XEP-0096: SI File Transfer

Thomas Muldowney
mailto:temas@jabber.org
xmpp:temas@jabber.org

Matthew Miller
mailto:linuxwolf@outer-planes.net
xmpp:linuxwolf@outer-planes.net

Ryan Eatmon
mailto:reatmon@jabber.org
xmpp:reatmon@jabber.org

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

2004-04-13
Version 1.2

Status	Type	Short Name
Draft	Standards Track	si-filetransfer

This specification defines a profile of the XMPP stream initiation extension 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

The traditional mechanism for transferring files in the Jabber community is the Out-of-Band Data [1] protocol. That protocol has several drawbacks:

1. It is not reliable.
2. It does not work when one of the parties is behind a firewall.
3. It provides limited metadata about files to be exchanged.

The current document defines a profile of Stream Initiation [2] that solves the problems with out-of-band data, thus providing a robust, reliable mechanism for file transfers over the Jabber network. Implementors are referred to XEP-0095 regarding the underlying concepts of stream initiation.

2 Requirements

- Enable seamless file transfer, including fall-back mechanisms as appropriate.
- Ensure that the profile will work even when one or both parties are behind a firewall.
- Define a full-featured set of metadata for file transfers, including the following:
  - description
  - size
  - name
  - date
  - hash
- Optionally support ranged transfers.

3 Protocol

The file transfer profile is in the ”http://jabber.org/protocol/si/profile/file-transfer” namespace. The profile is fairly simple: it consists of the root element with the possibility of one child describing the optional ranged transfers.
The root element is <file> and has four attributes. The attributes are used only during the offer stage of stream initiation:

[1] XEP-0066: Out of Band Data <http://xmpp.org/extensions/xep-0066.html>.
[2] XEP-0095: Stream Initiation <http://xmpp.org/extensions/xep-0095.html>.

- size - The size, in bytes, of the data to be sent.
- name - The name of the file that the Sender wishes to send.
- date - The last modification time of the file. This is specified using the DateTime profile as described in XMPP Date and Time Profiles [3].
- hash - The MD5 sum of the file contents.

The size and name attributes MUST be present in the profile. The other attributes MAY be present.
There are two possible child elements of the root: <desc> and <range>. Both are OPTIONAL.
<desc> is used to provide a sender-generated description of the file so the receiver can better understand what is being sent. It MUST NOT be sent in the result.
When <range> is sent in the offer, it should have no attributes. This signifies that the sender can do ranged transfers. When a Stream Initiation result is sent with the <range> element, it uses these attributes:

- offset - Specifies the position, in bytes, to start transferring the file data from. This defaults to zero (0) if not specified.
- length - Specifies the number of bytes to retrieve starting at offset. This defaults to the length of the file from offset to the end.

Both attributes are OPTIONAL on the <range> element. Sending no attributes is synonymous with not sending the <range> element. When no <range> element is sent in the Stream Initiation result, the Sender MUST send the complete file starting at offset 0. More generally, data is sent over the stream byte for byte starting at the offset position for the length specified.

3.1 Mandatory-to-Implement Technologies

In order to enable seamless file transfer and appropriate fall-back mechanisms, implementations of this profile MUST support both SOCKS5 Bytestreams [4] and In-Band Bytestreams [5], to be preferred in that order. The associated namespaces are to be included as option values for the ”stream-method” variable as shown in the examples below.
Additionally, implementations MAY support other mechanisms.

4 Examples

[3] XEP-0082: XMPP Date and Time Profiles <http://xmpp.org/extensions/xep-0082.html>.
[4] XEP-0065: SOCKS5 Bytestreams <http://xmpp.org/extensions/xep-0065.html>.
[5] XEP-0047: In-Band Bytestreams <http://xmpp.org/extensions/xep-0047.html>.

Listing 1: Simple Profile Usage in Stream Initiation Offer

```
<iq type='set' id='offer1' to='receiver@jabber.org/resource'>
  <si xmlns='http://jabber.org/protocol/si'
      id='a0'
      mime-type='text/plain'
      profile='http://jabber.org/protocol/si/profile/file-transfer'>
    <file xmlns='http://jabber.org/protocol/si/profile/file-transfer'
          name='test.txt'
          size='1022'/>
    <feature xmlns='http://jabber.org/protocol/feature-neg'>
      <x xmlns='jabber:x:data' type='form'>
        <field var='stream-method' type='list-single'>
          <option><value>http://jabber.org/protocol/bytestreams</value
              ></option>
          <option><value>http://jabber.org/protocol/ibb</value></
              option>
        </field>
      </x>
    </feature>
  </si>
</iq>
```

Listing 2: Simple Profile Usage in Stream Initiation Result

```
<iq type='result' to='sender@jabber.org/resource' id='offer1'>
  <si xmlns='http://jabber.org/protocol/si'>
    <feature xmlns='http://jabber.org/protocol/feature-neg'>
      <x xmlns='jabber:x:data' type='submit'>
        <field var='stream-method'>
          <value>http://jabber.org/protocol/bytestreams</value>
        </field>
      </x>
    </feature>
  </si>
</iq>
```

Listing 3: Complete Profile Usage in Stream Initiation Offer

```
<iq type='set' id='offer1' to='receiver@jabber.org/resource'>
  <si xmlns='http://jabber.org/protocol/si'
      id='a0'
      mime-type='text/plain'
      profile='http://jabber.org/protocol/si/profile/file-transfer'>
    <file xmlns='http://jabber.org/protocol/si/profile/file-transfer'
          name='test.txt'
          size='1022'
          hash='552da749930852c69ae5d2141d3766b1'
          date='1969-07-21T02:56:15Z'>
      <desc>This is a test. If this were a real file...</desc>
```

```
    </file>
    <feature xmlns='http://jabber.org/protocol/feature-neg'>
      <x xmlns='jabber:x:data' type='form'>
        <field var='stream-method' type='list-single'>
          <option><value>http://jabber.org/protocol/bytestreams</value
              ></option>
          <option><value>http://jabber.org/protocol/ibb</value></
              option>
        </field>
      </x>
    </feature>
  </si>
</iq>
```

Listing 4: Complete Profile Usage in Stream Initiation Result

```
<iq type='result' to='sender@jabber.org/resource' id='offer1'>
  <si xmlns='http://jabber.org/protocol/si'>
    <file xmlns='http://jabber.org/protocol/si/profile/file-transfer'/
        >
    <feature xmlns='http://jabber.org/protocol/feature-neg'>
      <x xmlns='jabber:x:data' type='submit'>
        <field var='stream-method'>
          <value>http://jabber.org/protocol/bytestreams</value>
        </field>
      </x>
    </feature>
  </si>
</iq>
```

This range should retrieve 256 bytes from the beginning of the file:

```
<range length='256'/>
```

This range should retrieve 256 bytes starting from the 128th byte in the file:

```
<range offset='128' length='256'/>
```

This range should retrieve the remainder of the file starting at the 128th byte in the file:

```
<range offset='128'/>
```

This range is the same as having not sent the range request and the entire file is sent:

```
<range/>
```

5 IANA Considerations

No interaction with the Internet Assigned Numbers Authority (IANA) [6] is required as a result of this document.

6 XMPP Registrar Considerations

6.1 Stream Initiation Profiles

The profile described in this document is included in the stream initiation profiles registry maintained by the XMPP Registrar [7] (see <http://xmpp.org/registrar/si-profiles.html>). The registry submission is as follows:

```
<profile>
  <name>http://jabber.org/protocol/si/profile/file-transfer</name>
  <doc>XEP-0096</doc>
  <desc>A profile for file transfer between any two entities.</desc>
</profile>
```

6.2 URI Query Types

As authorized by XMPP URI Query Components [8], the XMPP Registrar maintains a registry of queries and key-value pairs for use in XMPP URIs (see <http://xmpp.org/registrar/querytypes.html>).
As described below, the registered querytypes for file transfer actions are ”sendfile” and ”recvfile”. Note well that ”sendfile” means a second entity will send a file to the XMPP entity that controls the IRI/URI and that ”recvfile” means a second entity will receive a file from the XMPP entity that controls the IRI/URI.

6.2.1 sendfile

To enable a second entity to send a file, the IRI/URI is of the following form:

Listing 5: Sending a File: IRI/URI

```
xmpp:romeo@montague.net/orchard?sendfile
```

[6] 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/>.

[7] 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/>.

[8] XEP-0147: XMPP URI Query Components <http://xmpp.org/extensions/xep-0147.html>.

The application SHOULD then present an interface enabling the user to provide information about the file to be sent (e.g., by "browsing" the file system of the user's computer in order to choose a file). As a result, the application SHOULD then send a Publishing Stream Initiation Requests [9] message to the XMPP address encapsulated in the IRI/URI:

Listing 6: Sending a File: Resulting Stanza

```
<message from='juliet@capulet.com/balcony' to='romeo@montague.net'>
  <sipub xmlns='http://jabber.org/protocol/si-pub'
      id='publish-0123'
      mime-type='text/plain'
      profile='http://jabber.org/protocol/si/profile/file-transfer'>
    <file xmlns='http://jabber.org/protocol/si/profile/file-transfer'
          name='missive.txt'
          size='1024'
          date='2005-11-29T11:21Z'/>
  </sipub>
</message>
```

The following submission registers the "sendfile" querytype.

```
<querytype>
  <name>sendfile</name>
  <proto>http://jabber.org/protocol/si/profile/file-transfer</proto>
  <desc>enables initiation of an inbound file transfer to XMPP entity<
     /desc>
  <doc>XEP-0096</doc>
</querytype>
```

6.2.2 recvfile

To enable a second entity to receive a file, the IRI/URI is of the following form:

Listing 7: Receiving a File: IRI/URI

```
xmpp:romeo@montague.net/orchard?recvfile;sid=pub234;mime-type=text%2
    Fplain;name=reply.txt;size=2048
```

That IRI/URI is equivalent to the following XML stanza:

Listing 8: Receiving a File: Equivalent Stanza

```
<message from='romeo@montague.net' to='juliet@capulet.com/balcony'>
  <sipub xmlns='http://jabber.org/protocol/si-pub'
      id='pub234'
```

[9] XEP-0137: Publishing Stream Initiation Requests <http://xmpp.org/extensions/xep-0137.html>.

```
      mime-type='text/plain'
      profile='http://jabber.org/protocol/si/profile/file-transfer'>
    <file xmlns='http://jabber.org/protocol/si/profile/file-transfer'
          name='reply.txt'
          size='2048'/>
  </sipub>
</message>
```

In accordance with XEP-0137, the application SHOULD then initiate a file transfer exchange with by sending a stanza of the following form:

Listing 9: Receiving a File: Resulting Stanza

```
<iq from='juliet@capulet.com/balcony' to='romeo@montague.net/orchard'>
  <start xmlns='http://jabber.org/protocol/si-pub' id='pub234'/>
</iq>
```

Note well that the request to begin the stream is sent to the full JID (user@host/resource) of the XMPP entity identified by the XMPP IRI/URI. Therefore, the IRI/URI SHOULD include a full JID. If it does not, the receiver MUST discover a full JID via presence or service discovery. If the receiver cannot discover a full JID for the sender (e.g., in the last resort through sending a presence subscription request to the sender and receiving presence from the sender's resources), then it SHOULD abort the file transfer exchange.
The following submission registers the "recvfile" querytype.

```
<querytype>
  <name>recvfile</name>
  <proto>http://jabber.org/protocol/si/profile/file-transfer</proto>
  <desc>enables initiation of an outbound file transfer from XMPP
     entity</desc>
  <doc>XEP-0096</doc>
  <keys>
    <key>
      <name>algo</name>
      <desc>the hash algorithm used to generate the checksum</desc>
    </key>
    <key>
      <name>hash</name>
      <desc>a checksum of the file contents</desc>
    </key>
    <key>
      <name>mime-type</name>
      <desc>the MIME type of the file being offered</desc>
    </key>
    <key>
      <name>name</name>
      <desc>the name of the file being offered</desc>
    </key>
```

```
    <key>
      <name>sid</name>
      <desc>the session ID associated with the file being offered</
         desc>
    </key>
    <key>
      <name>size</name>
      <desc>the size in bytes of the file being offered</desc>
    </key>
  </keys>
</querytype>
```

7 XML Schema

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

<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='http://jabber.org/protocol/si/profile/file-
       transfer'
    xmlns='http://jabber.org/protocol/si/profile/file-transfer'
    elementFormDefault='qualified'>

  <xs:annotation>
    <xs:documentation>
      The protocol documented by this schema is defined in
      XEP-0096: http://www.xmpp.org/extensions/xep-0096.html
    </xs:documentation>
  </xs:annotation>

  <xs:element name='file'>
    <xs:complexType>
      <xs:sequence minOccurs='0'>
        <xs:element name='desc' type='xs:string'/>
        <xs:element ref='range'/>
      </xs:sequence>
      <xs:attribute name='date' type='xs:string' use='optional'/>
      <xs:attribute name='hash' type='xs:string' use='optional'/>
      <xs:attribute name='name' type='xs:string' use='required'/>
      <xs:attribute name='size' type='xs:integer' use='required'/>
    </xs:complexType>
  </xs:element>

  <xs:element name='range'>
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base='empty'>
```

```
          <xs:attribute name='length' type='xs:integer' use='optional'
              />
          <xs:attribute name='offset' type='xs:integer' use='optional'
              />
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>

  <xs:simpleType name='empty'>
    <xs:restriction base='xs:string'>
      <xs:enumeration value=''/>
    </xs:restriction>
  </xs:simpleType>

</xs:schema>
```