



XMPP

XEP-0111: A Transport for Initiating and Negotiating Sessions (TINS)

Joe Hildebrand
<mailto:jhildebr@cisco.com>
<xmpp:hildjj@jabber.org>

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

2005-12-21
Version 0.8

| Status | Type | Short Name |
|-----------|-----------------|------------|
| Retracted | Standards Track | tins |

This document defined a SIP-compatible transport for initiating and negotiating sessions using SDP over XMPP.

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 | Introduction | 1 |
| 2 | Requirements | 1 |
| 3 | Protocol | 1 |
| 4 | Discovering Support | 3 |
| 5 | Examples | 3 |
| 5.1 | Negotiating a Voice Call | 3 |
| 6 | Security Considerations | 7 |
| 7 | IANA Considerations | 7 |
| 8 | XMPP Registrar Considerations | 7 |
| 8.1 | Protocol Namespaces | 7 |
| 9 | XML Schemas | 7 |
| 9.1 | tins | 7 |
| 9.2 | sdp | 8 |

1 Introduction

Note Well: This proposal has been retracted by the authors in favor of [Jingle \(XEP-0166\)](#)¹.

The Session Description Protocol (SDP; see [RFC 2327](#)²) provides a mechanism for describing multimedia sessions that are advertised and negotiated over the Internet. The "Transport for Initiating and Negotiating Sessions" (TINS) specified herein describes how to use SDP to build a framework for media stream/session initiation and negotiation between entities that natively support XMPP (see [XMPP Core](#)³).⁴ In particular, TINS provides an XMPP representation of standard session management semantics such as those provided by the Session Initiation Protocol (SIP; see [RFC 3261](#)⁶). As a result, native XMPP clients that support TINS can negotiate out-of-band multimedia sessions (e.g., use of the Real-Time Transport Protocol or RTP; see [RFC 3550](#)⁷) and XMPP services that support TINS can easily interoperate with SIP services through gateways.

2 Requirements

This document addresses the following requirements:

1. Enable an XMPP entity to negotiate an out-of-band multimedia session with another XMPP entity.
2. Enable an XMPP entity to negotiate an out-of-band multimedia session with a non-XMPP entity through a gateway.
3. Maximize interoperability with existing gateways and devices by using standard Internet protocols.

3 Protocol

TINS exchanges are completed by sending `<message/>` stanzas containing a child `<tins/>` element qualified by the 'http://jabber.org/protocol/tins' namespace.⁸ In order to track the

¹XEP-0166: Jingle <https://xmpp.org/extensions/xep-0166.html>.

²RFC 2327: SDP: Session Description Protocol <http://tools.ietf.org/html/rfc2327>.

³RFC 6120: Extensible Messaging and Presence Protocol (XMPP); Core <http://tools.ietf.org/html/rfc6120>.

⁴The approach taken herein is to send pure SDP. While earlier versions of this document used [Session Description and Capability Negotiation \(SDPng\)](#)⁵ (an XML representation of SDP), SDPng is a more experimental technology; by contrast, SDP is a stable protocol and there is broad support for it by existing gateways and devices. The use of SDP rather than SDPng thus enables the Jabber/XMPP community to implement solutions that are deployable on the Internet today.

⁶RFC 3261: Session Initiation Protocol (SIP) <http://tools.ietf.org/html/rfc3261>.

⁷RFC 3550: RTP: A Transport Protocol for Real-Time Applications <http://tools.ietf.org/html/rfc3550>.

⁸While it may seem that the semantics of `<iq/>` stanzas are more appropriate, [RFC 3261](#)⁹ allows entities to send multiple results in response to a SIP request, which does not map to the syntax of the `<iq/>` stanza as defined in RFC 6120.

structure of the TINS "conversation", the <thread/> child of <message/> MAY also be included. The <tins/> element MUST possess a 'method' attribute, whose value SHOULD be either an IANA-registered value for a SIP method or "result", as described below. The following SIP methods will probably be used most frequently in TINS interactions:

- INVITE -- Used to invite the target user to an out-of-band session. The content inside the <tins/> element MAY be SDP descriptions of the connection types offered. If a session is already established for this transaction, the new INVITE serves as a renegotiation of session parameters.
- ACK -- Used by the initiator to tell the invitee that an out-of-band session has been established.
- BYE -- Used by either side of the conversation to terminate the transaction. This message SHOULD cause all resources associated with this transaction to be freed, and any associated network connections to be terminated.

The SDP data itself is included as the XML character data of an <sdp/> child of the <tins/> element, qualified by the 'urn:ietf:rfc:2327' namespace (this is consistent with RFC 2648 ¹⁰).

¹¹ Any restricted XML characters in the SDP data (i.e., & ' < > ") MUST be properly escaped when contained in the XML character data of the <sdp/> element (for example, the ' character MUST be escaped to '). It is the responsibility of the XMPP recipient or translating gateway to unescape these restricted characters for processing.

The request stanza MAY also include either or both of the following:

- Header information or Internet metadata (such as that defined by RFC 3261 ¹²) in the format specified by [Stanza Headers and Internet Metadata \(XEP-0131\)](#) ¹³.
- Multicast addresses as specified by [Extended Stanza Addressing \(XEP-0033\)](#) ¹⁴.

In reply to a request, the receiver MUST send zero or more replies, with the value of the 'method' attribute set to a value of "result" and the value of the 'code' attribute set to one of the valid SIP response codes as specified in Section 21 of RFC 3261 ¹⁵.

¹⁰RFC 2648: A URN Namespace for IETF Documents <<http://tools.ietf.org/html/rfc2648>>.

¹¹The <sdp/> element is qualified by a separate namespace because it may be desirable for TINS to support other formats (such as SDPng) in the future; these can then be added without changing the XML schema for TINS.

¹²RFC 3261: Session Initiation Protocol (SIP) <<http://tools.ietf.org/html/rfc3261>>.

¹³XEP-0131: Stanza Headers and Internet Metadata <<https://xmpp.org/extensions/xep-0131.html>>.

¹⁴XEP-0033: Extended Stanza Addressing <<https://xmpp.org/extensions/xep-0033.html>>.

¹⁵RFC 3261: Session Initiation Protocol (SIP) <<http://tools.ietf.org/html/rfc3261>>.

4 Discovering Support

Before initiating a TINS negotiation, an XMPP entity SHOULD determine that the target entity supports the 'http://jabber.org/protocol/tins' namespace. Such discovery SHOULD occur by means of [Service Discovery \(XEP-0030\)](#)¹⁶, either directly by querying the target entity or indirectly by means of [Entity Capabilities \(XEP-0115\)](#)¹⁷. If the target entity is a non-XMPP entity that is contacted through a gateway, the gateway itself SHOULD reply to service discovery queries on behalf of the non-XMPP entity and SHOULD insert a client capabilities extension into the presence stanzas it generates on behalf of the non-XMPP entity.

If an XMPP entity receives, or a gateway handles, a <message/> stanza containing a <tins/> element qualified by the 'http://jabber.org/protocol/tins' namespace but it does not understand the TINS protocol, it SHOULD either silently ignore it or return a <service-unavailable/> error (see [Error Condition Mappings \(XEP-0086\)](#)¹⁸ for error syntax).

5 Examples

5.1 Negotiating a Voice Call

The following XMPP stanzas could be used to initiate a voice call. The 'from' addresses will usually be added by the XMPP server or relevant gateway, but are shown here for the sake of clarity. Note the inclusion of SHIM headers and extended addresses.

Listing 1: Step 1: A sends an invite to B

```
<message
  from='A@example.com/work'
  to='B@example.com/laptop'
  id='tins01'>
  <thread>1234@hostA.example.com</thread>
  <tins method='INVITE' xmlns='http://jabber.org/protocol/tins'>
    <sdp xmlns='urn:ietf:rfc:2327'>
      v=0
      o=A@example.com 98765432 IN IP4 192.168.1.1
      s=TINS questions
      i=Let's talk about TINS
      e=A@example.com
      p=+1-303-555-1212
      c=IN IP4 192.168.1.1/127
      t=3288361865 0
      a=recvonly
      m=audio 7800 RTP/AVP 0
    </sdp>
  </tins>
</message>
```

¹⁶XEP-0030: Service Discovery <<https://xmpp.org/extensions/xep-0030.html>>.

¹⁷XEP-0115: Entity Capabilities <<https://xmpp.org/extensions/xep-0115.html>>.

¹⁸XEP-0086: Error Condition Mappings <<https://xmpp.org/extensions/xep-0086.html>>.

```

</tins>
<headers xmlns='http://jabber.org/protocol/shim'>
  <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
    z9hG4bK776asdhds</header>
  <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
  <header name='CSeq'>314159 INVITE</header>
</headers>
<addresses xmlns='http://jabber.org/protocol/address'>
  <address type='bcc' jid='compliance.example.com'/>
</addresses>
</message>

```

Listing 2: Step 2: B tells A that it is trying

```

<message
  from='B@example.com/laptop'
  to='A@example.com/work'
  id='tins01'>
  <thread>1234@hostA.example.com</thread>
  <tins method='result'
    code='100'
    xmlns='http://jabber.org/protocol/tins'/>
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhds</header>
    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com'/>
  </addresses>
</message>

```

Listing 3: Step 3: B tells A that it is ringing

```

<message
  from='B@example.com/laptop'
  to='A@example.com/work'
  id='tins01'>
  <tins method='result'
    code='180'
    xmlns='http://jabber.org/protocol/tins'/>
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhds</header>
    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com'/>
  </addresses>
</message>

```

```

</addresses>
</message>

```

Listing 4: Step 4: B sends an updated description to A

```

<message
  from='B@example.com/laptop'
  to='A@example.com/work'
  id='tins02'>
  <thread>1234@hostA.example.com</thread>
  <tins method='result'
    code='200'
    xmlns='http://jabber.org/protocol/tins'>
  <sdp xmlns='urn:ietf:rfc:2327'>
    v=0
    o=A@example.com 98765432 IN IP4 192.168.1.2
    s=TINS questions
    i=Let's talk about TINS
    e=A@example.com
    p=+1-303-555-1212
    c=IN IP4 192.168.1.2/127
    t=3288361865 0
    a=recvonly
    m=audio 7800 RTP/AVP 0
    a=recvonly
    m=audio 9410 RTP/AVP 0
  </sdp>
</tins>
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhd</header>
    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com'/>
  </addresses>
</message>

```

Listing 5: Step 5: A sends an acknowledgement to B

```

<message
  from='A@example.com/work'
  to='B@example.com/laptop'
  id='tins02'>
  <thread>1234@hostA.example.com</thread>
  <tins method='ACK' xmlns='http://jabber.org/protocol/tins'/>
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhd</header>
  </headers>
</message>

```



```

    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com' />
  </addresses>
</message>

```

Listing 6: Step 6: B hangs up

```

<message
  from='B@example.com/laptop'
  to='A@example.com/work'
  id='tins03'>
  <thread>1234@hostA.example.com</thread>
  <tins method='BYE' xmlns='http://jabber.org/protocol/tins' />
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhd</header>
    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com' />
  </addresses>
</message>

```

Listing 7: Step 7: A acknowledges the hang up

```

<message
  from='A@example.com/work'
  to='B@example.com/laptop'
  id='tins03'>
  <thread>1234@hostA.example.com</thread>
  <tins method='result'
    code='200'
    xmlns='http://jabber.org/protocol/tins' />
  <headers xmlns='http://jabber.org/protocol/shim'>
    <header name='Via'>SIP/2.0/UDP tins.example.com;branch=
      z9hG4bK776asdhd</header>
    <header name='Call-ID'>a84b4c76e66710@tins.example.com</header>
    <header name='CSeq'>314159 INVITE</header>
  </headers>
  <addresses xmlns='http://jabber.org/protocol/address'>
    <address type='bcc' jid='compliance.example.com' />
  </addresses>
</message>

```

More examples to follow.

6 Security Considerations

TINS is subject to the same security considerations as XMPP, particularly with regard to authentication and channel encryption; for details, refer to [RFC 6120](#)¹⁹.

This document does not describe how the media protocols (e.g. RTP) traverse firewalls and NATs.

There is no general-purpose way to ensure that media protocol connections are associated with the in-band TINS conversation.

7 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#)²⁰.

8 XMPP Registrar Considerations

8.1 Protocol Namespaces

The [XMPP Registrar](#)²¹ shall include 'http://jabber.org/protocol/tins' in its registry of protocol namespaces.

9 XML Schemas

9.1 tins

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

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

  <xs:import namespace='urn:ietf:rfc:2327' />
```

¹⁹RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <<http://tools.ietf.org/html/rfc6120>>.

²⁰The 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/>>.

²¹The 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/>>.

```
<xs:element name='tins'>
  <xs:complexType>
    <xs:choice xmlns:sdp='urn:ietf:rfc:2327'>
      <xs:element ref='sdp:sdp' />
      <xs:any namespace='##other' />
    </xs:choice>
    <xs:attribute name='code' type='xs:string' use='optional' />
    <xs:attribute name='method' type='xs:string' use='required' />
  </xs:complexType>
</xs:element>
</xs:schema>
```

9.2 sdp

```
<?xml version='1.0' encoding='UTF-8'?>
<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='urn:ietf:rfc:2327'
  xmlns='urn:ietf:rfc:2327'
  elementFormDefault='qualified'>
  <xs:element name='sdp' type='xs:string' />
</xs:schema>
```