XEP-0262: Use of ZRTP in Jingle RTP Sessions

Peter Saint-Andre
mailto:xsf@stpeter.im
xmpp:peter@jabber.org
http://stpeter.im/

2011-06-15
Version 1.0

Status	Type	Short Name
Draft	Standards Track	jingle-zrtp

This specification defines how to use ZRTP (RFC 6189) in the Jingle application type for the Real-time Transport Protocol (RTP) as a way to negotiate media path key agreement for secure RTP in one-to-one media sessions.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Protocol

Jingle RTP Sessions (XEP-0167) [1] recommends the use of the Secure Real-time Transport Protocol (SRTP) for end-to-end encryption of RTP sessions negotiated using Jingle (XEP-0166) [2]. An alternative approach to end-to-end encryption of RTP traffic is provided by RFC 6189 [3], developed by Phil Zimmermann, the inventor of "Pretty Good Privacy" (PGP). Although negotiation of ZRTP mainly occurs in the media channel rather than the signalling channel, the ZRTP specification defines one SDP attribute called "zrtp-hash" (this communicates the ZRTP version supported as well as a hash of the Hello message). Inclusion of this information is OPTIONAL in both SIP/SDP and Jingle.
The SDP format is shown below.

```
a=zrtp-hash:zrtp-version zrtp-hash-value
```

An example follows.

```
a=zrtp-hash:1.10
    fe30efd02423cb054e50efd0248742ac7a52c8f91bc2df881ae642c371ba46df
```

This SDP attribute can be translated into Jingle as a <zrtp-hash/> element qualified by the 'urn:xmpp:jingle:apps:rtp:zrtp:1' namespace, as shown below.

```
<zrtp-hash version='zrtp-version' xmlns='
    urn:xmpp:jingle:apps:rtp:zrtp:1'>zrtp-hash-value</zrtp-hash>
```

An example follows.

```
<zrtp-hash version='1.10' xmlns='urn:xmpp:jingle:apps:rtp:zrtp:1'>
  fe30efd02423cb054e50efd0248742ac7a52c8f91bc2df881ae642c371ba46df
</zrtp-hash>
```

The <zrtp-hash/> element is sent as a child of the <encryption/> element defined in Jingle RTP Sessions (XEP-0167) [4].
If the Jingle initiator wishes to use ZRTP, it includes the <zrtp-hash/> element in its session invitation (where it hashes over its own Hello message as described in the ZRTP specification).

Listing 1: Initiator sends session invitation with zrtp-hash

```
<iq from='romeo@montague.lit/orchard'
    id='uz61v4m4'
    to='juliet@capulet.lit/balcony'
```

[1] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.
[2] XEP-0166: Jingle <https://xmpp.org/extensions/xep-0166.html>.
[3] RFC 6189: ZRTP: Media Path Key Agreement for Unicast Secure RTP <http://tools.ietf.org/html/rfc6189>.
[4] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.

```
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@montague.lit/orchard'
          sid='a73sjjvkla37jfea'>
    <content creator='initiator' name='voice'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='audio'>
        <payload-type id='96' name='speex' clockrate='16000'/>
        <payload-type id='97' name='speex' clockrate='8000'/>
        <payload-type id='18' name='G729'/>
        <payload-type id='103' name='L16' clockrate='16000' channels='
            2'/>
        <payload-type id='98' name='x-ISAC' clockrate='8000'/>
        <encryption required='true'>
          <zrtp-hash xmlns='urn:xmpp:jingle:apps:rtp:zrtp:1' version='
              1.10'>
            fe30efd02423cb054e50efd0248742ac7a52c8f91bc2df881ae642c371ba46df

          </zrtp-hash>
        </encryption>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'
                 pwd='asd88fgpdd777uzjYhagZg'
                 ufrag='8hhy'>
        <candidate component='1'
                   foundation='1'
                   generation='0'
                   id='el0747fg11'
                   ip='10.0.1.1'
                   network='1'
                   port='8998'
                   priority='2130706431'
                   protocol='udp'
                   type='host'/>
        <candidate component='1'
                   foundation='2'
                   generation='0'
                   id='y3s2b30v3r'
                   ip='192.0.2.3'
                   network='1'
                   port='45664'
                   priority='1694498815'
                   protocol='udp'
                   rel-addr='10.0.1.1'
                   rel-port='8998'
                   type='srflx'/>
      </transport>
    </content>
  </jingle>
```

```
</iq>
```

If the receiving party wishes to proceed with ZRTP negotiation, it also includes the <zrtp-hash/> element in its session-accept message (where it hashes over its own Hello message as described in the ZRTP specification).

Listing 2: Responder sends session-accept

```
<iq from='juliet@capulet.lit/balcony'
    id='pn2va48j'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-accept'
          initiator='romeo@montague.lit/orchard'
          responder='juliet@capulet.lit/balcony'
          sid='a73sjjvkla37jfea'>
    <content creator='initiator' name='voice'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='audio'>
        <payload-type id='97' name='speex' clockrate='8000'/>
        <payload-type id='18' name='G729'/>
        <encryption>
          <zrtp-hash xmlns='urn:xmpp:jingle:apps:rtp:zrtp:1' version='
              1.10'>
            badfbe66ff87fe135750377509b09b0babd1c3ec25fa4314565e2bf7ccc30299

          </zrtp-hash>
        </encryption>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'
                 pwd='YH75Fviy6338Vbrhrlp8Yh'
                 ufrag='9uB6'>
        <candidate component='1'
                   foundation='1'
                   generation='0'
                   id='or2ii2syr1'
                   ip='192.0.2.1'
                   network='0'
                   port='3478'
                   priority='2130706431'
                   protocol='udp'
                   type='host'/>
      </transport>
    </content>
  </jingle>
</iq>
```

Note that a unique zrtp-hash is needed for each media stream, since the hash for each stream is computed from a different ZRTP Hello message (e.g., if a session includes both audio and

video then the value of the <zrtp-hash/> element included in the <description/> element for the audio stream will be different from the value for the video stream).

2 Determining Support

If an entity supports the use of ZRTP in Jingle as described in this document, it MUST advertise that fact in its responses to Service Discovery (XEP-0030) [5] information ("disco#info") requests by returning a feature of "urn:xmpp:jingle:apps:rtp:zrtp:1":

Listing 3: A disco#info query

```
<iq type='get'
    from='calvin@usrobots.lit/lab'
    to='herbie@usrobots.lit/home'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 4: A disco#info response

```
<iq type='result'
    from='herbie@usrobots.lit/home'
    to='calvin@usrobots.lit/lab'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    <feature var='urn:xmpp:jingle:1'/>
    <feature var='urn:xmpp:jingle:apps:rtp:zrtp:1'/>
  </query>
</iq>
```

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 (XEP-0115) [6]. However, if an application has not received entity capabilities information from an entity, it SHOULD use explicit service discovery instead.

3 Security Considerations

Security considerations for ZRTP itself are provided in RFC 6189.
XMPP stanzas such as Jingle invite messages and service discovery exchanges are not encrypted or signed. As a result, it is possible for an attacker to intercept these stanzas and modify them, thus convincing one party that the other party does not support ZRTP and

[5] XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.
[6] XEP-0115: Entity Capabilities <https://xmpp.org/extensions/xep-0115.html>.

therefore denying the parties an opportunity to use ZRTP. However, because the zrtp-hash is mostly advisory, the parties could still use ZRTP even if the signalling channel is compromised.

4 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [7].

5 Acknowledgements

Thanks to Werner Dittmann and Emil Ivov for their implementation feedback.

6 XMPP Registrar Considerations

6.1 Protocol Namespaces

This specification defines the following XML namespace:

- urn:xmpp:jingle:apps:rtp:zrtp:1

The XMPP Registrar [8] includes the foregoing namespace to the registry located at <https://xmpp.org/registrar/namespaces.html>, as described in Section 4 of XMPP Registrar Function (XEP-0053) [9].

6.2 Protocol 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.

[7] 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/>.

[8] 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/>.

[9] XEP-0053: XMPP Registrar Function <https://xmpp.org/extensions/xep-0053.html>.

7 XML Schemas

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

<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='urn:xmpp:jingle:apps:rtp:zrtp:1'
    xmlns='urn:xmpp:jingle:apps:rtp:zrtp:1'
    elementFormDefault='qualified'>

  <xs:annotation>
    <xs:documentation>
      The protocol documented by this schema is defined in
      XEP-0262: http://www.xmpp.org/extensions/xep-0262.html
    </xs:documentation>
  </xs:annotation>

  <xs:element name='zrtp-hash'>
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base='xs:string'>
          <xs:attribute name='version' type='xs:string' use='required'
              />
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>

</xs:schema>
```