The P-Answer-State Header Extension to the Session Initiation Protocol (SIP) for the Open Mobile Alliance (OMA) Push to talk over Cellular (PoC)
draft-allen-sipping-poc-p-answer-state-header-02

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on September 6, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document describes a private Session Initiation Protocol (SIP) header (P-header) used by the Open Mobile Alliance (OMA), for Push to talk over Cellular (PoC) along with its applicability, which is
limited to the OMA PoC application. The P-Answer-State header is used for indicating the answering mode of the handset which is particular to the PoC application.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [1].

Table of Contents

1. Overall applicability ........................................... 4
2. Introduction ..................................................... 4
3. Terminology ..................................................... 4
4. Background for the extension ................................. 5
5. Overview .......................................................... 6
6. The P-Answer-State header ...................................... 7
   6.1. Requirements ............................................... 8
   6.2. Alternatives considered .................................... 8
   6.3. Applicability statement for the P-Answer-State header ... 9
   6.4. Usage of the P-Answer-State header ....................... 10
      6.4.1. Procedures at the UA (terminal) ....................... 11
      6.4.2. Procedures at the UA (PTT Server) .................... 11
      6.4.3. Procedures at the proxy server ....................... 13
7. Formal syntax .................................................... 13
   7.1. P-Answer-State header syntax .............................. 14
   7.2. Table of the new header .................................... 14
8. Example usage session flows ..................................... 14
   8.1. Pre-arranged group call using on-demand session ........ 14
   8.2. 1-1 Call using pre-established session .................. 19
9. Security considerations ......................................... 25
10. IANA considerations ............................................ 26
    10.1. Registration of Header Fields ........................... 26
    10.2. Registration of header field parameters ................ 27
11. Acknowledgements .............................................. 27
1. Overall applicability

The SIP extension specified in this document makes certain assumptions regarding network topology, and the availability of transitive trust. These assumptions are generally NOT APPLICABLE in the Internet as a whole. The mechanism specified here was designed to satisfy the requirements specified by the Open Mobile Alliance for Push-to-talk over cellular for which either no general-purpose solution was found, where insufficient operational experience was available to understand if a general solution is needed, or where a more general solution is not yet mature. For more details about the assumptions made about this extension, consult the Applicability subsection for the extension.

2. Introduction

The Open Mobile Alliance (OMA) (http://www.openmobilealliance.org) is specifying the Push-to-talk Over Cellular (PoC) service where SIP is the protocol used to establish half duplex media sessions across different participants. This document describes a private extension to address specific requirements of the PoC service and may not be applicable to the general Internet.

The PoC service allows a SIP UA (PoC terminal) to establish a session to one or more SIP UAs simultaneously, usually initiated by the initiating user pushing a button.

OMA has defined a collection of very stringent requirements in support of the PoC service. In order to provide the user with a satisfactory experience the initial session establishment from the time the user presses the button to the time they get an indication to speak must be minimized.

3. Terminology

The terms "PTT Server", "Unconfirmed Indication", "Unconfirmed Response", "Confirmed Indication" and "Confirmed Response" are introduced in this document.

A "PTT Server" as referred to here is a SIP network server that performs the network based functions for the Push to Talk service. The PTT Server may act as a SIP Proxy or back-to-back UA (B2BUA) based on the functions it needs to perform. There may be one or more PTT Servers involved in a SIP Push to Talk session.

An "Unconfirmed Indication" as referred to here is an indication that
the final target UA for the request has yet to be contacted and an intermediate SIP node is indicating that it has information that hints that the request is likely to be answered by the target UA.

An "Unconfirmed Response" is a 18x or 2xx response containing an "Unconfirmed Indication".

A "Confirmed Indication" as referred to here is an indication that the target UA has accepted the session invitation and is ready to receive media.

A "Confirmed Response" is a SIP 200 OK response containing a "Confirmed Indication" and has the usual semantics of a SIP 200 OK response containing an answer (such as an SDP answer).

4. Background for the extension

The PoC terminal may support such hardware capabilities as a speaker phone and/or headset and software that provide the capability for the user to configure the PoC terminal to accept the session invitations immediately and play out the media as soon as it is received without requiring the intervention of the called user. This mode of operation is known as Automatic Answer mode. The user may alternatively configure the PoC terminal to first alert the user and require the user to manually accept the session invitation before media is accepted. This mode of operation is known as Manual Answer mode. The PoC terminal may support both or only one of these modes of operation. The user may change the Answer Mode (AM) configuration of the PoC terminal frequently based on their current circumstances and preference, (perhaps because the user is busy, or in a public area where she cannot use a speaker phone, etc).

The OMA PoC Architecture utilizes PTT Servers within the network that may perform such roles as a conference focus [8], a RTP translator or a network policy enforcement server. A possible optimization to minimize the delay in the providing of the caller with an indication to speak is for the PTT server to perform buffering of media packets in order to provide an early or "Unconfirmed Indication" back to the caller and allow the caller to start speaking before the called PoC terminal has answered. An event package and mechanisms for a SIP UA to indicate its current answer mode to a PTT Server in order to enable buffering are defined in [9]. In addition, particularly when multiple domains are involved in the session, more than one PTT server may be involved in the signaling path for the session. Also the PTT Server that performs the buffering may not be the PTT Server that has knowledge of the current answer mode of the SIP UA that is the final destination for the SIP INVITE request. A mechanism to
allow a terminal that acts as a SIP UA or a PTT server that acts as a SIP UA to indicate a preference to the final destination SIP UAS to answer in a particular mode is defined in [10]. However a mechanism is required for a PTT Server to relay the "Unconfirmed Indication" in a response back towards the originating SIP UAC.

5. Overview

The purpose of this extension is to support an optimization that makes it possible for the network to provide a faster push-to-talk experience, through an intermediate SIP agent (PTT Server) providing a 200 OK response before the called UA does, and a PTT Server buffering the media generated by the calling UA for replay to the called UA when it answers. Because of the half duplex nature of the call where media bursts are short in the order of 10-30 seconds the additional end to end latency can be tolerated and this considerably improves the user experience. However the PTT Server only can do this when there is a high probability that the called UA is in Automatic Answer mode. It is likely that PTT Servers near the called UA have up-to-date knowledge of the answering mode of the called UA, and due to the restricted bandwidth nature of the cellular network, they can pass upstream an indication of the called UA’s answering mode faster than the called UA can deliver an automatically generated 200 OK response.

Thus, when a PTT Server forwards an INVITE and knows that the called UA is likely to be in Automatic Answer mode, it also generates a 183 response with a P-Answer-State header field with a parameter of "Unconfirmed" to signal to upstream PTT Servers that they may buffer the caller’s media.

A PTT Server that wishes to buffer the caller’s media, upon seeing the provisional response with a P-Answer-State header field with a parameter of "Unconfirmed" absorbs it and generates a 200 OK response for the caller’s UA with an appropriate answer.

When the called UA generates a 200 OK response, the PTT Server that generated the provisional response with a P-Answer-State header field with a parameter "Unconfirmed" adds to the 200 OK response a P-Answer-State header field with a parameter of "Confirmed". The 200 OK response is absorbed by the PTT Server that is buffering the caller’s media, as it has already generated a 200 OK response. The buffering PTT Server then starts un-buffering the media.

This document proposes a new SIP header field to support this Unconfirmed Indication". The new SIP header field may be optionally included in a response to a SIP INVITE request or in a sipfrag of a
response included in a NOTIFY request sent as a result of a REFER request that requests an INVITE to be sent. The header field is used to provide an indication from a PTT Server acting as a SIP proxy or back-to-back UA that it has information that hints that the terminating UA will likely answer automatically. This provides an "Unconfirmed Indication" back towards the inviting SIP UA to transmit media prior to receiving a final response from the final destination of the SIP INVITE request. No supported or require headers are needed because the sender of the P-Answer-State header field does not depend on the receiver to understand the extension and if the extension is not understood the header field is simply ignored by the recipient. The extension is described below.

6. The P-Answer-State header

The purpose of the P-Answer-State header field is to provide an indication from a PTT Server acting as a SIP proxy or back-to-back UA that is has information that hints that the terminating UA identified in the Request-URI of the request will likely answer automatically. Thus enabling the PTT Server to provide an "Unconfirmed Indication" back towards the inviting SIP UA to transmit media prior to receiving a final response from the final destination of the SIP INVITE request. If a Provisional response contains the P-Answer-State header field with the value "Unconfirmed" and does not contain an answer then a receiving PTT Server may send a 200 OK response containing an answer and a P-Answer-State header field with the value "Unconfirmed" if the PTT Server is willing to perform media buffering. If the response containing the P-Answer-State header field with the value "Unconfirmed" also contains an answer the PTT Server that included the P-Answer-State header field and answer in the response is also indicating that it is willing to buffer the media until a final "Confirmed Indication" is received.

The P-Answer-State header field MAY be included in a provisional or final response to a SIP INVITE request or in the sipfrag of a NOTIFY request sent as a result of a REFER request to send an INVITE request. If the P-Answer-State header field with value "Unconfirmed" is included in a provisional response that contains an answer the PTT Server is leaving the decision where to do buffering to other PTT Servers upstream and will forward upstream a "Confirmed indication" in a 200 OK response when the final response is received from the destination UA.

The P-Answer-State header is only included in a provisional response when the node that sends the response has knowledge that there is a PTT Server that acts as a B2BUA that understands this extension in the signaling path between itself and the originating UAC that will
only pass the header field on in either a 200 OK response or in the sipfrag as defined in [2] of a NOTIFY request as defined in [3] sent as a result of a REFER request as defined in [4]. Such a situation only occurs with specific network topologies which is another reason why use of this header field is not relevant to the general internet. The originating UAC will only receive the P-Answer-state header field in a 200 OK response or in the sipfrag of a NOTIFY request.

6.1. Requirements

The OMA PoC service has initial setup performance requirements that can be met by a PTT Server acting as a B2BUA spooling media from the inviting PoC subscriber until one or more invited PoC subscribers have accepted the session. The specific requirements are

REQ-1: An intermediate server MAY spool media from the inviting SIP UA until one or more invited PoC SIP UAs has accepted the invitation.

REQ-2: An intermediate server that is capable of spooling media MAY accept an invite request from an inviting SIP UAC even if no invited SIP UAS has accepted the invite request if it has a hint that the invited SIP UAC is likely to accept the request without requiring user intervention.

REQ-3: An intermediate server or proxy that is incapable of spooling media or does not wish to, but has a hint that the invited SIP UAC is likely to automatically accept the session invitation MUST be able to indicate back to another intermediate server that can spool media that it has some hint that the invited UAC is likely to automatically accept the session invitation.

REQ-4: An intermediate server that is willing to spool media from the inviting SIP UA until one or more invited SIP UAs have accepted the invite SHOULD indicate that it is spooling media to the inviting SIP UAC.

6.2. Alternatives considered

In order to meet REQ-3, a PTT Server needs to receive an indication back that the invited SIP UA is likely to accept the invite request without requiring user intervention. In this case, the PTT Server that has a hint that the invited SIP UAC is likely to accept the request can include an answer state indication in the 183 Session Progress or 200 OK response.

A number of alternatives were considered for the PTT Server to inform
another PTT Server or the inviting SIP UAC of the invited PoC SIP UAs
answer mode settings.

One proposal was to create a unique reason-phrase in the 183 and 200
OK response. This was rejected because the reason phrases are
normally intended for human readers and not meant to be parsed by
servers for special syntactic and semantic meaning.

Another proposal was to use a Reason header [11] in the 183 and 200
OK response. This was rejected because this would be inconsistent
with the intended use of the reason header and its usage is not
defined for these response codes and would have required creating and
registering a new protocol identifier.

Another proposal was to use a feature-tag in the returned Contact
header as defined in [12]. This was rejected because it was not a
different feature, but is an attribute of the session and can be
applied to many different features.

Another proposal was to use a new SDP attribute. The choice of an
SDP parameter was rejected because the answer state applies to the
session and not to a media stream.

The P-Answer-State header was chosen to give additional information
about the state of the SIP session progress and acceptance. Even
though the UAC sees that its offer has been answered and accepted,
the header lets the UAC know whether the invited PoC subscriber has
accepted the invite or just an intermediary has done the acceptance.

6.3. Applicability statement for the P-Answer-State header

The P-Answer-State header is applicable in the following
circumstances:

- In networks where there are UAs that engage in half-duplex
  communication where there is not the possibility for the invited
  user to verbally acknowledge the answering of the session as is
  normal in full duplex communication;

- Where the invited UA may automatically accept the session
  without manual acceptance;

- The network also contains intermediate network SIP servers that
  are trusted;
o The intermediate network SIP servers have knowledge of the current answer mode setting of the terminating UAS; and,

o The intermediate network SIP servers have knowledge of the media types and codecs likely to be accepted by the terminating UAS; and,

o The intermediate network SIP servers can provide buffering of the media in order to reduce the time for the inviting user to send media.

o The intermediate network SIP servers assume knowledge of the network topology and the existence of similar intermediate network SIP servers in the signaling path.

Such configurations are generally not applicable to the internet as a whole where such trust relationships do not exist.

In addition security issues have only been considered for networks which are trusted and use hop by hop security mechanisms and security issues with usage of this mechanism in the general internet have not been evaluated.

6.4. Usage of the P-Answer-State header

A UAS B2BUA or proxy MAY include a P-Answer-State header field in any 18x or 2xx response that does not contain an offer, sent in response to an offer contained in an INVITE as specified in [5]. Typically the P-Answer-State header field is included in either a 183 Session Progress or a 200 OK response. A UA that receives a REFER request to send an INVITE MAY also include a P-Answer-State header field in the sipfrag of a response included in a NOTIFY request it sends as a result of the implicit subscription created by the REFER request.

When the P-Answer-State header field contains the parameter "Unconfirmed" the UAC or proxy is indicating that it has information that hints that the final destination UAS for the INVITE request is likely to automatically accept the session but that this is unconfirmed and it is possible that the final destination UAS will first alert the user and require manual acceptance of the session or not accept the session request. When the P-Answer-State header field contains the parameter "Confirmed" the UAC or proxy is indicating that the destination UAS has accepted the session and is ready to receive media. The parameter value of "Confirmed" has the usual semantics of a 200 OK response containing an answer and is included for completeness. A parameter value of "Confirmed" is only included in a 200 OK response.
A received 18x response without a P-Answer-State header field SHOULD NOT be treated as an "Unconfirmed Response". A 18x response containing a P-Answer-State header field containing the parameter "Confirmed" MUST NOT be treated as a "Confirmed Response" because this is an invalid condition.

A 200 OK response without a P-Answer-State Header field MUST be treated as a "Confirmed Response".

6.4.1. Procedures at the UA (terminal)

A UAC (terminal) that receives an "Unconfirmed Response" containing an answer MAY send media as specified in [5], however there is no guarantee that the media will be received by the final recipient.

How a UAC confirms whether the media was or was not received by the final destination when it has received a 2xx response containing an "Unconfirmed Indication" is application specific and outside of the scope of this document. If the application is a conference then the mechanism specified in [5] could be used to determine that the invited user joined. Alternatively a BYE request could be received or the media could be placed on hold if the final destination UAS does not accept the session.

A UAC (terminal) that receives in response to a REFER request a NOTIFY request containing an "Unconfirmed Response" in a sipfrag in the body of the NOTIFY request that was received on a pre-existing dialog that was established by an INVITE request and for which there has been a successful offer-answer exchange according to [5] MAY send media, however there is no guarantee that the media will be received by the final recipient that was indicated in the Refer-To header in the original REFER request.

A UAC (terminal) that receives an "Unconfirmed Response" that does not contain an answer MAY buffer media until it receives another "Unconfirmed Response" containing an answer or a "Confirmed Response".

There are no P-Answer-State procedures for a terminal acting in the UAS role.

6.4.2. Procedures at the UA (PTT Server)

A PTT Server that acts as a back-to-back UA and receives at its UAS an INVITE request MAY include in any 18x or 2xx response that does not contain an offer, a P-Answer-State header field with the parameter "Unconfirmed" in the response if it has not yet received a "Confirmed Response" from the final destination UA and it has
information that hints that the final destination UA for the INVITE is likely to automatically accept the session.

A PTT Server that acts as a back-to-back UA that receives at its UAC a 18x response to an INVITE request containing a P-Answer-State header field with the parameter "Unconfirmed" in the response MAY include the P-Answer-State header field with the parameter "Unconfirmed" in a 2xx response its UAS sends as a result of receiving that response. Otherwise a PTT Server that acts as a back-to-back UA that receives at its UAC a 18x or 2xx response to an INVITE request containing a P-Answer-State header field in the response SHOULD include the P-Answer-State header field unmodified in the 18x or 2xx response its UAS sends as a result of receiving that response. If the response sent by the UAS is a 18x response then the P-Answer-State header field included in the response MUST contain a parameter of "Unconfirmed".

A PTT Server that acts as a back-to-back UA MAY include an answer in the "Unconfirmed Response" its UAS sends even if the "Unconfirmed Response" received by its UAC did not contain an answer.

If a PTT Server that acts as a back-to-back UA receives at its UAC a "Confirmed Response" then the UAS MAY include in the forwarded response a P-Answer-State header field with the parameter "Confirmed". If the PTT Server UAS previously sent an "Unconfirmed Response" as part of this dialog the UAS SHOULD include in the forwarded "Confirmed Response" a P-Answer-State header field with the parameter "Confirmed".

If a UAS of a PTT Server that acts as a back-to-back UA, includes an answer in a response along with a P-Answer-State header field with the parameter "Unconfirmed" then the UAS needs to be ready to receive media as specified in [5] and MAY buffer any media it receives until it receives a "Confirmed Response" from the final destination UA or until its buffer is full.

A PTT Server that acts as a back-to-back UA that’s UAS receives a REFER request to send an INVITE request to another UA as specified in [4], MAY generate a sipfrag of a 200 OK response containing a P-Answer-State header field with the parameter "Unconfirmed" prior to its UAC receiving a response to the Invite, if it has information that hints that the final destination UA for the INVITE is likely to automatically accept the session.

If a PTT Server that acts as a back-to-back UA that’s UAC sent an Invite request as a result of receiving a REFER Request, receives a 18x or 2xx response containing a P-Answer-State header field at its UAC, then its UAS SHOULD include the P-Answer-State header field and
its parameters from that response unmodified in the sipfrag of the response contained in a NOTIFY request that the UAS sends in response to the REFER request. If the sipfrag of the response sent in the NOTIFY request is a 18x response then the P-Answer-State header field included in the sipfrag of the response MUST contain a parameter of "Unconfirmed". If the UAC receives a "Confirmed Response" that does not contain a P-Answer-State header field then the UAS MAY include a P-Answer-State header field with the parameter "Confirmed" in the sipfrag of the response contained in a NOTIFY request sent in response to the REFER request.

A PTT Server that acts as a back-to-back UA that’s UAS previously sent a NOTIFY request containing a P-Answer-State header field with the parameter "Unconfirmed" in the sipfrag of a response included in the NOTIFY request, that subsequently receives at its UAC a "Confirmed Response" to the INVITE request sent as a result of the REFER request SHOULD include a P-Answer-State header field with the parameter "Confirmed" in the sipfrag of the response included in the subsequent NOTIFY request that its UAS sends as a result of receiving the "Confirmed Response".

If the REFER was received on an existing dialog established by an INVITE request for which there has been a successful offer-answer exchange the UAS MUST be ready to receive media as specified in [5] and MAY buffer any media it receives until its UAC receives a "Confirmed Response" from the final destination UA or until its buffer is full.

6.4.3. Procedures at the proxy server

SIP proxy servers do not need to understand the semantics of the P-Answer-State header field. As part of the regular SIP rules for unknown headers, a proxy will forward unknown headers.

A PTT Server that acts as a proxy MAY include a P-Answer-State header field with the parameter "Unconfirmed" in a 18x response that it originates compliant with [6] if it has information that hints that the final destination UA for the INVITE is likely to automatically accept the session.

A PTT Server that acts as a proxy MAY add a P-Answer-State header field with the parameter "Confirmed" to a "Confirmed Response".

7. Formal syntax

The mechanisms specified in this document is described in both prose and an augmented Backus-Naur Form (BNF) defined in [7]. Further,
several BNF definitions are inherited from SIP and are not repeated here. Implementers need to be familiar with the notation and contents of SIP [6] and [7] to understand this document.

7.1. P-Answer-State header syntax

The syntax of the P-Answer-State header is described as follows:

    P-Answer-State = "P-Answer-State" HCOLON answer-type
    answer-type = "Confirmed" / "Unconfirmed" / token *(SEMI generic-param)

7.2. Table of the new header

Table 1 extends the headers defined in this document to Table 2 in SIP [6], section 7.1 of the SIP-specific event notification [3] tables 1 and 2 in the SIP INFO method [13], tables 1 and 2 in Reliability of provisional responses in SIP [14], tables 1 and 2 in the SIP UPDATE method [15], tables 1 and 2 in the SIP extension for Instant Messaging [16], table 1 in the SIP REFER method [4], and table 2 in the SIP PUBLISH method [17]:

<table>
<thead>
<tr>
<th>Header field</th>
<th>where</th>
<th>proxy</th>
<th>ACK</th>
<th>BYE</th>
<th>CAN</th>
<th>INV</th>
<th>OPT</th>
<th>REG</th>
<th>SUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>P-Answer-State</td>
<td>18x,2xx</td>
<td>ar</td>
<td>-</td>
<td>-</td>
<td>o</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>P-Answer-State</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1

8. Example usage session flows

For simplicity some details such as intermediate proxies and 100 Trying responses are not shown in the following example flows.

8.1. Pre-arranged group call using on-demand session

The following flow shows Alice making a Pre-arranged Group Call using a Conference URI which has Bob on the member list. The session initiation uses the On-demand Session establishment mechanism where a SIP INVITE containing an SDP offer is sent by Alice’s terminal when Alice pushes her push to talk button.

In this example Alice’s PTT Server acts a Call Stateful SIP Proxy and Bob’s PTT Server which is aware that the current Answer Mode setting
of Bob’s terminal is set to Auto Answer acts as a B2BUA.

For simplicity the invitations by the Conference Focus to the other members of the group are not shown in this example.

<table>
<thead>
<tr>
<th>Alice’s Terminal</th>
<th>Alices’s PTT Server</th>
<th>Conference Focus</th>
<th>Bob’s PTT Server</th>
<th>Bob’s Terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>---(1)INVITE---&gt;</td>
<td>--(2)INVITE---&gt;</td>
<td>--(3)INVITE---&gt;</td>
<td>--(4)INVITE---&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>---(5)183------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---(6)200------</td>
<td></td>
<td>---(7)200------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---(8)ACK------&gt;</td>
<td></td>
<td>---(9)ACK------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>======Early Media Session=====</td>
<td>MEDIA BUFFERING</td>
<td>---(10)200------</td>
<td>---(11)ACK------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>---(12)200------</td>
<td>---(13)ACK------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>=========Media Session========&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2

F1 INVITE Alice -> Alices’s PTT Server

INVITE sip:FriendsOfAlice@example.org SIP/2.0   
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8   
Max-Forwards: 70   
To: "Alice’s Friends" <sip:FriendsOfAlice@example.org>   
From: "Alice" <sip:alice@example.org>;tag=1928301774   
Call-ID: a84b4c76e66710   
CSeq: 314159 INVITE   
Contact: <sip:alice@pc33.example.org>   
Content-Type: application/sdp   
Content-Length: 142

(SDP not shown)

F2 INVITE Alice’s PTT Server -> Conference Focus
INVITE sip:FriendsOfAlice@example.org SIP/2.0
Via: SIP/2.0/UDP AlicesPTTServer.example.org;branch=z9hG4bK77ef4c2312983.1
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8
Record-Route: <sip:AlicesPTTServer.example.org>
Max-Forwards: 69
To: "Alice's Friends" <sip:FriendsOfAlice@example.org>
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.example.org>
Content-Type: application/sdp
Content-Length: 142

(SDP not shown)

The Conference Focus explodes the Conference URI and Invites Bob

F3 INVITE Conference Focus -> Bob’s PTT Server

INVITE sip:bob@example.com SIP/2.0
Via: SIP/2.0/UDP AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=2178309898
Call-ID: e60a4c784b6716
CSeq: 301166605 INVITE
Contact: <sip:AlicesConferenceFocus.example.org>
Content-Type: application/sdp
Content-Length: 142

(SDP not shown)

F4 INVITE Bob’s PTT Server -> Bob

INVITE sip:bob@example.com SIP/2.0
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bKa27bc93
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 INVITE
Contact: <sip:BobsPTTServer.example.com>
Content-Type: application/sdp
Content-Length: 142
(SDP not shown)

F5 183 Session Progress Bob’s PTT Server -> Conference Focus

SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP
AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
To: "Bob" <sip:bob@example.com>;tag=a6c85cf
From: "Alice’s Friends"
  <sip:FriendsOfAlice@example.org>;tag=2178309898
Contact: <sip:BobsPTTServer.example.com>
Call-ID: e60a4c784b6716
CSeq: 301166605 INVITE
P-Answer-State: Unconfirmed
Content-Length: 0

F6 200 OK Conference Focus -> Alice’s PTT Server

SIP/2.0 200 OK
Via: SIP/2.0/UDP
AlicesPTTServer.example.org;branch=z9hG4bK77ef4c2312983.1
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8
Record-Route: <sip:AlicesPTTServer.example.org>
To: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:AlicesConferenceFocus.example.org>
P-Answer-State: Unconfirmed
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F7 200 OK Alice’s PTT Server -> Alice

SIP/2.0 200 OK
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8
Record-Route: <sip:AlicesPTTServer.example.org>
To: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:AlicesConferenceFocus.example.org>
P-Answer-State: Unconfirmed
Content-Type: application/sdp
Content-Length: 131
F8 ACK Alice -> Alice’s PTT Server

ACK sip:AlicesConferenceFocus.example.org SIP/2.0
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds9
Route: <sip:AlicesPTTServer.example.org>
Max-Forwards: 70
To: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 ACK
Content-Length: 0

F9 ACK Alice’s PTT Server -> Conference Focus

ACK sip:AlicesConferenceFocus.example.org SIP/2.0
Via: SIP/2.0/UDP
AlicesPTTServer.example.org;branch=z9hG4bK77ef4c2312983.1
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds9
Max-Forwards: 69
To: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 ACK
Content-Length: 0

The early half duplex media session between Alice and the Conference Focus is now established and the Conference Focus buffers the media it receives from Alice.

F10 200 OK Bob -> Bob’s PTT Server

SIP/2.0 200 OK
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bKa27bc93
To: "Bob" <sip:bob@example.com>;tag=d28119a
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 INVITE
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F11 ACK Bob’s PTT Server -> Bob
ACK sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bKa27bc93
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>;tag=d28119a
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 ACK
Content-Length: 0

F12 200 OK Bob’s PTT Server --> Conference Focus

SIP/2.0 200 OK
Via: SIP/2.0/UDP
AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
To: "Bob" <sip:bob@example.com>;tag=a6670811
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=2178309898 Call-ID: e60a4c784b6716
Contact: <sip:BobsPTTServer.example.com>
CSeq: 301166605 INVITE
P-Answer-State: Confirmed
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F13 ACK Conference Focus --> Bob’s PTT Server

ACK sip:BobsPTTServer.example.com SIP/2.0
Via: SIP/2.0/UDP
AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>;tag=a6670811
From: "Alice’s Friends"
<sip:FriendsOfAlice@example.org>;tag=2178309898
Call-ID: e60a4c784b6716
CSeq: 301166605 ACK
Content-Length: 0

The media session between Alice and Bob is now established and the Conference Focus forwards the buffered media to Bob.

8.2. 1-1 Call using pre-established session

The following flow shows Alice making a 1-1 Call to Bob using a pre-established session. A pre-established session is where a dialog is established with Alice’s PTT Server using a SIP INVITE SDP offer
answer exchange to pre-negotiate the codecs and other media parameters to be used for media sessions ahead of Alice initiating a communication. When Alice initiates a communication to Bob a SIP REFER is used to Request Alice’s PTT Server to send an INVITE to Bob. In this example Bob’s Terminal does not use the Pre-established Session mechanism.

In this example Alice’s PTT Server acts a B2BUA and also performs the Conference Focus function. Bob’s PTT Server which is aware that the current Answer Mode setting of Bob’s terminal is set to Auto Answer acts as a B2BUA.

<table>
<thead>
<tr>
<th>Alice’s Terminal</th>
<th>Alice’s PTT Server</th>
<th>Bob’s PTT Server</th>
<th>Bob’s Terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conference Focus</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3**

F1 INVITE Alice -> Alice's PTT Server
INVITE sip: AlicesConferenceFactoryURI.example.org SIP/2.0
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8
Max-Forwards: 70
To: <sip:AlicesConferenceFactoryURI.example.org>
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.example.org>
Content-Type: application/sdp
Content-Length: 142

(SDP not shown)

F2 200 OK Alice’s PTT Server -> Alice

SIP/2.0 200 OK
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds8
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:AlicesPre-establishesSession@AlicesPTTServer.example.org>
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F3 ACK Alice -> Alice’s PTT Server

ACK sip:AlicesPre-establishesSession@AlicesPTTServer.example.org SIP/2.0
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnashds9
Max-Forwards: 70
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 ACK
Content-Length: 0

Alices’s terminal has established a Pre-established Session with Alice’s PTT Server. All the media parameters are pre-negotiated for use at communication time.

Alice initiates a Communication to Bob

F4 REFER Alice -> Alices’s PTT Server
REFER sip:AlicesPre-establishesSession@AlicesPTTServer.example.org
SIP/2.0
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnasahds8
Max-Forwards: 70
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314160 REFER
Refer-To: "Bob" <sip:bob@example.com>
Contact: <sip:alice@pc33.example.org>

F5 202 ACCEPTED Alice’s PTT Server -> Alice

SIP/2.0 202 ACCEPTED
Via: SIP/2.0/UDP pc33.example.org;branch=z9hG4bKnasahds8
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314160 REFER
Contact: <sip:AlicesPre-establishesSession@AlicesPTTServer.example.org>

F6 INVITE Conference Focus -> Bob’s PTT Server

INVITE sip:bob@example.com SIP/2.0
Via: SIP/2.0/UDP
AlicesConferenceFocus.example.org;branch=z9hG4bk4721d8
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>
From: "Alice" <sip:Alice@example.org>;tag=2178309898
Call-ID: e60a4c76e707116
CSeq: 301166605 INVITE
Contact: <sip:AlicesConferenceFocus.example.org>
Content-Type: application/sdp
Content-Length: 142

(SDP not shown)

F7 INVITE Bob’s PTT Server -> Bob

INVITE sip:bob@example.com SIP/2.0
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bKas27bc93
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>
From: "Alice" <sip:Alice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 INVITE
Contact: <sip:BobsPTTServer.example.com>
F8 183 Session Progress Bob’s PTT Server -> Conference Focus

SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
To: "Bob" <sip:bob@example.com>;tag=a6c85cf
From: "Alice" <sip:Alice@example.org>;tag=2178309898
Call-ID: e60a4c784b6716
Contact: <sip:BobsPTTServer.example.com>
CSeq: 301166605 INVITE
P-Answer-State: Unconfirmed
Content-Length: 0

F9 NOTIFY Alices’s PTT Server -> Alice

NOTIFY sip:alice@pc33.example.org SIP/2.0
Via: SIP/2.0/UDP AlicesPre-establishesSession@
AlicesPTTServer.example.org;branch=z9hG4bKnashds8
Max-Forwards: 70
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314161 NOTIFY
Contact: <sip:AlicesPre-establishesSession@
AlicesPTTServer.example.org>
Event: refer
Subscription-State: Active;Expires=60
Content-Type: message/sipfrag;version=2.0
Content-Length: 99

SIP/2.0 183 Session Progress
To: "Bob" <sip:bob@example.com>;tag=d28119a
P-Answer-State: Unconfirmed

F10 200 OK Alice -> Alice’s PTT Server

SIP/2.0 200 OK
Via: SIP/2.0/UDP AlicesPre-establishesSession@
AlicesPTTServer.example.org;branch=z9hG4bKnashds8
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314161 NOTIFY
The early half duplex media session between Alice and the Conference Focus is now established and the Conference Focus buffers the media it receives from Alice.

F11 200 OK Bob -> Bob’s PTT Server

SIP/2.0 200 OK
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bK927bc93
To: "Bob" <sip:bob@example.com>;tag=d28119a
From: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 INVITE
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F12 ACK Bob’s PTT Server -> Bob

ACK sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP BobsPTTServer.example.com;branch=z9hG4bK927bc93
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>;tag=d28119a
From: "Alice" <sip:Alice@example.org>;tag=781299330
Call-ID: 6eb4c66a847710
CSeq: 478209 ACK
Content-Length: 0

F13 200 OK Bob’s PTT Server -> Conference Focus

SIP/2.0 200 OK
Via: SIP/2.0/UDP AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
To: "Bob" <sip:bob@example.com>;tag=a6670811
From: "Alice’s Friends" <sip:FriendsOfAlice@example.org>;tag=2178309898
Call-ID: e60a4c784b6716
CSeq: 301166605 INVITE
P-Answer-State: Confirmed
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)

F14 ACK Conference Focus -> Bob’s PTT Server
ACK sip:BobsPTTServer.example.com SIP/2.0
Via: SIP/2.0/UDP AlicesConferenceFocus.example.org;branch=z9hG4bK4721d8
Max-Forwards: 70
To: "Bob" <sip:bob@example.com>;tag=a6670811
From: "Alice" <sip:Alice@example.org>;tag=1928301774
Call-ID: e60a4c784b6716
CSeq: 301166605 ACK
Content-Length: 0

The media session between Alice and Bob is now established and the Conference Focus forwards the buffered media to Bob.

F15 NOTIFY Alices’s PTT Server -> Alice

NOTIFY sip:alice@pc33.example.org SIP/2.0
Via: SIP/2.0/UDP AlicesPre-establishesSession@AlicesPTTServer.example.org;branch=z9hG4bKnashds8
Max-Forwards: 70
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314162 NOTIFY
Contact: <sip:AlicesPre-establishesSession@AlicesPTTServer.example.org>
Event: refer
Subscription-State: Active;Expires=60
Content-Type: message/sipfrag;version=2.0
Content-Length: 83

SIP/2.0 200 OK
To: "Bob" <sip:bob@example.com>;tag=d28119a
P-Answer-State: Confirmed

F16 200 OK Alice -> Alice’s PTTServer

SIP/2.0 200 OK
Via: SIP/2.0/UDP AlicesPre-establishesSession@AlicesPTTServer.example.org;branch=z9hG4bKnashds8
To: <sip:AlicesConferenceFactoryURI.example.org>;tag=c70ef99
From: "Alice" <sip:alice@example.org>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314162 NOTIFY

9. Security considerations

The information returned in the P-Answer-State header is not viewed...
as particularly sensitive. Rather, it is informational in nature, providing an indication to the UAC that delivery of any media sent as a result of an answer in this response is not guaranteed. An eavesdropper cannot gain any useful information by obtaining the contents of this header.

End-to-end protection is not appropriate because the P-Answer-State header is used and added by proxies and intermediate UAs. As a result, a "malicious" proxy between the UAs or attackers on the signaling path could add or remove the header or modify the contents of the header value. This attack either denies the caller the knowledge that the callee has yet to be contacted or falsely indicates that the callee has yet to be contacted when they have already answered. The falsely indicating that the callee has yet to be contacted when they have already answered attack could result in the caller deciding not transmit media because they do not wish to have their media stored by an intermediary even though in reality the callee has answered. The denying the callee the additional knowledge that the callee has yet to be contacted attack does not appear to be a significant concern since this is the same as the situation when a B2BUA sends a 200 OK before the callee has answered without the use of this extension.

It is therefore necessary to protect the messages between proxies and implementation SHOULD use a transport that provides integrity and confidentiality between the signaling hops. The TLS based signaling in SIP can be used to provide this protection.

10. IANA considerations

10.1. Registration of Header Fields

This document defines a private SIP extension header field (beginning with the prefix "P-" ) based on the registration procedures defined in RFC 3427 [18].

The following rows shall be added to the "Header Fields" section of the SIP parameter registry:

<table>
<thead>
<tr>
<th>Header Name</th>
<th>Compact Form</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>P-Answer-State</td>
<td>[RFCXXXX]</td>
<td>[RFCXXXX]</td>
</tr>
</tbody>
</table>

Editor Note: [RFCXXXX] should be replaced with the designation of this document.
10.2. Registration of header field parameters

This document defines parameters for the header fields defined in the preceding section. The header field named "P-Answer-State" may take the values "Unconfirmed", or "Confirmed".

The following rows shall be added to the "Header Field Parameters and Parameter Values" section of the SIP parameter registry:

<table>
<thead>
<tr>
<th>Header Field</th>
<th>Parameter Name</th>
<th>Predefined Values</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>P-Answer-State</td>
<td>Unconfirmed</td>
<td>Yes</td>
<td>[RFCXXXX]</td>
</tr>
<tr>
<td>P-Answer-State</td>
<td>Confirmed</td>
<td>Yes</td>
<td>[RFCXXXX]</td>
</tr>
</tbody>
</table>

Editor Note: [RFCXXXX] should be replaced with the designation of this document.

11. Acknowledgements

The authors would like to thank Cullen Jennings, Jeroen van Bemmel, Paul Kyzivat, Dale Worley, Dean Willis, Rohan Mahay, Christian Schmidt, Mike Hammer, and Miguel Garcia-Martin for their comments that contributed to the progression of this work. The authors would also like to thank the OMA POC Working Group members for their support of this document and in particular Tom Hiller for presenting the concept of the P-Answer-State header to SIPPPING at IETF#62.

12. References

12.1. Normative references

Session Description Protocol (SDP)", RFC 3264, June 2002.


12.2. Informative references


Authors’ Addresses

Andrew Allen (editor)
Research in Motion (RIM)
122 West John Carpenter Parkway, Suite 430
Irving, Texas  75039
USA
Phone: unlisted
Fax:   unlisted
Email: aallen@rim.com

Jan Holm
Ericsson
Gotalandsvagen 220
Stockholm  612526
Sweden
Phone: unlisted
Fax:   unlisted
Email: Jan.Holm@ericsson.com

Tom Hallin
Motorola
1501 W Shure Drive
Arlington Heights  60004
USA
Phone: unlisted
Fax:   unlisted
Email: thallin@motorola.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2006). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.