SIPPING Working Group A. Allen, Ed.
Internet-Draft Research in Motion (RIM)
Intended status: Experimental J. Holm
Expires: September 5, 2007 Ericsson
T. Hallin
Motorola
March 4, 2007

The P-Answer-State Header Extension to the Session Initiation Protocol for the Open Mobile Alliance Push-to-talk over Cellular

draft-allen-sipping-poc-p-answer-state-header-05

Status of this Memo

Copyright Notice

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.

Table of Contents

1. Overall Applicability

The SIP extension specified in this document makes certain assumptions regarding network topology and the existence 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 6.3.

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 User Agent (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 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].

The terms "PTT Server" (Push-to-talk 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 can act as a SIP Proxy as defined in [2] or a back-to-

back UA (B2BUA) as defined in [2] based on the functions it needs to perform. There can 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 SIP 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 a Session Description Protocol (SDP) answer).

4. Background for the Extension

The PoC terminal could 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 can 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 could support both or only one of these modes of operation. The user can 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 [3] utilizes PTT Servers within the network that can perform such roles as a conference focus [10], a real-time transport protocol (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 [11]. In addition, particularly when multiple domains are involved in the session, more than one PTT server could be involved in the signaling path for the session. Also the PTT Server that performs the buffering might 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 User Agent Server (UAS) to answer in a particular mode is defined in [12]. However a mechanism is required for a PTT Server to relay the "Unconfirmed Indication" in a response back towards the originating SIP User Agent Client (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 SIP 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 typically 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 the called SIP 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 SIP UA's answering mode faster than the called UA can deliver an automatically generated SIP 200 (OK) response.

This document proposes a new SIP header field the P-Answer-State header field to support an "Unconfirmed Indication". The new SIP header field can be optionally included in a response to a SIP INVITE request or in a sipfrag of a response included in a SIP NOTIFY request sent as a result of a SIP REFER request that requests a SIP INVITE request 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.

Thus, when a PTT Server forwards a SIP INVITE request and knows that the called UA is likely to be in Automatic Answer mode, it also generates a SIP 183 provisional response with a P-Answer-State header field with a parameter of "Unconfirmed" to signal to upstream PTT Servers that they can 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 SIP 200 (OK) response for the caller's SIP UA with an appropriate answer.

When the called UA generates a SIP 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 SIP 200 (OK) response a P-Answer-State header field with a parameter of "Confirmed". The SIP 200 (OK) response is absorbed by the PTT Server that is buffering the caller's media, as it has already generated a SIP 200 (OK) response. The buffering PTT Server then starts playing out the buffered media.

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 permitting it 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 can send a SIP 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 can be included in a provisional or final response to a SIP INVITE request or in the sipfrag of a SIP NOTIFY request sent as a result of a SIP REFER request to send a SIP 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 SIP 200 (OK) response when the final response is received from the destination UA.

NOTE It is not intended that multiple PTT servers perform buffering serially. If a PTT server includes an answer along with P-Answer-State header field with the value "Unconfirmed" in a provisional response then a receiving PTT Server can determine whether it buffers the media or whether to forward the media and allow the downstrean PTT Server that sent the "Unconfirmed Indication" to buffer the media. It is intended that if a PTT Server buffers media it does so until a final "Confirmed Indication" is received and therefore serial buffering by multiple PTT Servers does not take place

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 SIP 200 (OK) response or in the sipfrag as defined in [4] of a SIP NOTIFY request as defined in [5] sent as a result of a SIP REFER request as defined in [6]. 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 SIP 200 (OK) response or in the sipfrag of a SIP NOTIFY request.

Provisional responses containing the P-Answer-State header field can be sent reliably using the mechanism defined in [13] but this is not required. This is a performance optimization and the impact of a provisional response sent unreliably failing to arrive is simply that buffering does not take place. However, if the provisional responses are sent reliably and the provisional response fails to arrive the time taken for the provisional response sender to timeout on the receipt of a SIP PRACK request is likely to be such that by the time the provisional response has been resent the "Confirmed Response" could have already been received. Because when provisional responses that contain an answer are sent reliably, the 200 (OK) response for the SIP INVITE request cannot be sent before the SIP PRACK request is received, sending provisional responses reliably could potentially delay the sending of the "Confirmed Response".

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 a SIP INVITE request from an inviting SIP UAC even if no invited SIP UAS has accepted the SIP 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 SIP INVITE request 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 SIP 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 SIP 183 (Session Progress) response or SIP 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 SIP 183 response and SIP 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 [14] in the SIP 183 response and SIP 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 [15]. 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 SIP INVITE request 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:

- o 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;

- o Where the invited UA can automatically accept the session without user intervention;

- o 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 with transitive trust 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 SIP 18x or 2xx response that does not contain an offer, sent in response to an offer contained in a SIP INVITE request as specified in [7]. Typically the P-Answer-State header field is included in either a SIP 183 Session Progress or a SIP 200 (OK) response. A UA that receives a SIP REFER request to send a SIP INVITE request MAY also include a P-Answer-State header field in the sipfrag of a response included in a SIP NOTIFY request it sends as a result of the implicit subscription created by the SIP 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 SIP 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 SIP 200 (OK) response containing an answer and is included for completeness. A parameter value of "Confirmed" is only included in a SIP 200 (OK) response or in the sipfrag of a 200 (OK) contained in the body of a SIP NOTIFY request.

A received SIP 18x response without a P-Answer-State header field SHOULD NOT be treated as an "Unconfirmed Response". A SIP 18x response containing a P-Answer-State header field containing the parameter "Confirmed" MUST NOT be treated as a "Confirmed Response" because this in an invalid condition.

A SIP 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 [7], 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 SIP 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 [7] could be used to determine that the invited user joined. Alternatively a SIP 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 SIP REFER request, a SIP NOTIFY request containing an "Unconfirmed Response" in a sipfrag in the body of the SIP NOTIFY request related to a dialog 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 SIP REFER request. The dialog could be related either because the SIP REFER request was sent on the same dialog or because the SIP REFER request contained a Target-Dialog header as defined in [16] that identified the dialog.

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 receives a SIP INVITE request at the UAS part of its back-to-back UA MAY include in any SIP 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 that the final destination UA for the SIP INVITE request is likely to automatically accept the session.

A PTT Server that receives a SIP 18x response to a SIP INVITE request containing a P-Answer-State header field with the parameter "Unconfirmed" at the UAC part of its back-to-back UA MAY include the P-Answer-State header field with the parameter "Unconfirmed" in a SIP

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

The UAS part of the back-to-back UA of a PTT Server MAY include an answer in the "Unconfirmed Response" it sends even if the "Unconfirmed Response" received by the UAC part of the back-to-back UA did not contain an answer.

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

If the UAS part of the back-to-back UA of a PTT Server, includes an answer in a response along with a P-Answer-State header field with the parameter "Unconfirmed" then the UAS part of its back-to-back UA needs to be ready to receive media as specified in [7] 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 UAS part of the back-to-back UA of a PTT Server that receives a SIP REFER request to send a SIP INVITE request to another UA as specified in [6], MAY generate a sipfrag of a SIP 200 (OK) response containing a P-Answer-State header field with the parameter "Unconfirmed" prior to the UAC part of its back-to-back UA receiving a response to the SIP INVITE request, if it has information that hints that the final destination UA for the SIP INVITE request is likely to automatically accept the session.

If the UAC part of a back-to-back UA of a PTT Server sent a SIP INVITE request as a result of receiving a SIP REFER Request, receives a SIP 18x or 2xx response containing a P-Answer-State header field at the UAC part of its back-to-back UA, then the UAS part of its back-to-back UA SHOULD include the P-Answer-State header field and its parameters from that response unmodified in the sipfrag of the response contained in a SIP NOTIFY request that the UAS part of its

back-to-back UA sends in response to the SIP REFER request. If the sipfrag of the response sent in the SIP NOTIFY request is a SIP 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 part of its back-to-back UA receives a "Confirmed Response" that does not contain a P-Answer-State header field then the UAS part of its back-to-back UA MAY include a P-Answer-State header field with the parameter "Confirmed" in the sipfrag of the response contained in a SIP NOTIFY request sent in response to the SIP REFER request.

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

If the SIP REFER request related to an existing dialog established by a SIP INVITE request for which there has been a successful offer-answer exchange the UAS part of its back-to-back UA MUST be ready to receive media as specified in [7] and MAY buffer any media it receives until the UAC part of its back-to-back UA receives a "Confirmed Response" from the final destination UA or until its buffer is full. The dialog could be related either because the SIP REFER request was sent on the same dialog or because the SIP REFER request contained a Target-Dialog header as defined in [16] that identified the dialog.

A PTT Server that buffers media SHOULD be prepared for the possibility of not receiving a "Confirmed Response" and SHOULD release the session if a "Confirmed Response" is not received before the buffer overflows.

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 SIP 18x response that it originates compliant with [2] if it has information that hints that that the final destination UA for the SIP INVITE request 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 [8]. 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 [2] and [8] 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
                 *(SEMI generic-param)
answer-type = "Confirmed" / "Unconfirmed" / token
```

7.2. Table of the New Header

Table 1 provides the additional table entries for the P-Answer-State header needed to extend Table 2 in SIP [2], section 7.1 of the SIP-specific event notification [5] tables 1 and 2 in the SIP INFO method [17], tables 1 and 2 in Reliability of provisional responses in SIP [13], tables 1 and 2 in the SIP UPDATE method [18], tables 1 and 2 in the SIP extension for Instant Messaging [19], table 1 in the SIP REFER method [6], and table 2 in the SIP PUBLISH method [20]:

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	SUB
P-Answer-State	18x,2xx	ar	-	-	-	o	-	-	-

Header field	where	proxy	NOT	PRA	INF	UPD	MSG	REF	PUB
P-Answer-State	R		-	-	-	-	-	-	-

Figure 1

8. Example Usage Session Flows

For simplicity some details such as intermediate proxies and SIP 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 request containing an SDP offer is sent by Alices'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.

```
   Alice's        Alices's      Conference     Bob's          Bob's
   Terminal      PTT Server       focus      PTT Server    Terminal
      |              |              |             |              |
      |--(1)INVITE-->|              |             |              |
      |              |--(2)INVITE-->|             |              |
      |              |              |--(3)INVITE->|              |
      |              |              |             |--(4)INVITE-->|
      |              |              |<--(5)183----|              |
      |              |<---(6)200----|             |              |
      |<---(7)200----|              |             |              |
      |----(8)ACK--->|              |             |              |
      |              |---(9)ACK---->|             |              |
      |              |              |             |              |
      |=====Early Media Session====>|             |              |
      |              |            MEDIA           |              |
      |              |          BUFFERING         |              |
      |              |              |             |<---(10)200---|
      |              |              |             |---(11)ACK--->|
      |              |              |<--(12)200---|              |
      |              |              |--(13)ACK--->|              |
      |              |              |             |              |
      |              |              |========Media Session======>|
      |              |              |             |              |
      |              |              |             |              |
```

Figure 2

1 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)
```

2 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

3 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)
```

4 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)

5 183 (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
Call-ID: e60a4c784b6716
Contact: <sip:BobsPTTServer.example.com>
CSeq: 301166605 INVITE
P-Answer-State: Unconfirmed
Content-Length: 0

6 200 (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)

7 200 (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
```

(SDP not shown)

8 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
```

9 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.

```
10 200 (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)

11 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

12 200 (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)

13 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 Alices'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 request is used to request Alice's PTT Server to send a SIP INVITE request 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.

```
      Alice's                Alice's               Bob's          Bob's
      Terminal             PTT Server /          PTT Server     Terminal
                        Conference Focus
         |                       |                  |                |
         |-----(1)INVITE-- ----->|                  |                |
         |<-----(2)200-----------|                  |                |
         |-------(3)ACK--------->|                  |                |
         |                       |                  |                |
         |                       |                  |                |
         |                       |                  |                |
         |----(4)REFER---------->|                  |                |
         |<-----(5)202-----------|                  |                |
         |                       |----(6)INVITE---->|                |
         |                       |                  |--(7)INVITE---->|
         |                       |                  |                |
         |                       |<----(8)183-------|                |
         |<---(9)NOTIFY----------|                  |                |
         |-----(10)200---------->|                  |                |
         |                       |                  |                |
         |=Early Media Session==>|                  |                |
         |                     MEDIA                |                |
         |                   BUFFERING              |                |
         |                       |                  |<---(11)200-----|
         |                       |                  |---(12)ACK----->|
         |                       |<----(13)200------|                |
         |                       |-----(14)ACK----->|                |
         |                       |===========Media Session==========>|
         |                       |                  |                |
         |<---(15)NOTIFY---------|                  |                |
         |-----(16)200---------->|                  |                |
         |                       |                  |                |
```

Figure 3

1 INVITE Alice -> Alices'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)
```

2 200 (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-establishedSession@
AlicesPTTServer.example.org>
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)
```

3 ACK Alice -> Alice's PTT Server

```
ACK sip:AlicesPre-establishedSession@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

4 REFER Alice -> Alices's PTT Server

```
REFER sip:AlicesPre-establishedSession@AlicesPTTServer.example.org
     SIP/2.0
Via: SIP/2.0/UDP pc33.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: 314160 REFER
Refer-To: "Bob" <sip:bob@example.com>
Contact: <sip:alice@pc33.example.org>
```

5 202 (ACCEPTED) Alice's PTT Server -> Alice

```
SIP/2.0 202 ACCEPTED
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: 314160 REFER
Contact: <sip:AlicesPre-establishedSession@
AlicesPTTServer.example.org>
```

6 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: e60a4c784b6716
CSeq: 301166605 INVITE
Contact: <sip:AlicesConferenceFocus.example.org>
Content-Type: application/sdp
Content-Length: 142

(SDP not shown)
```

7 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" <sip:Alice@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)
```

8 183 (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
```

9 NOTIFY Alices's PTT Server -> Alice

```
NOTIFY sip:alice@pc33.example.org SIP/2.0
Via: SIP/2.0/UDP
     AlicesPre-establishedSession@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-establishedSession@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
```

10 200 (OK) Alice -> Alice's PTT Server

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
     AlicesPre-establishedSession@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.

11 200 (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
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)
```

12 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
Contact: <sip:BobsPTTServer.example.com>
CSeq: 301166605 INVITE
P-Answer-State: Confirmed
Content-Type: application/sdp
Content-Length: 131

(SDP not shown)
```

14 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.

15 NOTIFY Alices's PTT Server -> Alice

```
NOTIFY sip:alice@pc33.example.org SIP/2.0
Via: SIP/2.0/UDP
     AlicesPre-establishedSession@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-establishedSession@
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
```

16 200 (OK) Alice -> Alice's PTTServer

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
     AlicesPre-establishedSession@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 confidentially between the signaling hops. The Transport Layer Security (TLS) [9] based signaling in SIP can be used to provide this protection.

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

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 [21].

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

Header Name	Compact Form	Reference
P-Answer-State		[RFCXXXX]

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

11. Acknowledgements

The authors would like to thank Jon Peterson, 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 SIPPING at IETF#62.

12. References

12.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[3] OMA, "Push to talk over Cellular - Architecture", OMA-AD-PoC V1_0_1, 20061128 A, November 2006.

[4] Sparks, R., "Internet Media Type message/sipfrag", RFC 3420, November 2002.

[5] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[6] Sparks, R., "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003.

[7] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[8] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005.

[9] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

12.2. Informative References

[10] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol (SIP)", RFC 4353, February 2006.

[11] Garcia-Martin, M., "A Session Initiation Protocol (SIP) Event Package and Data Format for Various Settings in Support for the Push-to-Talk over Cellular (PoC) Service", RFC 4354, January 2006.

[12] Willis, D. and A. Allen, "Requesting Answering Modes for the Session Initiation Protocol (SIP)", draft-ietf-sip-answermode-01 (work in progress), May 2006.

[13] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, June 2002.

[14] Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, December 2002.

[15] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, August 2004.

[16] Rosenberg, J., "Request Authorization through Dialog Identification in the Session Initiation Protocol (SIP)", RFC 4538, June 2006.

[17] Donovan, S., "The SIP INFO Method", RFC 2976, October 2000.

[18] Rosenberg, J., "The Session Initiation Protocol (SIP) UPDATE Method", RFC 3311, October 2002.

[19] Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C., and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.

[20] Niemi, A., "Session Initiation Protocol (SIP) Extension for Event State Publication", RFC 3903, October 2004.

[21] Mankin, A., Bradner, S., Mahy, R., Willis, D., Ott, J., and B. Rosen, "Change Process for the Session Initiation Protocol (SIP)", BCP 67, RFC 3427, December 2002.

Authors' Addresses

Andrew Allen (editor)
Research in Motion (RIM)
102 Decker Court, Suite 100
Irving, Texas 75062
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

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).