constructing a Local Access HTTP(S) URI

The HTTP(S) URI upon which to perform X2R local access service is constructed by concatenating the value of any `<xrid:Descriptor/xrid:Service/xrid:Type>` URI element in the XRI Descriptor with the URI normal form of the path portion (matching the xri-path-absolute production described in section 2.2.3 of [XRISyntax]) of the XRI. If the URI from the XRI Descriptor ends in a forward slash "/", then the slash MUST be removed before concatenating the path portion.

The following pseudocode describes the process for creating, from the local access URI in the XRID, the concrete HTTP(S) URI to which a local access request is made:

```java
if (local-access-uri ends in "/"):
    remove trailing "/" in local-access-uri
local-access-uri = local-access-uri + uri-escape(relative-path)
```

The verb used in the resulting HTTP(S) request may be any of the verbs defined in [RFC2616], though not all verbs may be supported at every endpoint. All X2R local access endpoints SHOULD support at least the GET verb, and this should return either a representation of the identified resource or metadata about the resource.

The full suite of HTTP content negotiation features is available to clients when performing local access. For example, if the local access service URI is "http://xri.example.com/xri-local", then the following local access HTTP request for "xri://xri-local/home/foo*bar" could be made to "xri.example.com":

```
GET /xri-local/foo*bar HTTP/1.1
If-Modified-Since: Fri, 31 Oct 2003 19:43:33 GMT
<Other HTTP headers>
```

The following HTTP response might then be received from xri.example.com:
2.5 HTTP Headers

2.5.1 Caching

The full caching capabilities of [RFC2616] should be leveraged for both the default authority resolution service and the X2R local access service. Specifically, implementations of XRI resolution SHOULD implement the caching model described in section 13 of [RFC2616]. In particular, the “Expiration Model” of section 13.2 SHOULD be used, as this requires the fewest round-trip network connections.

All servers providing identifier authority lookup responses SHOULD send the Cache-Control or Expires headers per section 13.2 of [RFC2616] unless there are overriding security or policy reasons.

Note that HTTP Cache headers SHOULD NOT conflict with expiration information in an XRID.

That is, the expiration date specified by HTTP caching headers MUST NOT be later than any of the expiration dates for any of the xrid:XRI/Descriptor/xrid:Expires elements returned in the HTTP response. This implies that lookahead and proxy resolvers have to compute the "soonest" expiration date for XRI Descriptors in a resolution chain and ensure that the HTTP caching headers for the HTTP response do not express a later expiration date.

2.5.2 Location

During authority resolution HTTP interaction, “Location” headers may be present per [RFC2616] (i.e., during 3XX redirects). Redirects SHOULD be made cacheable through appropriate HTTP headers, as specified in section 2.5.1.

2.5.3 Content-Type

For default authority resolution, the “Content-type” header in the 2XX responses MUST contain the value “application/xrid+xml” or “application/xrid-t-saml+xml”, specifying that the content is an XRI Descriptor (section 0) or a trusted XRI Descriptor (section 3.3.1) respectively.

For X2R local access, clients and servers MAY negotiate content type using standard HTTP content negotiation features. Whether or not this feature is used, however, the server MUST respond with an appropriate media type in the “Content-type” header if the resource is found and an appropriate content type is returned.

2.6 Other HTTP Features

HTTP provides a number of other features including transfer-coding, proxying, validation-model caching, etc. All of these features may be used insofar as they do not conflict with the required uses of HTTP described in this document.

2.7 Caching and Efficiency

In addition to HTTP-level caching, resolution clients are encouraged to perform caching at the application level. For best results, however, resolution clients SHOULD be conservative with caching expiration semantics, including cache expiration dates. This implies that in a series of redirects downstream of this initial request, the target XRI is cached if the resource is found and a response is received that includes a cache-expiration date. The redirections SHOULD be made cacheable through appropriate HTTP headers, as specified in section 14.31 of [RFC2616].

For X2R local access, clients and servers MAY negotiate content type using standard HTTP content negotiation features. Whether or not this feature is used, however, the server MUST respond with an appropriate media type in the “Content-type” header if the resource is found and an appropriate content type is returned.
HTTP redirects, for example, the results of the entire process SHOULD only be cached as long as the shortest period of time allowed by any of the intermediate HTTP responses.

Because not all HTTP client libraries expose caching expiration to applications, identifier authorities and local access servers SHOULD NOT use cacheable redirects with expiration times that are relatively short compared to the expiration times of other HTTP responses in the authority resolution chain or local access interactions. In general, all XRI deployments should be mindful of limitations in current HTTP clients and proxies.

For XRI Descriptors, the cache expiration time may also be shortened by the expiration time provided in the `{xri:XRIDescriptor/xri:Expires element (if present)}. That is, if the expiration time in `{xri:XRIDescriptor/xri:Expires` is sooner than the expiration time calculated from the HTTP caching semantics, then the XRI Descriptor MUST be discarded before the expiration time in `{xri:XRIDescriptor/xri:Expires` element. Note also that the SAML assertion used in trusted resolution (section 3) may cause invalidation of a XRI Descriptor even before HTTP caching semantics or the `{xri:XRIDescriptor/xri:Expires` element.

With both application-level and HTTP-level caching, the resolution process is designed to have minimal overhead. In particular, because each qualified sub-segment of an authority identifier is described by a separate XRI Descriptor, each step of that resolution is independent, and intermediate results can typically be cached in their entirety. For this reason, resolution of higher-level (further left) qualified sub-segments, which are common to more identifiers, will naturally result in a greater number of cache hits than resolution of lower-level sub-segments.
3 Trusted Resolution

3.1 Introduction

This section defines a method for performing trusted XRI authority resolution as an extension of the generic XRI resolution protocol defined in section 2 of this document. This trusted resolution protocol does not provide a means to encrypt the contents of resolution requests and responses, nor does it provide a means for a responder to provide different responses for different requestors. These services may be provided by other security protocols used in conjunction with this specification, but confidentiality and client-authentication are explicitly out of scope of this version of this specification. This section assumes the reader is familiar with, at a minimum, the ABNF defined in Appendix A of [XRISyntax] and the generic resolution protocol defined in section 2 of this document.

3.2 Overview and Example (Non-Normative)

Trusted XRI Authority resolution is a straightforward enhancement to generic XRI resolution. The client application requests resolution of one or more qualified sub-segments in the XRI Authority segment exactly as described in section 2 of this document with one exception: instead of using "application/xrid+xml" in the "Accept" header of the HTTP(S) request, a content type of "application/xrid-t-saml+xml" is used. The XRI Authority responds with an XRI Descriptor that contains an additional element - a digitally signed SAML [SAML] assertion that asserts the validity of the containing XRI Descriptor. If the response does not contain a valid, digitally signed SAML assertion (as defined in section 3.2 of this document), trusted resolution may not proceed. The following example will step through resolution of the authority portion of the same XRI used in Section 3.3.3 below.

```
xrid://example*home*base/foo.bar
```

As in standard resolution, there is no defined discovery for the trusted resolution URI(s) of the community root—it must be known a priori and is part of the configuration of the resolver. A recommended practice is to publish an XRI Descriptor containing a valid SAML assertion signed by the community root. In this example, assume the \texttt{xrid:Authority/xrid:URI} element of the XRI descriptor for the global community root "=" specifies that the URI for the "=" global context symbol is "http://equals.example.org/xri-resolve". In trusted resolution, each XRI Authority is associated with an identifier called an AuthorityID. An AuthorityID is a URI or XRI in URI-normal form uniquely associated with a particular XRI Authority. Each XRI Authority MUST have at least one AuthorityID, and no two XRI Authorities can have the same AuthorityID. The AuthorityID of the community root, like the community root's URI, is defined in the \texttt{xrid:XRI Descriptor/xrid:Authority/xrid:AuthorityID} element of the \texttt{community root\'s XRI Descriptor [or its equivalent]}. For this example, assume the AuthorityID for the "=" global context symbol is \texttt{urn:uuid:498FB006-B9EF-4943-B10A-A71FC2ED1B89}. For more information on \texttt{xrid:XRI Descriptor/xrid:AuthorityID}, see Section 3.3.3 below.

Finally, in trusted resolution, each XRI Authority is associated with some key used to verify digital signatures. The key for the community root must be known and configured in advance. If an XRI Descriptor is used to describe the community root, information about this key may be found in the \texttt{xrid:XRI Descriptor/xrid:Authority/ds:KeyInfo} element of that document. Note that the digital signatures in the following examples are for reference only. The digest values are not valid and the signatures will not verify.
Resolving "=example"

The following HTTP request is made to "equals.example.org":

```plaintext
GET /xri-resolve/*example HTTP/1.1
Accept: application/xrid-t-saml+xml
<other HTTP headers>
```

Example 1: Request for =example

Notice the use of the Accept header with the value "application/xrid-t-saml+xml". The client is requesting a response that contains a signed SAML assertion. If the resolving client will accept either trusted or generic resolution, it could have used the value "application/xrid-t-saml+xml, application/xrid+xml" for the Accept header.

The following HTTP response is received from "equals.example.org":

```plaintext
200 OK HTTP/1.1
Content-Type: application/xrid-t-saml+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
<other HTTP headers>
```

```
<xri://$res://schema/XRIDescriptor>
<xri:id="baec221f3c0f17f5ca683998632056">
<Resolved>*example</Resolved>
<AuthorityID>urn:uuid:498FB006-B9EF-4943-B10A-A71FC2ED1B89</AuthorityID>
</Authority>
<TrustMechanism>xri://$res://trusted/XRITrusted</TrustMechanism>
<saml:Assertion Version="2.0"
ID="_ad9571ad-cd23-85e2-e928-abba20b6c424"
IssueInstant="2004-07-01T00:46:02Z"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
<saml:Issuer>urn:uuid:498FB006-B9EF-4943-B10A-A71FC2ED1B89</saml:Issuer>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
<ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
<ds:Reference URI="#baec221f3c0f17f5ca683998632056" />
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
</ds:Transforms>
<ds:SignatureValue>
<xs:string>...
</xs:string>
</ds:Signature>
</saml:Assertion>
</saml:XRIDescriptors>
```
Example 2 – Response for =example

The response contains an xrid:XRI_descriptor/saml:Assertion element that provides an assertion about the validity of the XRI Descriptor. For more information about SAML assertions in XRI Descriptors, see Section 3.3.3. The response also contains an xrid:XRI_descriptor/xrid:Authority/ds:KeyInfo element. This required element tells the client that digital signatures by the described XRI Authority are to be verified using the indicated key.

Finally, notice that two instances of xrid:AuthorityID appear in the XRI Descriptor, one as a child of xrid:XRI_descriptor and one as a child of xrid:Authority. The child of xrid:XRI_descriptor is the Authority ID of the current describing authority (the one publishing this XRI Descriptor) and matches the expected Authority ID of the community root (urn:uuid:498FB006-B9EF-4943-B10A-A71FC2ED1B89). The child of the xrid:Authority element contains the Authority ID of the described XRI Authority (the authority being described within the xrid:Authority element). Responses from that XRI Authority will contain this Authority ID as a child of xrid:XRI_descriptor.

The client validates the signed SAML assertion as described in Section 3.3 before continuing.

Resolving "=example"home"

Appending the next qualified sub-segment "home" to the URI "http://xri.example.com/xri-resolve/" yields the URI http://xri.example.com/xri-resolve/*home. The following HTTP request with an Accept header value of "application/xrid-t+saml+xml" is made to "xri.example.com".
GET /xri-resolve/*home HTTP/1.1
Accept: application/xrid-t-saml+xml

Example 3 – Request for *home

The following HTTP response is received from xri.example.com:

200 OK HTTP/1.1
Content-Type: application/xrid-t-saml+xml
If-Modified-Since: Fri, 31 Oct 2003 19:43:32 GMT

<xrIDescriptors xmlns="xri://$res*/schema/XRIDescriptor">
  <XRIDescriptor
    xrid:id="1f81b6e0-b64b-1026-f1bc-c0a80b9d3f5b">
    <Resolved>*home</Resolved>
    <AuthorityID>urn:uuid:C5C9EFDF-A3BC-4301-88C6-B1AE0A6DA77</AuthorityID>
    <Authority>
      <AuthorityID>urn:uuid:A9F28515-AB03-4883-8852-8EECB54CE1D5</AuthorityID>
      <URI xrid:trusted="true">
        http://xri.example.com/xri-resolve/*home/
      </URI>
    </Authority>
    <Service>…</Service> <!-- Local Access Service -->
    <TrustMechanism>xri://$res*trusted/XRITrusted</TrustMechanism>
  </XRIDescriptor>
</xrIDescriptors>

<saml:Assertion
  Version="2.0"
  ID="_66f1f3e0-b64b-1026-34a4-c0a80b9d59c1"
  IssueInstant="2004-05-01T00:46:03Z"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
  <saml:Issuer>urn:uuid:C5C9EFDF-A3BC-4301-88C6-B1AE0A6DA77</saml:Issuer>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:CanonicalizationMethod
      Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    <ds:SignatureMethod
      Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <ds:Reference URI="#1f81b6e0-b64b-1026-f1bc-c0a80b9d3f5b">
      <ds:Transforms>
        <ds:Transform
          Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
      </ds:Transforms>
      <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
        <ec:InclusiveNamespaces
          xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
          PrefixList="#default code ds kind rw saml samlp typens" />
      </ds:Transform>
    </ds:Reference>
  </ds:Signature>
</saml:Assertion>

<!-- Local Access Service -->

Example 4 – Response for *home

The client validates the SAML assertion as described in Section 3.3 before continuing.

Resolving "=example*home*base"

Appending the next qualified sub-segment "=base" to the URI "http://xri.example.com/xri-resolve/*home" gives the URI "http://xri.example.com/xri-resolve/*home/*base", This is the target of the next trusted resolution request, again with the Accept header value "application/xrid-t-saml+xml".

Example 5 – Request for *base

The following HTTP response is received from xri.example.com:

200 OK HTTP/1.1
Content-type: application/xrid-t-saml+xml
Expires: Fri, 7 Nov 2003 19:43:33 GMT
X-
XRIDescriptors
xmlns="#rest/schema/XRIDescriptor"
XRIDescriptor
xrid:id="7600e1a0-b64d-1026-ea89-c0a80b9d3814"
Resolved:*base
<Resolved>
<Resolved>
<AuthorityID>urn:uuid:A9F28515-AB03-4883-8852-8E8E5B4CE8D5
</AuthorityID>
<AuthorityID>
<Service>
</Service>
</Type>xri:///@res=local.access/X2R</Type>

Deleted: $res*localaccess

Deleted: xrid-t+xml

Deleted: xrid-t+xml

Deleted: xrid-t+xml

Deleted: xrid-t+xml

Deleted: xrid-t+xml

Deleted: xrid-t+xml

Deleted: xrid-t+xml
<TrustMechanism>xri://$res*trusted/XRITrusted</TrustMechanism>
<saml:Assertion
    Version="2.0"
    ID="_1a6a12d0-b64d-1026-c1ba-c0a80b9db964"
    IssueInstant="2004-06-03T00:46:03Z"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
    <saml:Issuer>urn:uuid:A9F28515-AB03-4883-8852-8EECB54CE1D5</saml:Issuer>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
            <ds:CanonicalizationMethod
                Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
            <ds:SignatureMethod
                Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
            <ds:Reference URI="#7600e1a0-b64d-1026-ea89-c0a80b9d3814">
                <ds:Transforms>
                    <ds:Transform
                        Algorithm="http://www.w3.org/2000/09/xmldsig#hmac-sha1" />
                </ds:Transforms>
            </ds:Reference>
            <ds:SignatureValue>
                kE9p35G4mcombsqEztJMX1R3J26gwc4cbjSz5fUv3aVg3j/iLhrbf0qK5wYNMLdQMjBencGg5N1N10
                Kv72UvK5dQ9d7m/563RzKAAIQoWop2pTf14eXw+ncSXEHKnXdu/R9DHOG9KO8KIF6
                BGk07
                xC6Q9XbyQWenPfAZlc=
            </ds:SignatureValue>
        </ds:SignedInfo>
    </ds:Signature>
    <saml:Subject>
        <saml:NameID NameQualifier="urn:uuid:A9F28515-AB03-4883-8852-8EECB54CE1D5">
            *example
        </saml:NameID>
        <saml:Subject>
            <saml:Conditions
                NotBefore="2004-06-03T00:46:03Z"
                NotOnOrAfter="2004-12-01T00:00:00Z" />
            <saml:AttributeStatement>
                <saml:Attribute Name="xri://$res*trusted/XRIDescriptor">
                    #baec221f3c0f17f53ca6839989632056</saml:AttributeValue>
                </saml:Attribute>
            </saml:AttributeStatement>
        </saml:Subject>
    </saml:Subject>
</saml:Assertion>
3.3 Trusted Resolution Protocol

This section normatively defines client and server behavior in trusted resolution. It also defines a new XML element for an XRI Descriptor, called trustMechanism.

3.3.1 XML Elements and Attributes

Three elements of an XRI Descriptor that were defined in section 2.2.2 have limited usage in generic resolution but play a critical role in trusted resolution.

- xrid:X RID escriptor/xrid:AuthorityID
- Always required, but critical in trusted resolution for identification of the current describing authority.

- xrid:X RID escriptor/xrid:Authority/xrid:AuthorityID
- Always required, but critical for trusted resolution for identification of the target described authority.

- xrid:X RID escriptor/xrid:TrustMechanism
- Required when providing trusted resolution. A URI or XRI in URI-normal form that specifies the mechanism used to provide trusted resolution. The URI for the trust mechanism defined in this specification is "xri://$res*trusted/XRI Trusted".

In addition, one element from the SAML [SAML] namespace is also critical for verifying the results of trusted resolution.

- xrid:X RID escriptor/saml:Assertion
- Required when providing trusted resolution. A SAML assertion from the describing Authority (the one providing the XRI Descriptor) that asserts that the describing authority believes the information contained in the enclosing XRI Descriptor is correct. Because the assertion is digitally signed and the digital signature encompasses the containing XRI Descriptor, it also provides a mechanism for the recipient to detect unauthorized changes since the time the XRI Descriptor was published.

Finally, trusted resolution adds several new elements and attributes to XRI Descriptors to assist in verifying XRIDs produced by the next authority in the resolution chain (the authority being described by the xrid:X RID escriptor/xrid:Authority element of the XRID):

- xrid:X RID escriptor/xrid:Authority/xrid:URI/@trusted
- Optional. Default value of "false" (or "0"). Indicates whether this service endpoint is capable of returning trusted resolution results. If the value is "1" or "true", the described authority is willing to return signed XRI Descriptors at this URI.

- xrid:X RID escriptor/xrid:Authority/ds:KeyInfo
- Required when providing trusted resolution. Provides the key data needed to validate an XRI Descriptor provided by the described Authority as a result of resolution at the...
Figure 4 below demonstrates the relationship between these elements for two descriptors in a resolution chain: one describing an authority, and one produced by the authority being described.

![Diagram of XRID Elements](image)

**3.3.2 Use and Correlation of AuthorityID Elements**

Each XRI Authority participating in trusted resolution MUST be associated with exactly one AuthorityID and this AuthorityID MUST NOT ever be assigned to any other XRI Authority. In other words, AuthorityID is a persistent unique identifier for a particular XRI Authority.

An AuthorityID may be any valid URI that meets the requirements of permanence and uniqueness described above. Examples of appropriate URIs include URNs as defined by [RFC2141] and fully persistent XRIs converted to “URI Normal Form” as defined by [XRISyntax].

Conceptually, AuthorityID assures a resolving client that the returned XRI Descriptor has not been maliciously replaced with a similar XRI Descriptor from a second, and possibly unauthorized, XRI Authority. To prevent this type of attack, the XRI Descriptor must be explicitly associated with a specific unique XRI Authority, and the client must have some means of verifying this association. The `xri:XRIIdentifier/xri:AuthorityID` element provides this explicit association.

There is no defined discovery for the AuthorityID of the community root; it must be published in the root XRI Description document and verified independently. The AuthorityID for an XRI Authority other than the community root is described by an `xri:XRIIdentifier/xri:AuthorityID` element in the authority’s XRI Descriptor.

**3.3.3 Client Behavior**

From a client's perspective, trusted resolution is identical to the generic resolution protocol described in section 2 of this document with the addition of the following REQUIRED behavior:

- The client MUST indicate to the remote server that a `signed` XRI Descriptor is requested. This is expressed by adding an `HTTP Accept` header with the media type (`xri:`) specified.

Deleted:
- that
- NEVER
- permanently
- "

Deleted:
- There must therefore be
- The client will accomplish this
- subsegment
- sub-segments being
- pair
- to sign for both the
- particular
- In trusted resolution
- ve different
- its own unique
- Because the client
- xri:
- xri:XRIIdentifier/
- xri:Authority
- xri:
- appearing as a child of the
- xri:
- Should we consider
- mechanism
- 3
- [XRISyntax]
- s
The client **MUST** confirm that each `xri:XRIDescriptor` element contains an `saml:Assertion` element as an immediate child, and that this assertion is valid per the processing rules described by [SAML]. In addition, the following requirements **MUST** be met:

- The `saml:Assertion` must contain a valid enveloped digital signature as defined by [XMLDSig] and constrained by Section 5.4 of [SAML].
- The signature must apply to the `xri:XRIDescriptor` element that contains the signed SAML assertion. Specifically, the signature must contain a single `ds:SignedInfo/ds:Reference` element, and the URI attribute of this reference must refer to the id (`xrid:id` attribute) of the `xri:XRIDescriptor` element that is the immediate parent of the signed SAML assertion.
- If the digital signature enveloped by the SAML assertion contains a `ds:KeyInfo` element, the client MAY reject the signature if this key does not match the signer’s “expected key”. The expected key is specified by the `ds:KeyInfo` element present in the XRI Descriptor that was used to describe the current Authority. For example, if Authority A provides an XRI Descriptor describing Authority B, and this XRI has an `xri:XRIDescriptor/xri:Authority/ds:KeyInfo` element that describes the key used to validate descriptors produced by Authority B, then this is Authority B’s “expected key”. This key should be used when validating XRI Descriptor elements produced by Authority B. **Note that for a community root authority, the expected key is known a priori as part of the configuration in the client for that particular community root.**
- The client confirms that the value of the `xri:XRIDescriptor/xri:Resolved` element matches the sub-segment whose resolution resulted in the current XRI Descriptor.
- The client confirms that the value of the `xri:XRIDescriptor/xri:AuthorityID` element matches the XRI Authority’s “expected AuthorityID”. As with the key information, the “expected AuthorityID” is the value of `xri:XRIDescriptor/xri:Authority/xri:AuthorityID` in the XRI Descriptor that describes the current Authority. **Note that for a community root authority, the XRI Authority’s expected AuthorityID is known a priori and is part of the configuration in the client for that particular community root.**
- The client confirms that the value of the `xri:XRIDescriptor/saml:Assertion/saml:Issuer` element and the NameQualifier attribute of the `xri:XRIDescriptor/saml:Assertion/saml:Subject/saml:NameID` element.
- The client confirms that the value of the `xri:XRIDescriptor/xri:Resolved` element matches the value of the `xri:XRIDescriptor/saml:Assertion/saml:Subject/saml:NameID` element.
If any of the above requirements are not met for any XRI Descriptor in the resolution chain, the result MUST NOT be considered a valid trusted resolution response as defined by this document. Note that this does not preclude a client from attempting an iteration multiple times or from performing an alternate resolution step if the above requirements are not met. For example, if two URIs are listed under an xrid:Authority element and the response from one fails to meet the requirements above, the client may repeat the current iteration using the second URI. If the second URI produces a sufficient response, it may be considered a valid trusted resolution response as defined by this document and trusted resolution may continue.

### 3.3.4 Server Behavior

From the server’s perspective, trusted resolution is identical to the generic resolution protocol described in section 2 of this document with the addition of the following behavior. This behavior is REQUIRED if a resolution client requests trusted resolution as described in section 3.2 and the server intends to honor the client’s request.

If, during the HTTP(S) request/response interaction, the server agrees to return a trusted resolution response (indicated by the content type of “application/xri-t-saml+xml”), the XRI Descriptor returned by the server must contain a saml:Assertion element as an immediate child of xrid:XRI Descriptor that is valid per the processing rules described by [SAML]. In addition, the following requirements MUST be met:

- The saml:Assertion MUST contain a valid enveloped digital signature as defined by [XMLDSig] and as constrained by section 5.4 of [SAML].
- The signature MUST apply to the xrid:XRI Descriptor element that contains the signed SAML assertion. Specifically, the signature must contain a single ds:SignedInfo/ds:Reference element, and the URI attribute of this reference MUST refer to the xrid:XRI Descriptor element that is the immediate parent of the signed SAML assertion. The URI reference MUST NOT be empty; it MUST refer to the identifier contained in the xrid:id attribute of the xrid:XRI Descriptor element.
- The digital signature enveloped by the SAML assertion MAY contain a ds:KeyInfo element. If it is included, it MUST match the xrid:Descriptor/xrid:Authority/ds:KeyInfo element in the XRI Descriptor that describes the current Authority, unless the signing XRI Authority is the community root. If the signing XRI Authority is the community root, the ds:KeyInfo element must match the well-known signing key for that XRI Authority (which may or may not be published via an XRI ID). Because the signing key is known in advance by the resolution client, the ds:KeyInfo element SHOULD be omitted from the digital signature.
- The xrid:Resolved element MUST be present, and the value of this field MUST match the XRI Authority sub-segment requested by the client.
- The xrid:XRI Descriptor element MUST have an xrid:AuthorityID element as an immediate child. The value of the xrid:AuthorityID element MUST be the Authority ID, as described in Section 3.2, of the responding XRI Authority.
- The xrid:XRI Descriptor/xrid:TrustMechanism MUST be present and the value MUST be “xri://$res*trusted/XRITrusted”.

If a resolving client requests trusted resolution and lookahead resolution, the responding authority SHOULD attempt to perform trusted resolution on behalf of the client as described in section 3. However, the server providing lookahead resolution MUST NOT return untrusted XRI Descriptors if the client requests trusted resolution.
3.3.5 Additional Requirements of Authorities Offering Trusted Resolution

The `xrid:XRI_DESCRIPTOR/xrid:Authority` element that describes an authority participating in trusted resolution as defined by this specification ("the described XRI Authority") has the following requirements:

- The "trusted" attribute of the `xrid:XRI_DESCRIPTOR/xrid:Authority/xrid:URI` element must contain the value "1" or "true".
- The `xrid:XRI_DESCRIPTOR/xrid:Authority` element MUST contain a `ds:KeyInfo` element as an immediate child. The value of this element MUST be the key that validates digital signatures created by the described XRI Authority.
- The `xrid:XRI_DESCRIPTOR/xrid:Authority` element MUST contain an `xrid:AuthorityID` element as an immediate child. The value of this field MUST be the `AuthorityID` of the described XRI Authority, i.e. the value that will appear in the `xrid:READER/xrid:AuthorityID` element of an XRI Descriptor returned from the described XRI Authority.

- In addition, an identifier community SHOULD publish an XRI Descriptor for the community root that meets the requirements listed above and it SHOULD make that XRI Descriptor easily available to relevant parties.
4 Extensibility and Versioning

4.1 Extensibility

The XRI Descriptors defined in this document are designed to be extended with other metadata. Therefore the XRI Descriptor schema is built on an open content model. In a number of places, extension elements and attributes from namespaces other than "xri://sres/schema/XRI_descriptor*($v%2F2.0)" are explicitly allowed. These extension points are designed to simplify default processing of XRI Descriptors using a "Must Ignore" rule. The base rule is that unrecognized elements and attributes, and the content and child elements of unrecognized elements, MUST be ignored. This means that even elements that would normally be recognized by a processor MUST be ignored if they appear as descendants of an unrecognized element.

Extension elements MUST NOT require new interpretation of elements defined in this document. That is, if an extension element is present, a processor must be able to ignore it and still correctly process the Descriptor document.

Extension specifications MAY simulate "Must Understand" behavior by applying an "enclosure" pattern. Elements defined by the XRI Descriptor schema whose meaning or interpretation are to be modified by extension elements can be wrapped in an extension container element, that is defined by the extension specification. This extension container element will be in the same namespace as the extension elements that must be understood by the consumer of the XRI Descriptor. Using this pattern, all elements whose interpretations are modified by the extension will now be contained in an element (the extension container element) that will be ignored by consumers unable to process the extension.

The following example illustrates this pattern using an extension container element from an extension namespace ("other:SuperAuthority") that contains an extension element ("other:ExtensionElement")

```xml
<XRI_descriptor>
  <other:SuperAuthority>
    <Authority>
      ...
    </Authority>
  </other:SuperAuthority>
</XRI_descriptor>
```

In this example, the other:ExtensionElement modifies the interpretation or processing rules for the parent xri:Authority element and therefore must be understood by the consumer for the proper interpretation of the parent xri:Authority element. To preserve the correct interpretation of the xri:Authority element in this context, the xri:Authority element is "wrapped" so only consumers that understand elements in the other:SuperAuthority namespace will attempt to process the xri:Authority element.

4.1.1 Specific Points of Extensibility

The use of HTTP and XML in the design of the generic resolution service, the trusted resolution service, and the X2R local access service provide the following specific points of extensibility:
4.2 Versioning

Versioning of the XRI specification set is expected to occur infrequently, however experience demonstrates such versioning is inevitable. Therefore guidelines on versioning are described in this section.

When version information is expressed as both a Major and Minor version, it is expressed in the form Major.Minor. The version number Major MINOR is higher than the version number Major.REV if and only if:

Major > Major OR ( ( Major = Major ) AND Minor > Minor )

4.2.1 Versioning of the XRI Resolution Specification

New releases of the XRI Resolution specification may specify changes to the resolution protocol and/or to resolution data structures. When changes affect either of these, the resolution specification version number will be changed. Where changes are purely editorial, the version number will not be changed.

In general, if a change is backwards compatible, the new version will be identified using the current major version number and a new minor version number. If the change is not backwards compatible, then the new version will be identified with a new major version number.

4.2.2 Versioning of XRI Descriptor Elements

Both the xrid:XRIDescriptors element and the xrid:XRIDescriptor element have Version attributes. The value of these attributes MUST be the version value of specification to which their containing elements conform.

When new versions of the XRI Resolution specification are released, the namespace for the XRI Descriptor schema may or may not be changed. If there is a major version number change, the namespace for the xrid:XRIDescriptors document is likely to change. If there is only a minor version number change, the namespace for the xrid:XRIDescriptors document may remain unchanged.

In general, maintaining namespace stability while adding or changing the content of a schema are competing goals. While certain design strategies can facilitate such changes, it is complex to predict how older implementations will react to any given change, making forward compatibility difficult to achieve. Nevertheless, the right to make such changes in minor revisions is reserved.

Except in special circumstances (for example, to correct major deficiencies or to fix errors),


implementations should expect forward-compatible schema changes in minor revisions, allowing new messages to validate against older schemas.

Implementations SHOULD expect and be prepared to deal with new extensions and message types in accordance with the processing rules laid out for those types. Minor revisions may introduce new types that leverage the extension facilities described in Section 4.1. Older implementations SHOULD reject such extensions gracefully when they are encountered in contexts that dictate mandatory semantics.

4.2.3 Versioning of Protocols

Both the authority resolution and local access protocols defined in this document may also be versioned by future releases of the XRI Resolution specification. If these protocols are not backwards compatible with older implementations, they will likely get a new XRI identifying them in an XRI Descriptor.

Note that it is possible for version negotiation to happen in the protocol itself. For example, HTTP provides a mechanism to negotiate the version of the HTTP protocol being used. When the XRI resolution protocol provides its own version negotiation mechanism, the specification is likely to continue to use the same XRI identifying the protocol as used in previous versions of the XRI Resolution specification.

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY

Deleted: MAY
5 Security and Data Protection

5.1 DNS Spoofing

XRI resolution responses that depend on DNS resolution of an XRI authority URI are no more secure than the underlying DNS query. To solve that, either the use of DNSSEC [DNSSEC] for securing and verifying zones, the use of server authenticated HTTPS, or the use of trusted resolution as defined in this specification is recommended.

5.2 HTTP Security

Many of the security considerations set forth in HTTP/1.1 [RFC2616] apply to XRI Resolution protocols defined here. In particular, confidentiality of the communication channel is not provided for by HTTP. Server-authenticated HTTPS should be considered in cases where confidentiality of resolution requests and responses is desired.

Special consideration should be given to proxy and caching behaviors to ensure accurate and reliable responses from resolution requests. In particular, for various reasons network topologies increasingly have transparent proxies, some of which may insert VIA and other headers as a consequence, or may even cache content without regard to proper caching policies set for by the HTTP authority for a resource.

5.3 Caching Authorities

In addition to traditional HTTP caching proxies, XRI resolution authority proxies may exist in the resolution topology. These entities should take special precautions against cache poisoning (as described in [RFC...]), as these caching entities may represent trust decision points within a deployment's resolution architecture.

5.4 Lookahead and Proxy Resolution

During proxy resolution, some or all of the XRI Authority is provided to the proxy resolver. During lookahead resolution sub-segments may be revealed to the network endpoint providing resolution for which that endpoint is not authoritative.

In both cases, privacy considerations should be evaluated before disclosing such information.

5.5 SAML Considerations

For trusted resolution, rules defined by the SAML 2.0 Core Specification must be adhered to. Particularly noteworthy are the XML Transform restrictions on XML Signature defined in SAML and the enforcement of the SAML Conditions element regarding the validity period.

5.6 Community Root Authorities

The XRI Authority information for a community root needs to be well-known to the clients that request resolution within that community. For trusted resolution, this includes the URIs, the AuthorityID, and the ds:KeyInfo information. A suitable way to provide this information is for the community root authority to provide a self-signed XRI Descriptor and publish it to a server-authenticated HTTPS endpoint. Special care should be taken to ensure the correctness of such
an XRID; if this information is incorrect, an attacker may be able to convince a client of an
incorrect result during trusted resolution.

5.7 DOS Attacks

XRI Resolution, including trusted resolution, is vulnerable to the typical DOS attacks found in
systems relying on DNS and HTTP.

5.8 Limitations of Trusted Resolution

While the trusted resolution mechanism specified in this document provides a way to verify the
integrity of a successful XRI resolution, it does not provide a way to verify the integrity of a
resolution failure. Reasons for this limitation include non-malicious network failures, DOS attacks,
and the ability for a MITM attacker to modify HTTP responses when resolution is not performed
over HTTPS.

Additionally, there is no revocation mechanism specified for trusted resolution. Therefore, a
signed resolution’s validity period should be limited appropriately to mitigate the risk of an
incorrect or invalid resolution.
6 Media Type Registration for application/xrid+xml

To: ietf-types@iana.org
Subject: Registration of MIME media type application/xrid+xml

MIME media type name: application
MIME subtype name: xrid+xml
Required parameters: None
Optional parameters: None
Encoding considerations: XML
Security considerations:
Interoperability considerations:
Published specification:
Applications which use this media type:
Additional information:
Magic number(s):
File extension(s):
Macintosh File Type Code(s):
Person & email address to contact for further information:
Intended usage: COMMON
Author/Change controller:
Media Type Registration for application/xrid-t-saml+xml

To: ietf-types@iana.org
Subject: Registration of MIME media type application/xrid-t-saml+xml

MIME media type name: application
MIME subtype name: xrid-t-saml+xml
Required parameters: None
Optional parameters: None
Encoding considerations:
Security considerations:
Interoperability considerations:
Published specification:
Applications which use this media type:
Additional information:
Magic number(s):
File extension(s):
Macintosh File Type Code(s):
Person & email address to contact for further information:
Intended usage: COMMON
Author/Change controller:
8 References

8.1 Normative


8.2 Informative


## Appendix A. Revision History

<table>
<thead>
<tr>
<th>Rev</th>
<th>Date</th>
<th>By Whom</th>
<th>What</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>2005-01-15</td>
<td>All Editors</td>
<td>Initial document.</td>
</tr>
<tr>
<td>3.0</td>
<td>?</td>
<td>Gabe Wachob</td>
<td>Incorporate feedback from other editors.</td>
</tr>
<tr>
<td>4.0</td>
<td>2005-02-17</td>
<td>Gabe Wachob</td>
<td>Incorporate feedback. Add Extensibility and Versioning section.</td>
</tr>
<tr>
<td>5.0</td>
<td></td>
<td>Gabe Wachob</td>
<td>Incorporate feedback and editorial revisions.</td>
</tr>
</tbody>
</table>
Appendix B. XML Schema for XRI Descriptor
(Normative)

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="xri://$res*schema/XRIDescriptor*(v%2F2.0)"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xrid="xri://$res*schema/XRIDescriptor*(v%2F2.0)"
elementFormDefault="qualified">
    <!-- Utility patterns -->
    <xs:attributeGroup name="otherattribute">
        <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:attributeGroup>
    <xs:group name="otherelement">
        <xs:choice>
            <xs:any namespace="##other" processContents="lax"/>
            <xs:any namespace="##local" processContents="lax"/>
        </xs:choice>
    </xs:group>
    <xs:complexType name="URIpattern">
        <xs:simpleContent>
            <xs:extension base="xs:anyURI">
                <xs:attributeGroup ref="xrid:otherattribute"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
    <xs:complexType name="Stringpattern">
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attributeGroup ref="xrid:otherattribute"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
    <!-- Patterns for elements -->
    <xs:element name="XRIDescriptors">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="xrid:XRIDescriptor" maxOccurs="unbounded"/>
                <xs:group ref="xrid:otherelement" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attributeGroup ref="xrid:otherattribute"/>
            <xs:attribute ref="xrid:version"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="XRIDescriptor">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="xrid:Resolved" maxOccurs="unbounded"/>
                <xs:element ref="xrid:AuthorityID" maxOccurs="unbounded"/>
                <xs:element ref="xrid:Expires" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="xrid:Authority" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="xrid:Service" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="xrid:Synonyms" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element ref="xrid:TrustMechanism" minOccurs="0" maxOccurs="unbounded"/>
                <xs:group ref="xrid:otherelement" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute ref="xrid:id" maxOccurs="unbounded"/>
            <xs:attributeGroup ref="xrid:otherattribute"/>
            <xs:attribute ref="xrid:version" maxOccurs="unbounded"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Resolved" type="xrid:Stringpattern"/>
    <xs:element name="Expires"/>
</xs:schema>
```
<xs:complexType>
  <xs:simpleContent>
    <xs:extension base="xs:dateTime">
      <xs:attributeGroup ref="xrid:otherattribute"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="Authority">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="xrid:AuthorityID" minOccurs="0" maxOccurs="unbounded"/>
      <xs:group ref="xrid:TrustableURI" maxOccurs="unbounded"/>
      <xs:group ref="xrid:otherelement" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attributeGroup ref="xrid:otherattribute"/>
  </xs:complexType>
</xs:element>

<xs:element name="AuthorityID" type="xrid:URIpattern"/>
<xs:element name="Type" type="xrid:URIpattern"/>
<xs:group name="TrustableURI">
  <xs:sequence>
    <xs:element name="URI">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xrid:URIpattern">
            <xs:attribute ref="xrid:trusted"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:group>

<xs:element name="Service">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="xrid:Type" minOccurs="0" maxOccurs="unbounded"/>
      <xs:group ref="xrid:URI" maxOccurs="unbounded"/>
      <xs:element ref="xrid:MediaType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:group ref="xrid:otherelement" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attributeGroup ref="xrid:otherattribute"/>
  </xs:complexType>
</xs:element>

<xs:group name="URI">
  <xs:sequence>
    <xs:element name="URI" type="xrid:URIpattern"/>
  </xs:sequence>
</xs:group>

<xs:element name="MediaType" type="xrid:Stringpattern"/>
<xs:element name="Synonyms">
  <xs:complexType>
    <xs:sequence>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="xrid:Internal"/>
        <xs:element ref="xrid:External"/>
      </xs:choice>
      <xs:group ref="xrid:otherelement" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attributeGroup ref="xrid:otherattribute"/>
  </xs:complexType>
</xs:element>

<xs:element name="Internal" type="xrid:URIpattern"/>
<xs:element name="External" type="xrid:URIpattern"/>
<xs:element name="TrustMechanism" type="xrid:URIpattern"/>
<xs:attribute name="version" type="xs:string" fixed="2.0"/>
<xs:attribute name="trusted" type="xs:boolean"/>
<xs:attribute name="id" type="xs:ID"/>
</xs:schema>
Appendix C. RelaxNG Compact Syntax Schema for XRI Descriptor (Non-normative)

namespace xrid="xri://$res*schema/XRIDescriptor*(v%2F2.0)"
namespace xml="http://www.w3.org/XML/1998/namespace"
namespace local=""

start=XRIDescriptors

# Utility patterns
anything = ( element * | attribute * | text ) *
otherelement = element *-xrid:* { anything }
URIpattern = (xsd:anyURI, otherattribute *)
Stringpattern = (xsd:string, otherattribute *)
versionattribute = attribute xrid:version ( text )
idattribute = attribute xrid:id ( xsd:ID )

##########################
# XRIDescriptors Container
XRIDescriptors = element xrid:XRIDescriptors {
  versionattribute,
  XRIDescriptor+, XRIDescriptor-ex-elem, XRIDescriptor-ex-attr
}

# XRIDescriptors Extension
XRIDescriptors-ex-elem = otherelement *
XRIDescriptors-ex-attr = otherattribute *

##########################
# XRIDescriptor Definition
XRIDescriptor = element xrid:XRIDescriptor {
  attribute xrid:id {xsd:ID}?,
  versionattribute,
  Authority?, Expires ?,
  Authority *, Service *,
  Synonyms ?,
  TrustMechanism ?,
  XRIDescriptor-ex-elem,
  XRIDescriptor-ex-attr
}

# XRIDescriptor Extension
XRIDescriptor-ex-elem = otherelement *
XRIDescriptor-ex-attr = otherattribute *

##########################
# Resolved Definition
Resolved = element xrid:Resolved { Resolved-content}

# Resolved Extension
Resolved-content = Stringpattern

# Expires Definition
Expires = element xrid:Expires {
  xsd:dateTime,
  Expires-ex-attr
}
Expires-ex-attr = otherattribute *

# Authority Definition
Authority = element xrid:Authority {
  AuthorityID?,
  Type?,
  TrustableURI+,
  Authority-ex-attr,
  Authority-ex-elem
}

# Authority Extension
Authority-ex-attr = otherattribute *
Authority-ex-elem = otherelement *

# AuthorityID Definition
AuthorityID = element xrid:AuthorityID { AuthorityID-content}

# AuthorityID extension
AuthorityID-content = URIpattern

# Type Definition
Type = element xrid:Type { Type-content}

# Type Extension
Type-content = URIpattern

# Trustable URI Definition
TrustableURI = element xrid:URI { TrustableURI-content }

TrustableURI-content = {
  URIpattern,
  attribute xrid:trusted {xsd:boolean}?
}

# Service Definition
Service = element xrid:Service {
  Type?,
  URI+,
  MediaType *,
  Service-ex-attr,
  Service-ex-elem
}

# Service Extension
Service-ex-attr = otherattribute *
Service-ex-elem = otherelement *

# URI Definition (for Service element)
URI = element xrid:URI { URI-content }

# URI Extension
URI-content = URIpattern

# MediaType Definition
MediaType = element xrid:MediaType { MediaType-content }

# MediaType Extension
MediaType-content = URIpattern

# Synonyms Definition
Synonyms = element xrid:Synonyms {
  }

Deleted: ?
Internal &
External
)+,
Synonyms-ex-attr,
Synonyms-ex-elem
)

Synonyms-ex-attr = otherattribute *
Synonyms-ex-elem = otherelement *

# Internal Definition
Internal = element xrid:Internal { Internal-content }

# Internal Extension
Internal-content = URIpattern

# External Definition
External = element xrid:External { External-content }

# External Extension
External-content = URIpattern

# TrustMechanism Definition
TrustMechanism = element xrid:TrustMechanism { TrustMechanism-content }

# TrustMechanism Extension
TrustMechanism-content = URIpattern
Appendix D. 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 2005. 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.
Appendix E. LAST MINUTE TODOs

- Last Minute: Make sure all the examples show XRI’s in URI-normal form
- Last Minute: Make sure all XRIs are legal (esp. making them start with “xri://”)
- Last Minute: Confirm editors/contributors list
- Last Minute: Update, prune and verify the references, esp to other XRI specs and new RFCs
- Last Minute: Ensure that cross references to other XRI document sections are still correct
- Last Minute: Ensure correct formatting of XRIs, XPATHs, XML examples, etc

All XML Paths should use XMLPath style and all element names should be qualified (mostly that’s xrid: for the elements we define)

- Last Minute: What about Acknowledgements section?
xri://$res*types Namespace for resource representation types 2.4.2.2

together

with

its preceding

The

qualified

specifies

Each XRI community, as identified

that contains the same information as an XRID.

This left to right Resolution results

in

Descriptor

(often abbreviated as "XRIDs"), each one associated with a particular sub-segment of the XRI authority segment

being resolved
each HTTP request in XRI authority resolution may resolve one or more sub-segments at a time. each HTTP response therefore

XRI Descriptor elements in a XML container document. The number of sub-segments resolved depends on:

- Whether the resolution request is done in “proxy” mode
- How many sub-segments the resolving client presents to a responding XRI Authority

Configuration, policy, and state of the responding XRI Authority (e.g. previously cached requests).

No matter how the XRI Descriptor documents are retrieved, a successful XRI Authority resolution results in an ordered chain of XRI Descriptor documents.

**The Chain of XRI Descriptors**

The chain of XRI Descriptors associated with an XRI Authority begins with an XRI Descriptor document (or its equivalent) associated with the community root sub-segment (the “root XRID”). This sub-segment is either a GCS character or a cross-reference. In either case, The root XRID (or its equivalent) will contain references to one or more Authority resolution services at which the community root authority will resolve one or more sub-segments after the community root sub-segment made. The resolving client adds to the chain by resolving one or more sub-segments via the community root authority, parses the resulting XRIDs, and proceeds to the next unresolved XRI Authority identifier sub-segment until the entire XRI Authority identifier is resolved. This document specifies one Authority resolution protocol built on HTTP (and HTTPS), and identified by the XRI “xri://$/res*authres/XRIA”, defined in section 2.2.5.2. Other authority resolution services may be defined in the future by other documents.

Each XRI Descriptor in the chain contains HTTP or HTTPS URIs (or other URIs, depending on the authority resolution service type) that point to the next authority at which resolution can be performed for the next sub-segment. (Note that this will be true even if the resolving authority performs lookahead resolution, where an authority resolves multiple behalf of the resolving client, possibly even beyond those for which the authority is actually authoritative. This allows both the client and server the ability to cache intermediate results for future resolution requests.)

The last XRI Descriptor in the chain typically would provide the available local access service protocol(s) as discussed in section 2.4. In addition, this XRI Descriptor can provide a mapping of other XRIs that are synonymous to the resolved XRI authority.
All three options—next authority, local access, or synonyms—may be available at every step of resolution. For example, the XRI authority identifier `@a*b*c` may be the prefix to another XRI authority with the XRI `@a*b*c*d`. Or `@a*b*c` may be a local access endpoint itself, in which case its XRI Descriptor will contain references to local access services. Finally, this XRI Descriptor can also assert that the identifier `xri://@a*b*c` maps to the identifier `xri:///@!1000!2!3`, for example, in order to provide resolvers or caches with an equivalent persistent XRI.

<table>
<thead>
<tr>
<th>Page 7: [12] Deleted</th>
<th>Drummond Reed</th>
<th>3/ 6/ 2005 7:11:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>the relationships between authority descriptors, authorities, and authority identifier</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence, activity or other UML diagram instead? This has been marked as &quot;confusing&quot;, so if you agree and have a different way of presenting the information, PLEASE give ideas!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MUST no longer be relied upon.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MUST no longer be relied upon a</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 9: [16] Deleted</th>
<th>Drummond Reed</th>
<th>2/ 27/ 2005 6:06:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub-segment in the authority</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>in generic resolution. Required for trusted resolution</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A unique identifier for the authority being described in the Authority element.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 10: [19] Deleted</th>
<th>Drummond Reed</th>
<th>2/ 27/ 2005 6:09:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>extensions</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 10: [19] Deleted</th>
<th>Drummond Reed</th>
<th>2/ 27/ 2005 6:09:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>use</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 10: [19] Deleted</th>
<th>Drummond Reed</th>
<th>2/ 27/ 2005 6:09:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>in</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 10: [19] Deleted</th>
<th>Drummond Reed</th>
<th>2/ 27/ 2005 6:09:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>in the XRI Descriptor</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>more</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$res*localaccess</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
XRI resolution

the Service

the services defined in this document

Future extensions

content of this attribute

as

XRI authority

authority identifiers

in the ABNF,

See the description of XRIdescriptors/Synonyms/External below for the difference between internal and external synonym XRIs.

XRIs

that is
alternative

that is described

another, external,

The

XRI serves a slightly different purpose than an internal synonym XRI. Resolution of an internal synonym XRI typically results in an XRIDescriptor containing the same information as the one in which the internal synonym element appears.

, on the other hand,

s

escriptor

3.2.8

It

an

escriptor

Descriptor

e

then
indicating no specific trust mechanism

The trusted resolution mechanism defined in this document is implemented through additional elements as described in section 3.

Formatted Bullets and Numbering

qualified

or a cross-reference.

associated community must have published a

XRI Descriptor, called the "

", that contains one or more Authority/Service elements with URIs.

the Authority section of this XRID contains Authority/

with a value other than the default "xri://$res* (which is the default authority resolution service type and indicates the use of the authority resolution protocol described in this document)

, the Authority/URI element(s) must contain HTTP

declaring the root authority resolvers for the community
it is important to note that if following the GCS character

n exclamation point ( 

) 

* 

reassignable 

n asterisk ( 

) 

, 

a asterisk 

demonstrate 

of such a sub-segment in the case of

character
based on HTTP and HTTPS.

This present by the initiated using the root XRID representing the community root authority as described in the previous section. The process for determining the next authority descriptor (and thus discovering the next authority to communicate with) begins by constructing the “Next Resolution URI”. This URI is used for resolving one or more of the remaining XRI authority identifier sub-segments. This

present with an authority resolving

presents multiple sub-segments to an authority resolver

authority resolver

authority resolution

for the additional sub-segments
Any resolution beyond the first sub-segment is considered "lookahead" and is performed on behalf of the resolving client.

In lookahead resolution,

In this case, the resolver acts.

resolver

resolvers corresponding to

the authority resolver

will
s
the
s
the authority

resolver

, in the same order as they appear

, 

in the same order

appeared

It is up to the authority resolver to determine how many

to resolve on behalf of the resolving

(which it must resolve because it is the responsible authority)
demonstrates an XRI only

Thus, an XRI Authority identifier

the entire unresolved portion of the XRI Authority segment

at which the default HTTP/HTTPS authority resolution protocol is performed

At each step, this "Next Authority URI" is used to resolve a sub-segment in the current context.

another identifying
Resolution

list of pieces of information

XMLPath

XMLPath

this list of

s which

always

( )

the clarification regarding cross-references in

that forms the base of the Next Authority URI is the value of a/ URI element found at element path XRIDescriptor/Authority/URI in the XRI Descriptor.
As noted above, if there is no separator character preceding the sub-segment, a "**" MUST be added when creating the qualified sub-segment.

As shown in

a resolving client may attempt to resolve the entire authority resolution identifier (or any leading subset of it)

If the XRI Authority resolver URI for "xri://@" is determined (by configuration) to be "http://at.example.com/xri-authority/", then

assume the "@" authority responds to resolution requests at http://at.example.com/xri-authority/. The resolving client wishes the "@" authority to . In this case, the Next Authority URI would be the concatenation of "http://at.example.com/xri-authority/" with "**a*b*c", yielding "http://at.example.com/xri-authority/*a*b*c"

. As described above, this resolving authority may choose to only provide Descriptors for 1) *a, 2) *a and *a*b, or 3) *a, *a*b and *a*b*c.

which is

(a list of one or more XRI authority identifier sub-segments)
xa-uri =

<table>
<thead>
<tr>
<th>Page 13</th>
<th>Deleted</th>
<th>Gabe Wachob</th>
<th>3/4/2005 10:47:00 AM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>append</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Page 13</td>
<td>Deleted</td>
<td>Gabe Wachob</td>
<td>3/4/2005 10:47:00 AM</td>
</tr>
<tr>
<td></td>
<td>path portion of xa-uri</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Page 13</td>
<td>Deleted</td>
<td>Dave McAlpin</td>
<td>2/22/2005 5:10:00 PM</td>
</tr>
<tr>
<td></td>
<td>else:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>xa-uri  = xa-uri + separator</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Page 13</td>
<td>[73] Formatted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:08:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Normal, Don't keep with next</td>
</tr>
<tr>
<td>Page 13</td>
<td>[74] Deleted</td>
<td>Dave McAlpin</td>
<td>2/22/2005 5:16:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>either</td>
</tr>
<tr>
<td>Page 13</td>
<td>[74] Deleted</td>
<td>Dave McAlpin</td>
<td>2/22/2005 5:16:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>with</td>
</tr>
<tr>
<td>Page 13</td>
<td>[75] Deleted</td>
<td>Dave McAlpin</td>
<td>3/5/2005 8:45:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>new</td>
</tr>
<tr>
<td>Page 13</td>
<td>[75] Deleted</td>
<td>Dave McAlpin</td>
<td>3/5/2005 8:46:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Ds</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Used to be section 2.2.9 and it has also been modified</td>
</tr>
<tr>
<td>Page 15</td>
<td>[77] Deleted</td>
<td>Dave McAlpin</td>
<td>3/5/2005 9:18:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ir</td>
</tr>
<tr>
<td>Page 15</td>
<td>[77] Deleted</td>
<td>Dave McAlpin</td>
<td>3/5/2005 9:18:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>s</td>
</tr>
<tr>
<td>Page 15</td>
<td>[78] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:19:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>Page 15</td>
<td>[78] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:19:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>It</td>
</tr>
<tr>
<td>Page 15</td>
<td>[79] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:19:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>. In this case,</td>
</tr>
<tr>
<td>Page 15</td>
<td>[79] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:19:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>,</td>
</tr>
<tr>
<td>Page 15</td>
<td>[79] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:19:00 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>with</td>
</tr>
<tr>
<td>Page 15</td>
<td>[80] Deleted</td>
<td>Drummond Reed</td>
<td>3/6/2005 10:21:00 PM</td>
</tr>
</tbody>
</table>
somewhat novel

Thus, the return result is indeed an “explanation of the error situation”.

Passing the error code back to the resolving client aids the client (and the proxy/lookahead resolver) by giving it information it needs in determining how to proceed. If the resolving client weren’t given an error code, it might decide to continue resolution where the proxy or lookahead resolver left off, not knowing that the resolution ended in an error. The resolving client would then reproduce the same request that already produced the error when performed by the proxy/lookahead resolver.

Both an an would be participating in and of a to be
An XRIDescriptor may contain an XRIDescriptor/Synonyms/External element and not contain any XRIDescriptor/Authority or XRIDescriptor/LocalAccess elements. This is called an "XRI redirect" because the XRIDescriptor is effectively redirecting to a new XRI Authority. In this case, the unresolved portion of the original XRI (i.e. the XRI being resolved) is added to contents of the /XRIDescriptor/Synonyms/External element to create a new XRI. This new XRI is then resolved as described in Section 2.2.2 of this document.

The example in Section 2.2.6 demonstrates the resolution of xri://=example*home*base/foo*bar. The first request is to " http://equals.example.org/xri-resolve. If the response had been as follows, a new XRI would be constructed as xri://=example2*home*base/foo*bar and the resolution process would start again with this new XRI.

```
200 OK HTTP/1.1
Content-Type: application/xrid+xml
Expires: Fri, 7 Nov 2003 19:43:31 GMT
<other HTTP headers>

<xRIDescriptors xmlns="...">  
  <XRIDescriptor>  
    <Resolved>*example</Resolved>  
    <Synonyms>  
      <External>  
      </External>  
    </Synonyms>  
  </XRIDescriptor>  
</xRIDescriptors>
```

Lookahead resolution may be performed on a subset of the list of authority sub-segments. Proxied resolution is performed on the entire authority identifier.

Lookahead resolution never includes the initial authority sub-segment of an authority identifier (since at least the first authority performing lookahead resolution is identified by the first sub-segment). Proxied resolution always includes the first authority identifier sub-segment since proxied resolution resolves the entire authority identifier.

that specifies the location of the authority resolution endpoint
with which to perform local access

This section defines the default protocol for retrieving an XRI Descriptor with the IRI authority form of authority.

This process consists of

URL

out of the

and performing a HTTP GET request on that HTTP URL

that results in an XRIDescriptors document containing one XRI Descriptor for the authority

That

constructed from the request

to the front of it

An

(Note that this is not defined – see section 3)
The would be

and

be performed at

This section defines the construction of URIs for the X2R local access service.

of type “xri://$res*local.access/X2R”

xri:Service[Type='xri://$res*localaccess/X2R']/URI f

from the XRI Descriptor (section 2.2.3)

which results from the resolution of the authority part of the XRI. This URI is concatenated

Normal, Don't keep with next

Normal, Don't keep with next

Normal, Don't keep with next
Using a Cross-Reference to Specify a Representation Type

As an alternative to using a Service/MediaType element, a cross-reference MAY be used to specify a desired resource representation type when performing local access. The XRI namespace “$res*type” is reserved for this purpose. This specification does not enumerate such types; they may be further defined in the [XRIMetadata] or other specifications, or a list or “dictionary” of such XRIs may be published by the identifier authority or community.

To specify a particular resource representation type using “$res*type” cross-reference is appended to the XRI during a local access request. For example, an RDDL document could be specified by appending the cross-reference “($res*type/RDDL)”.

The following example using the X2R local access service illustrates this technique. Assuming the original XRI being resolved is “xri://=example*home/foo*bar” and the local access URI is “http://xri.example.com/xri-local/”, the following HTTP request would request the RDDL document describing this resource:

```
GET /xri-local/foo*bar/($res*types%2FRDDL) HTTP/1.1
<other HTTP headers>
```
Note that the cross-reference is escaped per the rules for the URI-

normal form of an XRI.

The resulting HTTP response might be:

```
200 OK HTTP/1.1
<cache-headers>
<other HTTP headers>

<content of representation of RDDL for xri:=example*home/foo*bar>
```

X2R local access servers MAY return a 404 HTTP status code if they do not have an appropriate representation of the resource, or if they do not recognize the use of the cross-reference to specify a representation type.

Note that proxied and lookahead resolution may reduce the number of http cache hits that occur during resolution. It is expected, however, that the benefit in lookahead and proxied resolution, with the reduction of round trip HTTP interactions, will more than compensate for the lack of HTTP caching benefits.

During the X2R local access HTTP interaction, redirects may be returned, and the “Location” field may contain an HTTP normal form. This use of redirects constitutes a mapping facility that allows one XRI to resolve into another XRI synonym during local access. If the local access server is aware of the HTTP
(S) URI where the XRI may be accessed, it can provide a “Location” header containing an HTTP

(S) URI. In this case, it SHOULD provide an “X-XRI-Canonical” header (see below) to describe the target XRI that the redirection is targeting

If the local access server knows only the target XRI, then it MUST return a redirection header (3XX code) with the “Location” field containing an XRI.

Content-Location

“Content-Location” may be used during local access where the resource being accessed is an “attribute” or “view” of another resource. This usually would occur in the case where metadata is being accessed using a trailing cross reference to an XRI value under the “$res$type” namespace (see section 2.4.2.2). Such a “Content-Location” header would specify where the resource itself may be accessible (rather than the metadata). This is not required and MUST NOT be required by resolving clients for proper operation. The content-location SHOULD be an HTTP

(S) URI if the local access server is aware of the HTTP

(S) location, otherwise it MAY be an XRI.
### X-XRI-Canonical

This header may be present only in HTTP (S) redirects while performing the X2R local access service. Its purpose is to notify a resolving client that the redirect is occurring because the original XRI is a mapping to another XRI. The value of this header is the target XRI in URI-normal form. This header MAY be present even when the Location: header is present and contains an XRI. This header SHOULD be present when the Location: header is present and contains a HTTP(S) or other URI.

**Form:**

```
X-XRI-Canonical: <xri-in-uri-normal-form>
```
This section gives a brief overview and an example of trusted resolution using HTTP. Specific processing rules are defined in Section 3.3. The basic approach to

simple change

a content type of "application/xrid-t+xml" is requested using the HTTP mechanism instead of "application/xrid+xml"

(as defined in section 3.2 of this document),

steps through resolution of the authority portion of

The example that follows shows the equivalent process using trusted resolution.

good practice, and one followed by the global communities rooted on @ and =,
in the Authority/URI element

, found in the xri:XRI Descriptor/xri:Authority/xri: URI element of the XRI Descriptor for the global community rooted on =.

known and configured in advance

. If the community root publishes an XRI Descriptor, the AuthorityID

In

that

. If the community root publishes a signed XRI Descriptor,

information about
Several elements in XRI Descriptors are either added to XRI Descriptors or that are optional or have limited usage in standard resolution are required for trusted resolution. These elements allow resolving clients to verify the Descriptor in which they are contained:

- **AuthorityID** and

```
xri: AuthorityID
```

Required when providing trusted resolution, optional otherwise.

```
xri:
```

Required when providing trusted resolution.

```
AuthorityID
```

A unique identifier for the authority that produced this Descriptor, of type xs:anyURI. If present, the value of this attribute MUST be such that there is negligible probability that the same value will be assigned as an identifier to any other authority. Note that the authority identified by this element is the describing authority (the producer of the current XRID), NOT the authority described by the
XRID. The latter is specified in the
xri:XRI_Descriptor/xri:Authority/xri:AuthorityID element (see below).

Rather, this AuthorityID element identifies the
authority that produced this Descriptor
that describes another authority.

Reference enclosed
has not been modified from the time it was published by the describing authority, and that
the describing authority believes the information to be correct.
Several descriptors are added.

Described by this xri:Authority element, of type xs:anyURI. If present, the value of this attribute MUST be such that there is negligible probability that the same value will be assigned as an identifier to any other authority. This element is correlated to the xri:XRIDescriptor/xri:AuthorityID element corresponding to a resolution result from the described Authority.
To accomplish this, a chain of identifiers between describing authority and described authority is created that is independent of the sub-segments being resolved. Consider the following scenario: Imagine that ExampleCorp acts as the both global community root and uses the same key to sign for both the @ and = namespaces. ExampleCorp's public key is described in a certificate associated with example.com. ExampleCorp responds to resolution requests in the @ namespace at http://at.xri.example.com and to resolution requests in the = namespace at http://equals.xri.example.com. A client attempts to resolve xri://@example by sending a request to http://at.xri.example.com. The client receives an XRI Descriptor, properly signed by example.com, with an xrid:Resolved element of "@example". Although the response appears to be valid, the XRI Descriptor is in fact fraudulent. A malicious party intercepted the request and sent it to http://equals.xri.example.com instead of to the intended http://at.xri.example.com. The XRI Descriptor describes =example, not @example.

To detect In trusted resolution as defined by this document, In the example above, the two XRI Authorities responsible for the @ and = namespaces, respectively, each has its own unique AuthorityID. Because the client requested resolution in the @ namespace, it knows the AuthorityID associated with the XRI Authority responsible for @. With this knowledge, the client detects that the XRI Descriptor is not provided by @ authority because the value of xrid:AuthorityID is incorrect.
xrid:XRIDescriptor element in the XRI Authority's

[[@@ Should we discuss restrictions on HTTP redirects, particularly redirects that contain an XRI?]

List Bullet .25" indent, No bullets or numbering

For complete trusted resolution, each XRI Descriptor in a resolution chain MUST be validated. While
xri:

subject of the digital signature

The digital signature MUST be verified using the responding authority’s “expected key” as defined below.

XMLPath

That is

. This XRI Descriptor from Authority A

info

verify

T
and the one that must

For the initial iteration of resolution (e.g. the authority sub-segments resolved at a global community root), the signer's authority

For the initial iteration of resolution ( 
<table>
<thead>
<tr>
<th>Page 34: [229] Deleted</th>
<th>Drummond Reed</th>
<th>3/6/2005 11:44:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub-segments resolved at a global community root)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 34: [229] Deleted</th>
<th>Drummond Reed</th>
<th>2/28/2005 5:00:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>authority</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 34: [231] Formatted</th>
<th>Dave McAlpin</th>
<th>2/24/2005 12:16:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMLPath</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 34: [231] Formatted</th>
<th>Dave McAlpin</th>
<th>2/24/2005 12:16:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMLPath</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 34: [232] Formatted</th>
<th>Dave McAlpin</th>
<th>2/24/2005 12:17:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMLPath</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 34: [232] Formatted</th>
<th>Dave McAlpin</th>
<th>2/24/2005 12:17:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMLPath</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>xri:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 35: [235] Deleted</th>
<th>Drummond Reed</th>
<th>2/28/2005 5:01:00 PM</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Page 35: [235] Deleted</th>
<th>Drummond Reed</th>
<th>2/28/2005 5:01:00 PM</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Page 35: [236] Deleted</th>
<th>Drummond Reed</th>
<th>2/28/2005 5:01:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>attempt</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 35: [236] Deleted</th>
<th>Drummond Reed</th>
<th>3/6/2005 11:45:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution may continue and</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Page 35: [236] Deleted | Drummond Reed | 2/28/2005 5:01:00 PM |
“trusted”

mechanism

3 of [XRI Syntax]

xri-t+xml

xri:

at the root of the XRI Descriptor being signed

should

not

should

XMLPath

XMLPath

, Descriptor

typically

xri:
[XML DigSig has this to say about KeyInfo: “KeyInfo indicates the key to be used to validate the signature. Possible forms for identification include certificates, key names, and key agreement algorithms and information – we define only a few. KeyInfo is optional for two reasons. First, the signer may not wish to reveal key information to all document processing parties. Second, the information may be known within the application’s context and need not be represented explicitly.” Does this argument for making KeyInfo optional apply to us? GMW: I don’t think it should be optional.]

What happens if the AuthorityID isn’t provided? Should this be a MUST?

Thus,

xri://$res*schema/XRIDescriptor

-Resolution

an

would

In doing this

the container and

the

are
is

In other words, if the consumer is

declared as the extension, the consumer will

be shielded from even being

not even be aware that elements in the XRI-Resolution schema are included, and thus they will be ignored.

XMLPath

other" namespace (specifically

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath

XMLPath
but shows that
Thus, and the mechanics of expressing and processing version information

Each r will have a new version number

behavior or the definition of data to be transmitted in the process of resolution

A new release occurs when there is some non-editorial change to the document specifying a change in behavior or a change in the definition of data on the wire with respect to resolution
to be transmitted in the process of resolution.

then change only the

but with the previous major version number
ese version attributes, while having a fixed value in XRI 2.0, may be used to indicate in future versions of the XRI specification that the relevant element is conformant to a specific version of the XRI specification.

<table>
<thead>
<tr>
<th>Page 38: [284] Deleted</th>
<th>Gabe Wachob</th>
<th>3/4/2005 4:04:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>this specification</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>parent</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>should</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>not change</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 38: [286] Deleted</th>
<th>Drummond Reed</th>
<th>2/28/2005 5:54:00 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>, in the interest of namespace stability</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I don't get this. Why do we go to the trouble of defining trusted resolution if we turn around and say you should use DNSSEC? If for some reason we DO want to make this point (which I don't think we should), isn't SSL or some other TLS approach more appropriate? We should also note that this concern does not apply to trusted res.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 40: [288] Deleted</th>
<th>Drummond Reed</th>
<th>3/7/2005 12:15:00 AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>or server authenticated HTTPS should be considered</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 40: [289] Deleted</th>
<th>Drummond Reed</th>
<th>3/7/2005 12:14:00 AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>, when unambiguous and authoritative responses are paramount</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 40: [290] Formatted</th>
<th>Dave McAlpin</th>
<th>3/6/2005 8:16:00 AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space Before: 4 pt, After: 4 pt</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 40: [291] Formatted</th>
<th>Dave McAlpin</th>
<th>3/6/2005 8:16:00 AM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space Before: 4 pt, After: 4 pt</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Page 40: [292] Deleted</th>
<th>Drummond Reed</th>
<th>3/7/2005 12:21:00 AM</th>
</tr>
</thead>
</table>
**Trusted Resolution**

While the trusted resolution mechanism specified in this document provides a way to verify the integrity of a successful XRI resolution, it does not provide a way to verify the integrity of a resolution failure. Reasons for this limitation include non-malicious network failures, DOS attacks, and the ability for a MITM attacker to modify HTTP responses when resolution is not performed over HTTPS.

Additionally, there is no revocation mechanism specified for trusted resolution. Therefore, a signed resolution's validity period should be limited appropriately to mitigate the risk of an incorrect or invalid resolution.

DNS Spoofing

As this system is dependant on the DNS, one cannot be sure that the information one gets back from DNS, and subsequently, the XRI resolution response, is no more secure than original DNS query. To solve that, the use of DNSSEC [DNSSEC] for securing and verifying zones is recommended, and the use of trusted resolution as defined in this specification or server authenticated HTTPS should be considered, when unambiguous and authoritative responses are paramount.

HTTP Security

Much of the security considerations set forth in HTTP/1.1 [RFC2616] apply to XRI Resolution protocols defined here. Special consideration should be given proxy and caching behaviours to ensure accurate and reliable responses from resolution requests. In particular, network topologies increasingly have transparent proxies, for various reasons, some of which may insert VIA and other headers as a consequence, or may even cache content without regard to proper caching policies set for by the HTTP authority for a resource.

Caching Authorities

In addition to traditional HTTP caching proxies, XRI resolution authority proxies may exist in the resolution topology. These entities are especially instructed to take precautions against cache poisoning (as described in [RFC7838]), as these caching entities may represent trust decision points within a deployments resolution architecture.

LookAhead and Proxy Resolution

- special trust considerations for in-network resolvers

Well-Know locations

- secure distribution of community root keys and authority URI's (perhaps trusted XRID?)
DDOS
SAML Considerations
Interpreting Network Failures
Discuss that network failures are NOT to be considered negative resolution results.
Discuss how HTTP and XRID-based redirects introduce new opportunities for denial of service?

This entire document deals with security considerations related to XRI Authority resolution.

Here’s some text from DNSSEC that might be a useful model for this section:

This document specifies extensions to the Domain Name System (DNS) protocol to provide data integrity and data origin authentication, public key distribution, and optional transaction and request security.

It should be noted that, at most, these extensions guarantee the validity of resource records, including KEY resource records, retrieved from the DNS. They do not magically solve other security problems. For example, using secure DNS you can have high confidence in the IP address you retrieve for a host name; however, this does not stop someone for substituting an unauthorized host at that address or capturing packets sent to that address and falsely responding with packets apparently from that address. Any reasonably complete security system will require the protection of many additional facets of the Internet beyond DNS.

The implementation of NXT RRs as described herein enables a resolver to determine all the names in a zone even if zone transfers are prohibited (section 5.6). This is an active area of work and may change.

A number of precautions in DNS implementation have evolved over the years to harden the insecure DNS against spoofing. These precautions should not be abandoned but should be considered to provide additional protection in case of key compromise in secure DNS.