XEP-0041: Reliable Entity Link

Justin Karneges
mailto:justin@affinix.com
xmpp:justin@andbit.net

2003-09-30
Version 0.2

Status	Type	Short Name
Retracted	Standards Track	rel

Protocol for linking a bytestream between two Jabber entities.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 by the XMPP Standards Foundation (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF's Intellectual Property Rights Policy (a copy of which can be found at <https://xmpp.org/about/xsf/ipr-policy> or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

Contents

1 Overview

1.1 Introduction

Reliable Entity Link (or simply 'REL'), is a system for coordinating reliable bytestreams between two Jabber entities for the purpose of keeping applications (and application specifications) simple. However, this proposal does not define any specific bytestream protocol. It is expected that there will be multiple ways to obtain a bytestream between Jabber entities (thru-server and peer-to-peer are two methods that come to mind), but applications can refer to REL instead of some particular stream transport.

1.2 Stream transport properties

A REL-compatible stream transport must have the following properties:

- Provides a reliable bytestream between two Jabber entities, which means that the bytestream transport handles all data delivery issues, such that the application need not worry about them.

- Has link states from the following table.

- Defines a stream identifier, which MUST have a unique ASCII representation. The stream protocol MUST be able to use any ASCII identifier chosen during REL negotiation, as long as the sending party doesn't use the same identifier more than once.

Code	Description
INIT	Initiation
GOOD	Successful initiation (connected)
BAD	Unsuccessful initiation (stream is closed, no further state)
CLOS	Successful closure after establishment (stream is closed, no further state)
ERR	Link failure after establishment (stream is closed, no further state)

The following stream transports that meet these guidelines are:

Short name	Protocol
ibb	In-Band Bytestreams (XEP-0047) XEP-0047: In-Band Bytestreams <https://xmpp.org/extensions/xep-0047.html>.
s5b	SOCKS5 Bytestreams (XEP-0065) XEP-0065: SOCKS5 Bytestreams <https://xmpp.org/extensions/xep-0065.html>.

2 Usage

2.1 Service discovery

Before using REL, ensure it is a supported service of the remote entity by using Service Discovery (XEP-0030) [1]:

Listing 1: Requesting disco information

```
<iq type="get" to="joe@blow.com/Home" id="sd_1">
  <query xmlns="http://jabber.org/protocol/disco#info"/>
</iq>
```

The remote entity will advertise the "http://jabber.org/protocol/rel" namespace as a feature to represent they implement this protocol.

Listing 2: Response

```
<iq type="result" from="joe@blow.com/Home" id="sd_1">
  <query xmlns="http://jabber.org/protocol/disco#info">
    <feature var="http://jabber.org/protocol/rel"/>
  </query>
</iq>
```

2.2 Obtaining a REL context

To use REL, the entities must obtain a REL Context ID (or *cid*) through some action. A *cid* is simply an opaque alphanumeric string. For example, perhaps the link is needed for a file transfer:

Listing 3: Possible File Transfer

```
<iq type="set" id="ft_1" to="joe@blow.com/Home">
  <query xmlns="filexfer" filename="coolfile.txt"/>
</iq>
```

Listing 4: Possible response

```
<iq type="result" id="ft_1" from="joe@blow.com/Home">
  <query xmlns="filexfer">
    <cid xmlns="http://jabber.org/protocol/rel" value="myCID"/>
  </query>
</iq>
```

---

[1]XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.

All high-level protocols that use Reliable Entity Link MUST have a way of providing such a cid. The cid must be unique among all other REL cids between the two entities.

2.3 Selecting a Stream

The next step is to ask the remote entity which stream method it would like to use. We will use Feature Negotiation (XEP-0020) [2] for this. The streams are listed using the short names from the table of supported streams.

Listing 5: Selecting a stream

```
<iq type="get" id="rel_1" to="joe@blow.com/Home">
  <query xmlns="http://jabber.org/protocol/rel" cid="myCID" keepAlive=
      'true'>
    <feature xmlns="http://jabber.org/protocol/feature-neg">
      <x xmlns="jabber:x:data">
        <field var="method" type="list-single">
          <option><value>s5b</value></option>
          <option><value>ibb</value></option>
        </field>
      </x>
    </feature>
  </query>
</iq>
```

The keepAlive attribute indicates that the initiator is planning on trying another method if the one selected here is to fail. An entity SHOULD use keepAlive for all attempts but the last for a given application. If keepAlive is omitted, then it is considered false.
The remote entity will then agree on a method:

Listing 6: Possible response

```
<iq type="result" id="rel_1" from="joe@blow.com/Home">
  <query xmlns="http://jabber.org/protocol/rel" cid="myCID">
    <feature xmlns="http://jabber.org/protocol/feature-neg">
      <x xmlns="jabber:x:data" type="submit">
        <field var="method">
          <value>s5b</value>
        </field>
      </x>
    </feature>
  </query>
</iq>
```

Or maybe an error:

---

[2]XEP-0020: Feature Negotiation <https://xmpp.org/extensions/xep-0020.html>.

Listing 7: Error

```
<iq type="error" id="rel_1" from="joe@blow.com/Home">
  <error code="501">No supported protocols.</error>
</iq>
```

If the entity returns error, then the REL cid is invalidated and the application fails. If a stream method has been chosen successfully, then now it must be initiated using the REL cid as the stream's identifier (the stream goes into **INIT** state).

On **GOOD**: This indicates the stream is ready for use within the original context, and data exchanged over the stream is to be left up to the application.

On **BAD**: If the keepAlive="true" attribute was specified, then the initiator MUST repeat this section over again to attempt with a different method. If keepAlive was not specified, then the REL cid is invalidated and the application fails.

On **CLOS** or **ERR**, the REL cid is invalidated.

3 Security Considerations

There are no security considerations.

4 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [3].

5 XMPP Registrar Considerations

The XMPP Registrar [4] shall register the 'http://jabber.org/protocol/rel' namespace as a result of this document.

6 XML Schema

---

[3]The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.

[4]The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <https://xmpp.org/registrar/>.

```
<?xml version='1.0' encoding='UTF-8'?>

<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='http://jabber.org/protocol/rel'
    xmlns='http://jabber.org/protocol/rel'
    elementFormDefault='qualified'>

  <xs:element name='query'>
    <xs:complexType>
     <xs:attribute name='cid' type='xs:string' use='required'/>
     <xs:attribute name='keepAlive' type='xs:boolean' use='optional'/
         >
    </xs:complexType>
  </xs:element>

</xs:schema>
```