XEP-0234: Jingle File Transfer

Peter Saint-Andre
mailto:stpeter@jabber.org
xmpp:stpeter@jabber.org
https://stpeter.im/

2012-02-08
Version 0.15

Status	Type	Short Name
Experimental	Standards Track	NOT_YET_ASSIGNED

This specification defines a Jingle application type for transferring files between two entities. The protocol provides a modular framework that enables the exchange of information about the file to be transferred as well as the negotiation of parameters such as the transport to be used.

Legal

Copyright

Permissions

Warranty

Liability

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 <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

Contents

1 Introduction

SI File Transfer [1] was the original XMPP protocol extension for file transfer negotiation. However, that protocol has several drawbacks, most related to the Stream Initiation [2] protocol on which it depends:

1. It does not enable a true, bidirectional negotiation; instead, the initiator sets the terms for the file transfer and the responder either accepts the terms or cancels the negotiation.
2. It is the only technology in the Jabber/XMPP protocol "stack" that uses XEP-095: Stream Initiation. More modern technologies such as voice and video session negotiation use Jingle [3], and it would be helpful if implementors could re-use the same code for all negotiation use cases.

To overcome these drawbacks, this specification defines a file transfer negotiation method that meets the following requirements:

- Use the session negotiation semantics from XEP-0166.
- Use Jingle SOCKS5 Bytestreams Transport Method [4] and Jingle In-Band Bytestreams Transport Method [5].
- Define a file description format that, unlike XEP-0096 enables hash agility (via Use of Cryptographic Hash Functions in XMPP [6]).
- Define a clear upgrade path from SI File Transfer to Jingle File Transfer.

Jingle file transfer is only as reliable as the transports on which it depends. In particular, SOCKS5 Bytestreams ("S5B") does not always result in NAT or firewall traversal. To work around that problem, this specification requires all implementations to support as a fallback mechanism In-Band Bytestreams ("IBB"), which usually results in a successful (if slow) file transfer.
Note: It is likely that a future version of this specification will also recommend implementation of a Jingle transport method that emulates the IETF's ICE-TCP technology, which is currently a work in progress (see TCP Candidates with Interactive Connectivity Establishment (ICE) [7]); however, a future Jingle ICE-TCP transport method is dependent on the outcome of IETF work in this area.

[1] XEP-0096: SI File Transfer <http://xmpp.org/extensions/xep-0096.html>.
[2] XEP-0095: Stream Initiation <http://xmpp.org/extensions/xep-0095.html>.
[3] XEP-0166: Jingle <http://xmpp.org/extensions/xep-0166.html>.
[4] XEP-0260: Jingle SOCKS5 Bytestreams Transport Method <http://xmpp.org/extensions/xep-0260.html>.
[5] XEP-0261: Jingle In-Band Bytestreams Transport Method <http://xmpp.org/extensions/xep-0261.html>.
[6] XEP-0300: Use of Cryptographic Hash Functions in XMPP <http://xmpp.org/extensions/xep-0300.html>.
[7] TCP Candidates with Interactive Connectivity Establishment (ICE) <http://tools.ietf.org/html/draft-ietf-mmusic-ice-tcp>. Work in progress.

2 How It Works

This section provides a friendly introduction to Jingle file transfer.
First, the party that wishes to initiate the file transfer determines the responder's capabilities (via Service Discovery [8] or Entity Capabilities [9]). Here we assume that the responder supports the following service discovery features:

- urn:xmpp:jingle:1 as described in XEP-0166
- urn:xmpp:jingle:apps:file-transfer:3 as defined in this document If the protocol defined in this specification undergoes a revision that is not fully backwards-compatible with an older version, the XMPP Registrar shall increment the protocol version number found at the end of the XML namespaces defined herein, as described in Section 4 of XEP-0053.
- urn:xmpp:jingle:transports:s5b:1 as defined in XEP-0260
- urn:xmpp:jingle:transports:ibb:1 as defined in XEP-0261

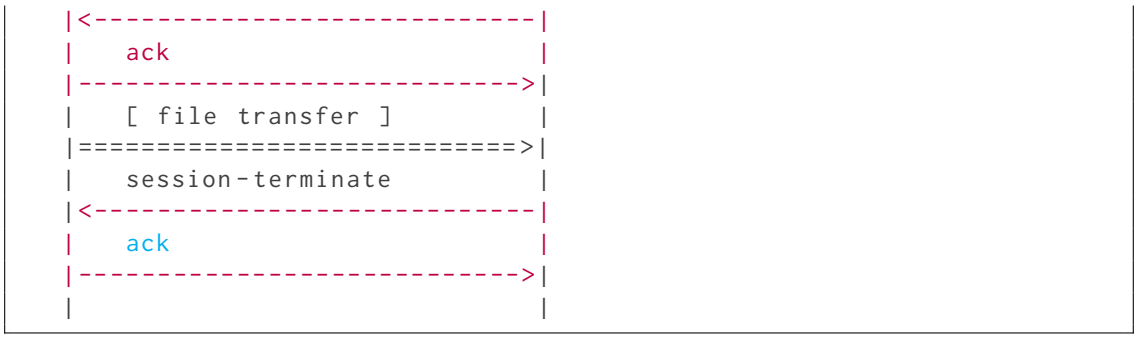
The initiator then sends a Jingle session-initiation request to a potential responder. The content-type of the request specifies two things:

1. An application type of "urn:xmpp:jingle:apps:file-transfer:3". In particular, the <description/> element contains an <offer/> element or a <request/> element that in turn contains one or more <file/> elements defining a file to be sent.
2. An appropriate transport method. So far the suggested methods are jingle-s5b (XEP-0260) and, as a fallback, jingle-ibb (XEP-0261).

Note: All attributes of the <file/> element are defined in XEP-0096, not in this specification.
In this example, the initiator is <romeo@montague.lit>, the responder is <juliet@capulet.lit>, the application type is a file offer, and the transport method is jingle-s5b.
The flow is as follows.

```
Initiator                    Responder
   |                             |
   |   session-initiate          |
   |   (with <offer/> or         |
   |   <request/> and S5B)       |
   |---------------------------->|
   |   ack                       |
   |<----------------------------|
   |   session-accept            |
```

[8] XEP-0030: Service Discovery <http://xmpp.org/extensions/xep-0030.html>.
[9] XEP-0115: Entity Capabilities <http://xmpp.org/extensions/xep-0115.html>.



First the initiator sends a Jingle session-initiate.

Listing 1: Initiator sends session-initiate

```
<iq from='romeo@montague.lit/orchard'
    id='nzu25s8'
    to='juliet@capulet.lit/balcony'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@montague.lit/orchard'
          sid='851ba2'>
    <content creator='initiator' name='a-file-offer'>
      <description xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
        <offer>
          <file>
            <date>1969-07-21T02:56:15Z</date>
            <desc>This is a test. If this were a real file...</desc>
            <name>test.txt</name>
            <range/>
            <size>1022</size>
            <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
                da749930852c69ae5d2141d3766b1</hash>
          </file>
        </offer>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                 mode='tcp'
                 sid='vj3hs98y'>
        <candidate cid='hft54dqy'
                   host='192.168.4.1'
                   jid='romeo@montague.lit/orchard'
                   port='5086'
                   priority='8257636'
                   type='direct'/>
        <candidate cid='hutr46fe'
                   host='24.24.24.1'
                   jid='romeo@montague.lit/orchard'
```

```
                    port='5087'
                    priority='8258636'
                    type='direct'/>
      </transport>
    </content>
  </jingle>
</iq>
```

Note: As in XEP-0096, inclusion of the <range/> child of the <file/> element indicates that the initiatior supports ranged transfers as described below under Ranged Transfers.
Note: Computing the hash of the file before sending it can slow down the process of file transfer, since the sending application needs to process the file twice. The sender might prefer to send the hash after the file transfer has begun, using a transport-info message as described under Communicating the Hash.
The responder immediately acknowledges receipt of the Jingle session-initiate.

Listing 2: Responder acknowledges session-initiate

```
<iq from='juliet@capulet.lit/balcony'
    id='nzu25s8'
    to='romeo@montague.lit/orchard'
    type='result'/>
```

The initiator then attempts to initiate a SOCKS5 Bytestream with the responder as described in XEP-0260 and XEP-0065. In the meantime, the responder returns a Jingle session-accept. In the session-accept message, the <file/> element MAY contain a <range/> element to indicate that the receiver also supports ranged transfers as described below under Ranged Transfers.

Listing 3: Responder sends session-accept

```
<iq from='juliet@capulet.lit/balcony'
    id='jn2vs71g'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-accept'
          initiator='romeo@montague.lit/orchard'
          sid='851ba2'>
    <content creator='initiator' name='a-file-offer'>
      <description xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
        <offer>
          <file>
            <date>1969-07-21T02:56:15Z</date>
            <desc>This is a test. If this were a real file...</desc>
            <name>test.txt</name>
            <range/>
            <size>1022</size>
```

```
            <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
                da749930852c69ae5d2141d3766b1</hash>
          </file>
        </offer>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                 mode='tcp'
                 sid='vj3hs98y'>
        <candidate cid='ht567dq'
                   host='192.169.1.10'
                   jid='juliet@capulet.lit/balcony'
                   port='6539'
                   priority='8257636'
                   type='direct'/>
        <candidate cid='hr65dqyd'
                   host='134.102.201.180'
                   jid='juliet@capulet.lit/balcony'
                   port='16453'
                   priority='7929856'
                   type='assisted'/>
        <candidate cid='grt654q2'
                   host='2001:638:708:30c9:219:d1ff:fea4:a17d'
                   jid='juliet@capulet.lit/balcony'
                   port='6539'
                   priority='8257606'
                   type='direct'/>
      </transport>
    </content>
  </jingle>
</iq>
```

The initiator acknowledges the Jingle session-accept.

Listing 4: Initiator acknowledges session-accept

```
<iq from='romeo@montague.lit/orchard'
    id='jn2vs71g'
    to='juliet@capulet.lit/balcony'
    type='result'/>
```

Once one client has successfully created a connection, it sends a <candidate-used/> element to the peer inside a Jingle transport-info message. If a client receives a candidate-used notification it SHOULD continue trying to connect to candidates sent by its peer if it has not tried all candidates with a higher priority than the one successfully used by the peer.

Listing 5: Initiator sends candidate-used in Jingle transport-info

```
<iq from='romeo@montague.lit/orchard'
    id='hjdi8'
```

```
    to='juliet@capulet.lit/balcony'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='transport-info'
          initiator='romeo@montague.lit/orchard'
          sid='a73sjjvkla37jfea'>
    <content creator='initiator' name='ex'>
      <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                 sid='vj3hs98y'>
        <candidate-used cid='hr65dqyd'/>
      </transport>
    </content>
  </jingle>
</iq>
```

The peer immediately acknowledges receipt.

Listing 6: Responder acknowledges candidate-used message

```
<iq from='juliet@capulet.lit/balcony'
    id='hjdi8'
    to='romeo@montague.lit/orchard'
    type='result'/>
```

(See XEP-0260 for further details.)
Now the parties exchange the file using the negotiated transport (here, SOCKS5 Bytestreams). Once the transfer is completed, either party can acknowledge completion (see Sending Multiple Files) or terminate the Jingle session; preferably this is done by the entity that receives the file to ensure that the complete file (up to the advertised size) has been received.

Listing 7: Receiver sends session-terminate

```
<iq from='juliet@capulet.lit/balcony'
    id='og61bvs98'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-terminate'
          sid='a73sjjvkla37jfea'>
    <reason>
      <success/>
    </reason>
  </jingle>
</iq>
```

After terminating the session, the parties would close the data transport as described in the relevant specification (e.g., XEP-0260 or XEP-0261).
For a description of the transport fallback scenario (from SOCKS5 Bytestreams to In-Band

Bytestreams), refer to XEP-0260.

3 Communicating the Hash

At any time during the lifetime of the file transfer session, the hosting entity (i.e., the entity where the file resides) can communicate the checksum of the file to the receiving entity. This is done by sending a session-info message containing a <checksum/> element qualified by the 'urn:xmpp:jingle:apps:file-transfer:3' namespace, which in turn contains a <file/> element that MUST at least contain a child element of <hash/> qualified by the 'urn:xmpp:hashes:1' namespace and MAY contain other elements qualified by the 'urn:xmpp:jingle:apps:file-transfer:3' namespace (e.g. <name/> and <date/>). Each <hash/> element contains a checksum of the file contents produced in accordance with the hashing function specified by the 'algo' attribute, which MUST be one of the functions listed in the IANA Hash Function Textual Names Registry [10].

Listing 8: Hosting entity sends hash in session-info

```
<iq from='romeo@montague.lit/orchard'
    id='kqh401b5'
    to='juliet@capulet.lit/balcony'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-info'
          initiator='romeo@montague.lit/orchard'
          sid='a73sjjvkla37jfea'>
    <checksum xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
      <file>
        <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
           da749930852c69ae5d2141d3766b1</hash>
      </file>
    </checksum>
  </jingle>
</iq>
```

4 Aborting a Transfer

If either party wishes to abort the transfer of a file but not the entire session (e.g., when the parties are exchanging multiple files), it SHOULD send a Jingle session-info message containing an <abort/> child element qualified by the 'urn:xmpp:jingle:apps:file-transfer:3' namespace.

[10] IANA registry of Hash Function Textual Names <http://www.iana.org/assignments/hash-function-text-names>.

Listing 9: Initiator aborts transfer of file

```
<iq from='romeo@montague.lit/orchard'
    id='hv9sx61j'
    to='juliet@capulet.lit/balcony'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-info'
          initiator='romeo@montague.lit/orchard'
          sid='a73sjjvkla37jfea'>
    <abort xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
      <file>
        <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
            da749930852c69ae5d2141d3766b1</hash>
      </file>
    </abort>
  </jingle>
</iq>
```

If either party wishes to end the entire file transfer session instead of aborting transfer of a particular file, it MUST instead send a session-terminate message containing a reason of <cancel/> as described in XEP-0166.

Listing 10: Receiver sends session-terminate

```
<iq from='juliet@capulet.lit/balcony'
    id='og61bvs98'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-terminate'
          sid='a73sjjvkla37jfea'>
    <reason>
      <cancel/>
    </reason>
  </jingle>
</iq>
```

After terminating the session, the parties would close the data transport as described in the relevant specification (e.g., XEP-0260 or XEP-0261).

5 Requesting a File

If the entity that hosts a file has advertised its existence (or if a previous file transfer attempt has failed and the receiver would like to initiate another attempt), the entity that wishes to receive the file can ”pull” the file from the hosting entity. This is done by sending a Jingle session-initiate to the hosting entity, including a <description/> element qualified by the

'urn:xmpp:jingle:apps:file-transfer:3' namespace and containing a <request/> element that defines the requested file.

Listing 11: Receiver requests hosted file

```
<iq from='juliet@capulet.lit/balcony'
    id='wsn361c3'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@montague.lit/orchard'
          sid='uj3b2'>
    <content creator='initiator' name='a-file-request'>
      <description xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
        <request>
          <file>
            <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
                da749930852c69ae5d2141d3766b1</hash>
          </file>
        </request>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                 mode='tcp'
                 sid='xig361fj'>
        <candidate cid='ht567dq'
                   host='192.169.1.10'
                   jid='juliet@capulet.lit/balcony'
                   port='6539'
                   priority='8257636'
                   type='direct'/>
        <candidate cid='hr65dqyd'
                   host='134.102.201.180'
                   jid='juliet@capulet.lit/balcony'
                   port='16453'
                   priority='7929856'
                   type='assisted'/>
        <candidate cid='grt654q2'
                   host='2001:638:708:30c9:219:d1ff:fea4:a17d'
                   jid='juliet@capulet.lit/balcony'
                   port='6539'
                   priority='8257606'
                   type='direct'/>
      </transport>
    </content>
  </jingle>
</iq>
```

The parties would then complete a session negotiation flow similar to that outlined above for offering a file.
Note: If the requesting entity knows the hash of the file, it can include only that metadata in its request. If not, the requesting entity needs to include enough metadata to uniquely identify the file, such as the date, name, and size. For similar considerations, see RFC 5547 [11].

6 Ranged Transfers

As in XEP-0096, a transfer can include only part of a file (e.g., to restart delivery of a truncated transfer session at a point other than the start of the file). This is done using the <range/> element from XEP-0096. The usage is illustrated in the following examples.
Let us imagine that the parties negotiate a file transfer session using, say, In-Band Bytestreams. During the transfer, the recipient goes offline unexpectedly and IBB stanzas from the sender to the recipient begin to bounce. When the recipient comes back online, the recipient could initiate a new Jingle session (to retrieve the file) and specify that it wants to receive all chunks after byte 270336 (which might be the 66th chunk of size 4096).

Listing 12: Receiver requests hosted file, with range

```
<iq from='juliet@capulet.lit/balcony'
    id='wsn361c3'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@montague.lit/orchard'
          sid='uj3b2'>
    <content creator='initiator' name='a-file-request'>
      <description xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
        <request>
          <file>
            <range offset='270336'/>
            <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>552
                da749930852c69ae5d2141d3766b1</hash>
          </file>
        </request>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                 mode='tcp'
                 sid='xig361fj'>
        <candidate cid='ht567dq'
                   host='192.169.1.10'
                   jid='juliet@capulet.lit/balcony'
```

[11] RFC 5547: A Session Description Protocol (SDP) Offer/Answer Mechanism to Enable File Transfer <http://tools.ietf.org/html/rfc5547>.

```
                    port='6539'
                    priority='8257636'
                    type='direct'/>
        <candidate cid='hr65dqyd'
                    host='134.102.201.180'
                    jid='juliet@capulet.lit/balcony'
                    port='16453'
                    priority='7929856'
                    type='assisted'/>
        <candidate cid='grt654q2'
                    host='2001:638:708:30c9:219:d1ff:fea4:a17d'
                    jid='juliet@capulet.lit/balcony'
                    port='6539'
                    priority='8257606'
                    type='direct'/>
      </transport>
    </content>
  </jingle>
</iq>
```

Alternatively, the sender could initiate a new file transfer, indicating that it supports ranged transfers, and in the Jingle session-accept message the receiver could indicate that it wants the transfer to begin at the specified offset.

7 Sending Multiple Files

The initiator can send multiple files by including multiple <file/> elements in its session-initiate message.

Listing 13: Initiator sends session-initiate with multiple files

```
<iq from='romeo@montague.lit/orchard'
    id='bv2gs986'
    to='juliet@capulet.lit/balcony'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@montague.lit/orchard'
          sid='h2va419i'>
    <content creator='initiator' name='a-file-offer'>
      <description xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
        <offer>
          <file>
            <date>2011-06-01T15:58:15Z</date>
            <name>somefile.txt</name>
            <size>1234</size>
```

```
            <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>
                a749930852c69ae5d2141d3766b1552d
            </file>
            <file>
              <date>2011-06-01T15:58:15Z</date>
              <name>anotherfile.txt</name>
              <size>2345</size>
              <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>930852
                c69ae5d2141d3766b1552da749</hash>
            </file>
            <file>
              <date>2011-06-01T15:58:15Z</date>
              <name>yetanotherfile.txt</name>
              <size>3456</size>
              <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>52
                c69ae5d2141d3766b1552da7499308</hash>
            </file>
          </offer>
        </description>
        <transport xmlns='urn:xmpp:jingle:transports:s5b:1'
                   mode='tcp'
                   sid='vj3hs98y'>
          <candidate cid='hft54dqy'
                     host='192.168.4.1'
                     jid='romeo@montague.lit/orchard'
                     port='5086'
                     priority='8257636'
                     type='direct'/>
          <candidate cid='hutr46fe'
                     host='24.24.24.1'
                     jid='romeo@montague.lit/orchard'
                     port='5087'
                     priority='8258636'
                     type='direct'/>
        </transport>
      </content>
    </jingle>
</iq>
```

The parties would negotiate the file transfer session as previously described.
After exchange of the first file, the recipient SHOULD send a Jingle session-info message indicating receipt of the complete file.

Listing 14: Receiver sends ack in session-info

```
<iq from='juliet@capulet.lit/balcony'
    id='jp2ba614'
    to='romeo@montague.lit/orchard'
    type='set'>
```

```
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-info'
          initiator='romeo@montague.lit/orchard'
          sid='a73sjjvkla37jfea'>
    <received xmlns='urn:xmpp:jingle:apps:file-transfer:3'>
      <file>
        <hash xmlns='urn:xmpp:hashes:1' algo='sha-1'>
            a749930852c69ae5d2141d3766b1552d</hash>
      </file>
    </received>
  </jingle>
</iq>
```

The hosting entity SHOULD NOT wait for arrival of the <received/> acknowledgement before starting to send the next file in its list.
After the recipient has received all of the files, it SHOULD send a final acknowledgement and then terminate the session.
After terminating the session, the parties would close the data transport as described in the relevant specification (e.g., XEP-0260 or XEP-0261).
OPEN ISSUE: Provide a way for the hosting entity to add more files to the original "manifest"?
Support for transferring multiple files is OPTIONAL. If an application supports multi-file exchange, it MUST advertise a service discovery feature of "urn:xmpp:jingle:apps:file-transfer:multi".

8 Use of Jingle Actions

Jingle file transfer uses only a few of the actions defined in XEP-0166. Jingle usage is summarized in the following table.

Action	Use
content-accept	Unused
content-add	Unused
content-modify	Unused
content-reject	Unused
content-remove	Unused
description-info	Unused
security-info	Unused
session-accept	Accepting a file offer or request
session-info	Communicating the file hash
session-initiate	Initiating a file offer or request
session-terminate	Ending a file transfer session
transport-accept	Accepting fallback from S5B to IBB
transport-info	Used in SOCKS5 Bytestreams

Action	Use
transport-reject	Rejecting fallback from S5B to IBB
transport-replace	Fallback from S5B to IBB

9 Implementation Notes

9.1 Mandatory to Implement Technologies

All implementations MUST support the Jingle In-Band Bytestreams Transport Method (XEP-0261) as a reliable method of last resort. An implementation SHOULD support other transport methods as well, especially the Jingle SOCKS5 Bytestreams Transport Method (XEP-0260).

9.2 Preference Order of Transport Methods

An application MAY present transport methods in any order, except that the Jingle In-Band Bytestreams Transport Method MUST be the lowest preference.

9.3 Migration from XEP-0096

Support for Jingle file transfer can be determined through discovery of the 'urn:xmpp:jingle:apps:file-transfer:3' namespace (see Namespace Versioning regarding the possibility of incrementing the version number), via either service discovery (XEP-0030) or entity capabilities (XEP-0115). If the initiator knows that the responder supports Jingle file transfer, it SHOULD first attempt negotiation using Jingle rather than Stream Initiation.

10 Determining Support

To advertise its support for the Jingle File Transfer, when replying to service discovery information ("disco#info") requests an entity MUST return URNs for any version of this protocol that the entity supports -- e.g., "urn:xmpp:jingle:apps:file-transfer:3" for this version (see Namespace Versioning regarding the possibility of incrementing the version number).

Listing 15: Service discovery information request

```
<iq from='romeo@montague.lit/orchard'
    id='uw72g176'
    to='juliet@capulet.lit/balcony'
    type='get'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 16: Service discovery information response

```
<iq from='juliet@capulet.lit/balcony'
    id='uw72g176'
    to='romeo@montague.lit/orchard'
    type='result'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    <feature var='urn:xmpp:jingle:1'/>
    <feature var='urn:xmpp:jingle:apps:file-transfer:3'/>
    <feature var='urn:xmpp:jingle:transports:s5b:1'/>
    <feature var='urn:xmpp:jingle:transports:ibb:1'/>
  </query>
</iq>
```

As noted, if an application supports exchange of multiple files, it MUST advertise a service discovery feature of "urn:xmpp:jingle:apps:file-transfer:multi".
In order for an application to determine whether an entity supports this protocol, where possible it SHOULD use the dynamic, presence-based profile of service discovery defined in Entity Capabilities [12]. However, if an application has not received entity capabilities information from an entity, it SHOULD use explicit service discovery instead.

11 Security Considerations

For historical reasons and for backward-compatibility with XEP-0096, the hashing algorithm used in computing the file checksum defaults to MD5. It is RECOMMENDED for implementations to use stronger hashing algorithms.
In order to secure the data stream, implementations SHOULD use encryption methods appropriate to the transport method being used. For example, end-to-end encryption can be negotiated over either SOCKS5 Bytestreams or In-Band Bytestreams as described in XEP-0260 and XEP-0261.
Refer to XEP-0047, XEP-0065, XEP-0096, XEP-0260, and XEP-0261 for related security considerations.

12 IANA Considerations

No interaction with the Internet Assigned Numbers Authority (IANA) [13] is required as a result of this document.

[12] XEP-0115: Entity Capabilities <http://xmpp.org/extensions/xep-0115.html>.
[13] 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/>.

13 XMPP Registrar Considerations

13.1 Protocol Namespaces

This specification defines the following XML namespace:

- urn:xmpp:jingle:apps:file-transfer:3

Upon advancement of this specification from a status of Experimental to a status of Draft, the XMPP Registrar [14] shall add the foregoing namespace to the registry located at <http://xmpp.org/registrar/namespaces.html>, as described in Section 4 of XMPP Registrar Function [15].

13.2 Namespace Versioning

If the protocol defined in this specification undergoes a revision that is not fully backwards-compatible with an older version, the XMPP Registrar shall increment the protocol version number found at the end of the XML namespaces defined herein, as described in Section 4 of XEP-0053.

13.3 Service Discovery Features

The service discovery feature for advertising support for exchange of multiple files is ”urn:xmpp:jingle:apps:file-transfer:multi”.
The registry submission is as follows.

```
<var>
  <name>urn:xmpp:jingle:apps:file-transfer:multi</name>
  <desc>Signals support for exchange of multiple files.</desc>
  <doc>XEP-0234</doc>
</var>
```

13.4 Jingle Application Formats

The XMPP Registrar shall include ”file-transfer” in its registry of Jingle application formats. The registry submission is as follows:

[14] 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 <http://xmpp.org/registrar/>.
[15] XEP-0053: XMPP Registrar Function <http://xmpp.org/extensions/xep-0053.html>.

```
<application>
  <name>file-transfer</name>
  <desc>Jingle sessions for the transfer of a file</desc>
  <transport>streaming</transport>
  <doc>XEP-0234</doc>
</application>
```

14 XML Schema

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

<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='urn:xmpp:jingle:apps:file-transfer:3'
    xmlns='urn:xmpp:jingle:apps:file-transfer:3'
    elementFormDefault='qualified'>

  <xs:import namespace='urn:xmpp:hashes:1'/>

  <xs:element name='description'>
    <xs:complexType>
      <xs:choice>
        <xs:element name='offer' type='fileTransferElementType'/>
        <xs:element name='request' type='fileTransferElementType'/>
      </xs:choice>
    </xs:complexType>
  </xs:element>

  <xs:element name='abort' type='fileTransferElementType'/>

  <xs:element name='received' type='fileTransferElementType'/>

  <xs:element name='checksum' type='fileTransferElementType'/>

  <xs:complexType name='fileTransferElementType'>
    <xs:sequence>
      <xs:element name='file'/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name='file'>
    <xs:complexType>
      <xs:sequence xmlns:h='urn:xmpp:hashes:1'>
        <xs:element name='date' type='xs:date'/>
        <xs:element name='name' type='xs:string'/>
        <xs:element name='size' type='xs:positiveInteger'/>
        <xs:element ref='h:hash' minOccurs='0' maxOccurs='unbounded'/>
```

```
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

15 Acknowledgements

Thanks to Diana Cionoiu, Olivier Crête, Viktor Fast, Waqas Hussain, Justin Karneges, Steffen Larsen, Yann Leboulanger, Marcus Lundblad, Robert McQueen, Joe Maissel, Glenn Maynard, Ali Sabil, Sjoerd Simons, Will Thompson, Matthew Wild, and Jiří Zárevúcky for their feedback.